

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО

Факультет інформатики та обчислювальної техніки

(назва факультету, інституту)

Кафедра автоматизованих систем обробки інформації і управління

(назва кафедри)

"На правах рукопису"

УДК 004.93(015.7)

«До захисту допущено»

Завідувач кафедри

О.А.Павлов

(підпис)

(ініціали, прізвище)

“ ” 20 18 р.

МАГІСТЕРСЬКА ДИСЕРТАЦІЯ

на здобуття ступеня магістра

за спеціальністю 126 Інформаційні системи та технології

(код та назва спеціальності)

ОПП

Інформаційні управляючі системи та технології

(код та назва спеціалізації)

на тему: Моделювання та аналіз протоколів багатошляхової маршрутизації
для контролера програмно-конфігурованої мережі

Виконала: студентка

VI курсу Групи ІС-72мп

(шифр групи)

Пастрелло Ніколь

(прізвище, ім'я, по батькові)

(підпис)

Науковий керівник

старший викладач, к.т.н., Коган А.В.

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Консультант

к.т.н., доц. Жданова О.Г.

(науковий ступінь, вчене звання, прізвище, ініціали)

(підпис)

Рецензент

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

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

Студент

(підпис)

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. ІГОРЯ СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної техніки
(повна назва)

Кафедра автоматизованих систем обробки інформації та управління
(повна назва)

Рівень вищої освіти другий (магістерський) за освітньо-професійною програмою

Спеціальність 126 Інформаційні системи та технології
(код і назва)

ОПП Інформаційні управляючі системи та технології
(код і назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри
О.А.Павлов
(підпис) (ініціали, прізвище)

« » грудня 2018 р.

ЗАВДАННЯ

на магістерську дисертацію студенту

Пастрелло Ніколь

(прізвище, ім'я, по батькові)

1. Тема дисертації Моделювання та аналіз протоколів багатошляхової маршрутизації для контролера програмно-конфігурованої мережі

науковий керівник дисертації Коган Алла Вікторівна, к.т.н.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затвержені наказом по університету від “ 7 ” листопада 2018 р. № 4112с

2. Строк подання студентом дисертації “ 3 ” грудня 2018 р.

3. Об'єкт дослідження Процес маршрутизації у програмно-конфігурованій мережі

4. Перелік завдань, які потрібно розробити виконати аналіз традиційної архітектури мереж та архітектури програмно-конфігурованих мереж; розробити алгоритм для протокола маршрутизації, що враховує особливості програмно-конфігурованої мережі; промоделювати роботу розробленого алгоритму; дослідити ефективність розробленого алгоритму

5. Орієнтовний перелік ілюстративного матеріалу схема структурна
варіантів використання; схема структурна діяльності; схема бази даних;
блок-схема алгоритму маршрутизації; результати експериментальних
досліджень; схема структурна класів; копії екранних форм

6. Орієнтовний перелік публікацій дві публікацій: одна стаття у фаховому
виданні, одні тези доповідей на науковій конференції

7. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

8. Дата видачі завдання “ 29 ” жовтня 20 18 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів магістерської дисертації	Примітка
1	<i>Систематизація результатів огляду літератури</i>	02.11.2018	
2	<i>Порівняльний аналіз існуючих протоколів маршрутизації у програмно-конфігурованих мережах</i>	4.11.2018	
3	<i>Постановка та формалізація математичної моделі задачі маршрутизації</i>	8.11.2018	
4	<i>Розробка алгоритму розв'язання задачі маршрутизації</i>	15.11.2018	
5	<i>Розробка інформаційного та програмного забезпечення</i>	23.11.2018	
7	<i>Проведення експериментальних досліджень розробленого алгоритму</i>	27.11.2018	
8	<i>Оформлення документації</i>	29.11.2018	
9	<i>Подання роботи на попередній захист</i>	30.11.2018	
10	<i>Подання роботи на основний захист</i>	03.12.2018	

Студент

_____ (підпис)

Н.Пастрелло

_____ (ініціали, прізвище)

Науковий керівник

_____ (підпис)

А.В.Коган

_____ (ініціали, прізвище)

Реферат

Магістерська дисертація: 87 с., 21 рис., 26 табл., 1 додаток, 21 джерел.

Актуальність. Кожен день збільшується кількість і різноманітність пристроїв комп'ютерної мережі. Розвиток різних технологій призвели до того, що процес контролю мережі великих розмірів стає практично не контрольованим. Комп'ютерні мережі складаються з великої кількості окремих елементів, де кожен відповідає за себе.

Тому науковою задачею є використання нового підходу до архітектури мережі такої як SDN, за допомогою якої можна оптимізувати транспортні потоки та протоколи.

Зв'язок роботи з науковими програмами, планами, темами. Робота виконувалась на кафедрі автоматизованих систем обробки інформації та управління Національного технічного університету України «Київський політехнічний інститут ім. Ігоря Сікорського» в рамках теми «Ефективні методи розв'язання задач теорії розкладів» (№ ДР 0117U000919).

Мета і завдання дослідження – підвищення ефективності організації маршрутизації в SDN шляхом формування множини шляхів, що не перетинаються.

Для досягнення мети необхідно виконати наступні **завдання**:

- виконати аналіз традиційної архітектури мереж та архітектури програмно-конфігурованих мереж;
- розробити алгоритми для протоколу маршрутизації, що враховує особливості програмно-конфігурованої мережі;
- промодельовати роботу розробленого алгоритму;
- розробити програмну реалізацію розробленого протоколу маршрутизації;
- дослідити ефективність розробленого алгоритму.

Об'єкт дослідження – процес маршрутизації у програмно-конфігурованої мережі.

Предмет дослідження – способи і засоби багатошляхові маршрутизації в SDN.

Наукова новизна одержаних результатів полягає у розробці нового алгоритму багатошляхової маршрутизації, що буде використовуватися протоколом маршрутизації в SDN мережі.

Публікації. За даною дисертацією було опубліковано 2 наукові роботи:

- тези доповіді для Всеукраїнської науково-практичної конференції молодих вчених та студентів «Інформаційні системи та технології управління» (ІСТУ-2018);
- подана до друку наукова стаття у науковому журналі «Східно-Європейський журнал передових технологій».

ПРОТОКОЛИ БАГАТОШЛЯХОВОЇ МАРШРУТИЗАЦІЇ, АЛГОРИТМИ МАРШРУТИЗАЦІЇ, SDN МЕРЕЖІ, МЕРЕЖІ З ТРАДИЦІЙНОЮ АРХІТЕКТУРОЮ

ABSTRACT

Master's thesis: 87 pages, 21 pictures, 26 tables, 1 appendix, 21 references.

Topicality. Every day, the number and the variety of computer network devices increases. The development of various technologies has led to the fact that the process of monitoring the network of large sizes becomes very difficult or sometimes virtually impossible. Today, computer networks consist of a large number of individual elements, where everyone is responsible for themselves.

Therefore, the scientific task is to use a new approach to network architecture such as SDN (software-defined network), with which we can optimize traffic flows and protocols.

The work communication with academic programs, plans, themes. The work was carried out at the Department of Automated Systems for Information Processing and Management of the National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute" within the objective of the work "Effective Methods for Solving the Problems of the Theory of Schedules" (№ ДП 0117U000919).

Aim of the research is to improve efficiency of routing in SDN by forming non-intersecting multi paths.

To archive the goal we need to accomplish the following **tasks**:

- perform analysis of traditional network architecture vs. architecture of software-defined network;
- develop algorithms for the routing protocol that takes into account the features of the software-defined network;
- simulate the flow of the developed algorithm;
- develop software implementation of the created routing protocol;
- make efficiency analysis of the developed algorithm.

Object of the research is the routing process in software-defined network.

Subject of the research – methods and algorithms of multipath routing in SDN.

Scientific novelty of the results is obtained by developing a new multipath routing algorithm to be used in routing protocol in the SDN network.

Publications. On the materials of the dissertation was published 2 scientific works:

- abstract of the report for Ukrainian scientific-practical conference of young scientists and students “The actual problems of informatization of management decisions” (APIMD 2018);
- article was submitted for publication in Eastern-European Journal of Enterprise Technologies.

MULTIPATH ROUTING PROTOCOL, ROUTING ALGORITHMS, SDN,
NETWORKS WITH TRADITIONAL ARCHITECTURE

ЗМІСТ

ВСТУП.....	10
1 ПРОЕКТНІ РІШЕННЯ З РОЗРОБКИ СИСТЕМИ МОДЕЛЮВАННЯ ТА АНАЛІЗУ ПРОТОКОЛІВ БАГАТОШЛЯХОВОЇ МАРШРУТИЗАЦІЇ У ПРОГРАМНО- КОНФІГУРОВАНІЙ МЕРЕЖІ.....	12
1.1 Опис бізнес – процесів.....	12
1.1.1 Архітектура мережі	12
1.1.2 Опис процесу діяльності.....	16
1.1.3 Актори і функції	17
1.1.4 Структура бізнес-процесів.....	18
1.2 Опис постановки задачі.....	19
1.3 Рішення з інформаційного забезпечення.....	19
Висновки до розділу	21
2 МОДЕЛІ ТА МЕТОДИ МАРШРУТИЗАЦІЇ	22
2.1 Змістовна постановка задачі.....	22
2.2 Математична модель.....	22
2.3 Огляд методів розв’язання	23
2.3.1 Метрика	23
2.3.2 Вимоги до сучасних протоколів маршрутизації.....	23
2.3.3 Проблеми вузлів при використанні алгоритмів маршрутизації	24
2.3.4 Алгоритми маршрутизації що використовуються в протоколах маршрутизації	24
2.3.5 Алгоритм Беллмана-Форда.....	24

2.3.6	Алгоритм Дейкстри	25
2.3.7	Алгоритм Флойда-Уоршелла	26
2.3.8	RIP протокол	26
2.3.9	OSPF протокол	28
2.3.10	OSPF протокол vs. RIP протокол	30
2.3.11	IGRP/EIGRP протокол	32
2.3.12	OSPF протокол vs. EIGRP протокол	33
2.3.13	Створення каналів зв'язку між комутатором та контролером SDN	34
2.4	Удосконалення методу розв'язання задачі	36
2.5	Розробка алгоритму розв'язання	36
2.6	Результати досліджень ефективності методу	40
	Висновки до розділу	40
3	ОПИС ПРОГРАМНОГО ТА ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ	41
3.1	Засоби розробки	41
3.2	Архітектура програмного забезпечення	42
3.2	Інструкція користувача	44
3.3	Вимоги до апаратно-програмного забезпечення	46
3.4	Використання mininet для тестування протоколу	46
	Висновки до розділу	51
4	РОЗРОБКА СТАРТАП-ПРОЕКТУ	52
4.1	Опис ідеї проекту	52
4.2	Технологічний аудит ідеї проекту	56

	9
4.3 Аналіз ринкових можливостей запуску стартап-проекту.....	57
4.4 Розроблення ринкової стратегії проекту	67
4.5 Розроблення маркетингової програми стартап-проекту	70
Висновки до розділу	73
ВИСНОВКИ.....	75
ПЕРЕЛІК ПОСИЛАНЬ	77
Додаток А Графічний матеріал	80
Структурна схема діяльності.....	81
Структурна схема варіантів використання.....	82
Схема структурна бази даних.....	83
Структурна схема класів.....	84
Блок-схема алгоритму	85
Результати експериментальних досліджень	86
Копії екранних форм	87

ВСТУП

Кожен день збільшується кількість і різноманітність пристроїв комп'ютерної мережі. Розвиток різних технологій призвели до того, що процес контролю мережі великих розмірів стає практично не контрольованим. Комп'ютерні мережі складаються з великої кількості окремих елементів, де кожен відповідає сам за себе.

Тому науковою задачею є використання нового підходу до архітектури мережі такої як SDN, за допомогою якої можна оптимізувати транспортні потоки та протоколи.

На сьогодні в мережі використовують багату кількість протоколів на різних рівнях моделі OSI. Наприклад: протокол визначення адрес, протокол тунелювання, багатопрокольна комутація за мітками, протокол керування передачею, протокол багатошляхової маршрутизації, та інше.

Темою даної роботи було обрано протоколи маршрутизації, адже за допомогою використання SDN мережі ми можемо змінити алгоритм маршрутизації, що використаний в протоколі багатошляхової маршрутизації.

Протокол маршрутизації – прокол в мережі, що використовується для визначення найкоротшого шляху між вузлами для подальшої передачі даних. Даний протокол належить до третього рівня мережевої моделі OSI. До даного рівня входять наступні задачі: трансляція логічних адрес й імен у фізичні, комутація, відстеження проблем у мережі.

Головною метою програмно-конфігурованої мережі є розділення між рівнем управління та рівнем передачі даних. Рівень управління, тобто контролер, визначає куди отриманні пакети повинні бути переслані а рівень пересилки даних фізичні відправляє ці пакети через вузли в мережі. Програмно-конфігуровано мережа пропонує використання централізованого рівня контролю для гнучкого конфігурування мережі.

SDN з самого початку стали використовувати такі компанії як Google, Facebook, Amazon та Yahoo. Традиційна архітектура мережа все частіше становиться обмежувачим фактором розвитку обчислювальних технологій. Обсяг трафіку стрімко зростає через

мобільних користувачів та Інтернету речей, використання високопродуктивних кластерів для обробки великих даних (Big Data) та використання масштабованих віртуальних середовищ для надання хмарних сервіс, змінило вимоги до архітектури мереж.

Проблема маршрутизації є часто розглядаємо, але використовуючи традиційну архітектуру мережі. Тому комбінування нової парадигми SDN та протоколів маршрутизації, може дати нові підходи до змінення сучасної архітектури мережі.

1 ПРОЕКТНІ РІШЕННЯ З РОЗРОБКИ СИСТЕМИ МОДЕЛЮВАННЯ ТА АНАЛІЗУ ПРОТОКОЛІВ БАГАТОШЛЯХОВОЇ МАРШРУТИЗАЦІЇ У ПРОГРАМНО- КОНФІГУРОВАНІЙ МЕРЕЖІ

1.1 Опис бізнес – процесів

1.1.1 Архітектура мережі

В 1983 році ARPANET почав використовувати протокол TCP/IP, і з того часу дослідники почали створювати “мережу мережей”, яка стала сучасним Інтернетом, і мала величезний приріст в 1990 році, Тим Бернерс-Лі винайшов Всесвітню мережу (WWW). [1].

Модель Інтернету можна описати як “пісочний годинник”, що з початку заснування все більше набував складності. [2] Архітектура пісочної моделі є візуальним та концептуальним представленням підходу до багаторівневих систем, які прагнуть підтримувати велику різноманітність використання та створювання. Архітектуру Інтернету можна представити на рисунку 1.1.

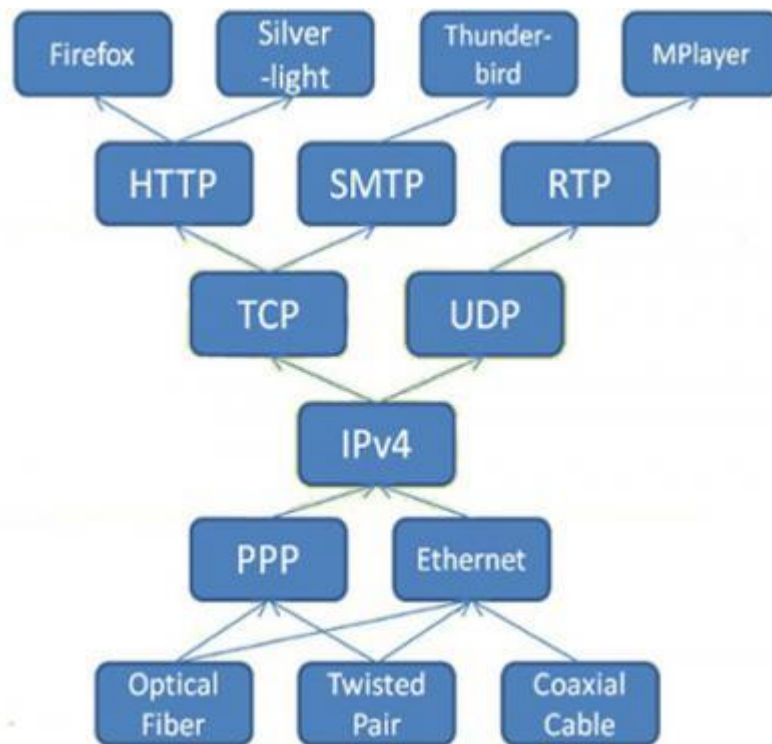


Рисунок 1.1 – Архітектура Інтернету

На самому вищому рівні створюються численні нові додатки, а на нижньому рівні протоколи що наближаються до фізичного рівня.

Змінивши цю архітектуру може спричинити непередбачувані ефекти, тому один із принципів що використовується для вирішення проблеми, це принцип end-to-end, тобто покращення тільки зверху або знизу моделі. [3]

Перелік проблем з якими зіштовхується з мережами через давню архітектуру та протоколи:

- вирішення проблем з мережами. Це завжди було досить важко, зі збільшенням складності мереж та збільшення типів мереж. На сьогодні немає інструментів які б відслідковували дефекти на всіх шарах архітектури;
- забезпечення належного використання трафіку. Забезпечення трафіку в бізнесі, особливо коли стосується WAN дуже складно, адже додатки для користувачів ускладнюються та більше просять використання у реальному режимі;

- забезпечення ресурсів для ключових програм. Операції з мережами є грою компромісів. Використання ресурсів не є централізованою;
- скорочення витрат на WAN; [4]
- швидкість інновацій. Ті компанії, що можуть швидко впроваджувати інновації мають перевагу над іншими. На жаль, на даний час мережі не вистачає гнучкості для бізнесу.

Одним з вирішення даної проблеми є використання SDN (програмно-конфігуровано мережа), а сама ця технологія буде розглянутою в даній роботі.

Головною задачею SDN є те що, адміністратор мережі може формувати трафік з централізованої консолі управління, не торкаючи окремі комутаторів у мережі. Тобто існує централізований контролер мережі, що керує комутатором для сервісів мережі, незалежно від конкретних зв'язків між сервером та пристроями.

Архітектура SDN має рівень управління мережею відділений від пристроїв передачі даних і реалізується програмно, тобто це одна з форм віртуалізації ресурсів. Прикладний рівень містить типові мережеві програми або функції (виявлення атак, firewall, розподіл завантаженості). Тобто, SDN змінює мережу тільки використовуючи додаток, а традиційна мережа буде використовувати купу спеціалізованих пристроїв. Рівень контролю являє собою централізоване програмне забезпечення контролера, яке діє як мозок програмно-визначеної мережі. Цей контролер знаходиться на сервері та керує політикою та потоком трафіку по всій мережі. Архітектура SDN представлена на рисунку 1.2.

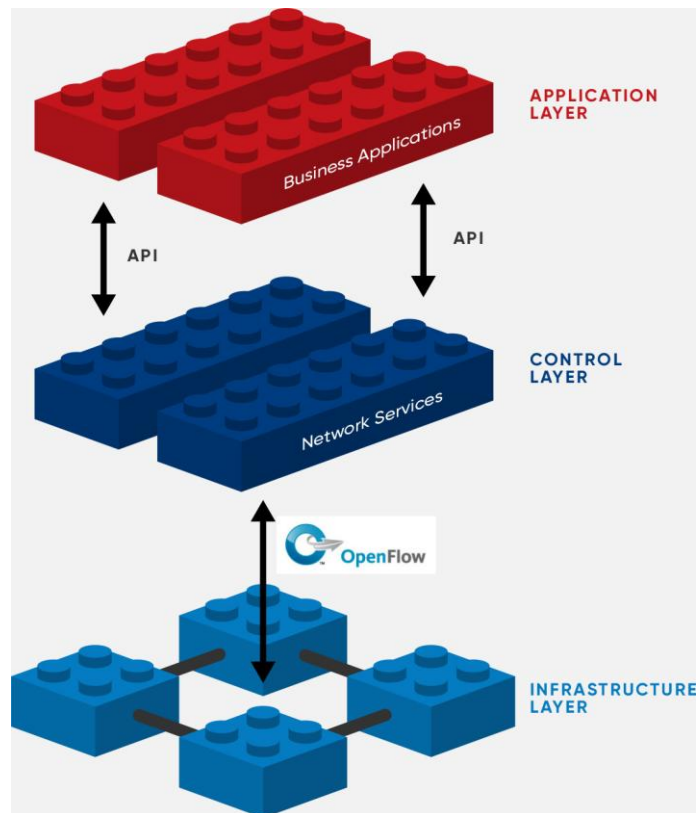


Рисунок 1.2 — Архітектура SDN

Використовуючи традиційну мережу неможливо впроваджувати зміни, що стосуються протоколів середини “пісочного годинника”. Це стосується більш застосованих алгоритмів в мережі: алгоритми маршрутизації, управління переповнення черг в маршрутизації, визначення оцінки часу затримки повідомлень, кількості відправлень пакетів.

Для даної роботи було обрано покращення алгоритму (багатошляхової) маршрутизації, адже можна використати властивість контролера про централізованість мережі, адже маршрутизатори традиційної мережі мають задачу відповідати тільки за себе, і не знати про навколишній світ.

SDN використовується в наступних галузях [5]:

- хмарні платформи. Наприклад, OpenStack покладаються на здатність мережі SDN переміщувати навантаження в різні області фізичної інфраструктури, без помітного впливу на їх продуктивність;

- 4G. Впровадження технології під назвою NFV викликало радикальне спрощення архітектури центру обробки телекомунікацій;
- OSS. Коли виявляються несправності, розподіл служб у мережі дозволяє адміністратору ізолювати, ідентифікувати та, можливо, повністю замінити всю службу через центральну консоль;
- IoT (Інтернет речей). SDN дозволить більш ефективно розподіляти ресурси мережі та вдосконалювати обслуговування на рівні SLA (Угода про рівень послуг).

1.1.2 Опис процесу діяльності

В даній роботі буде розглянуто алгоритми багашляхової маршрутизації, що будуть використовуватися для контролера мережі SDN. Маршрутизація, це процес відправлення пакетів даних між мережами та підмережами. В традиційній архітектурі мережі, пристрій що вирішує траєкторію шляху називається маршрутизатор, та використовує таблиці маршрутизації.

Централізована архітектура програмно-конфігурованої мережі дає можливість впроваджувати нові алгоритми в реальному житті та тестувати їх для різних типів мереж.

Для виконання даної роботи, використовується середовище симуляції, де буде використана архітектура програмно-конфігурованої мережі. В даному середовищі створюється різні типи топології для тестування мережі. Контролер використовує алгоритм для маршрутизації, який використовує API для контролю контролера. В подальшому для тестування змінюються метрика топології, та тестується алгоритм.

Для впровадження алгоритму в реальному житті, симуляція що використовується для тестування є абсолютно ефективною та може відповідати обставинам реальних умов.

Алгоритм маршрутизації повинен відповідати наступним вимогам:

- оптимальність вибраного маршруту відправки даних базуючи на метриці топології мережі;
- використання ресурсів не повинно перенапружити пристрій;

- стійкість – алгоритм повинен працювати при обставинах відмови зв'язків мережі;
- гнучкість – алгоритм повинен швидко адаптуватися до змінення мережі.

1.1.3 Актори і функції

Наведемо діаграму прецедентів для розробки алгоритму з використанням mininet та контролера рох на рисунку 1.3.

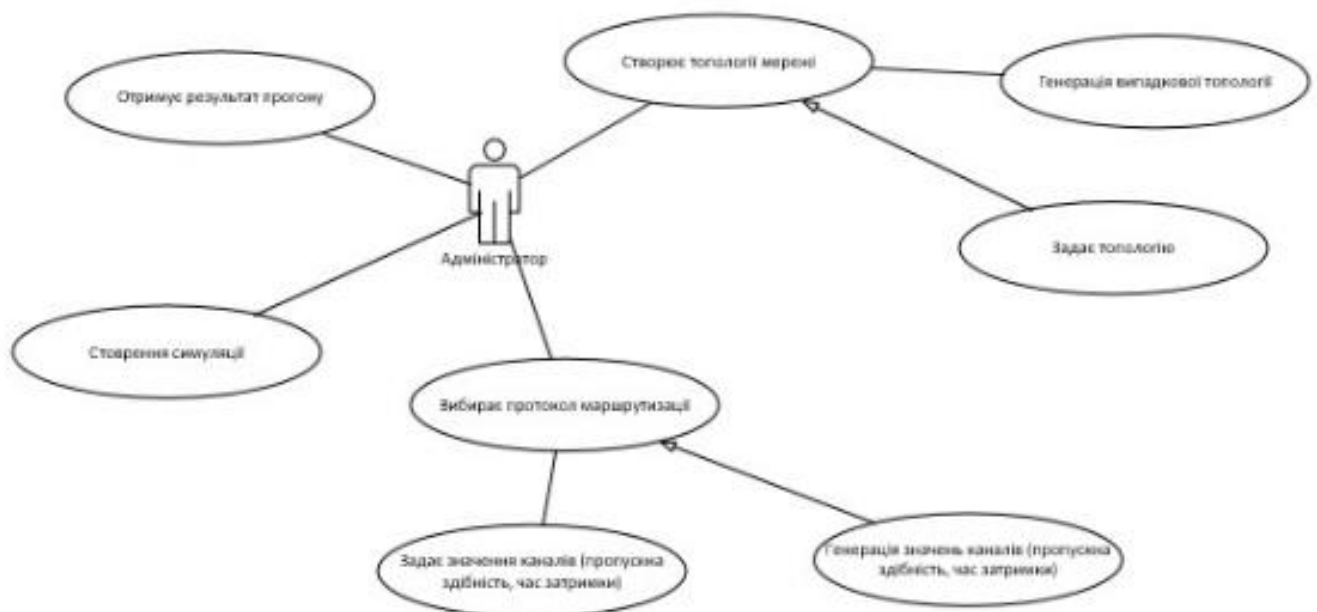


Рисунок 1.3 – Діаграма прецедентів

Акторами системи є адміністратор мережі. Тобто людина, що створює алгоритми, конфігурує середовище, запускає тестування та аналізує результати алгоритмів на різних топологіях та різних умовах мережі.

Адміністратор мережі має наступні можливості:

- створювання різних топологій мереж, з різною кількістю пристроїв, таких як, контролер SDN мережі, хости, мережеві комутатори;
- змінювати характеристики мережі, додавати можливість відмови деяких маршрутів;

- використовувати інструменти для статистики відправлень пакетів, з метою аналізу різних алгоритмів для мереж.

1.1.4 Структура бізнес-процесів

Для того щоб протестувати алгоритми маршрутизації потрібно запустити SDN контролер, створити топологію чи використати топологію за умовчанням, додати характеристики до пристроїв, запустити алгоритм та згенерувати статистику. Наведемо на рисунку 1.4 діаграму діяльності модуля тестування Mininet.

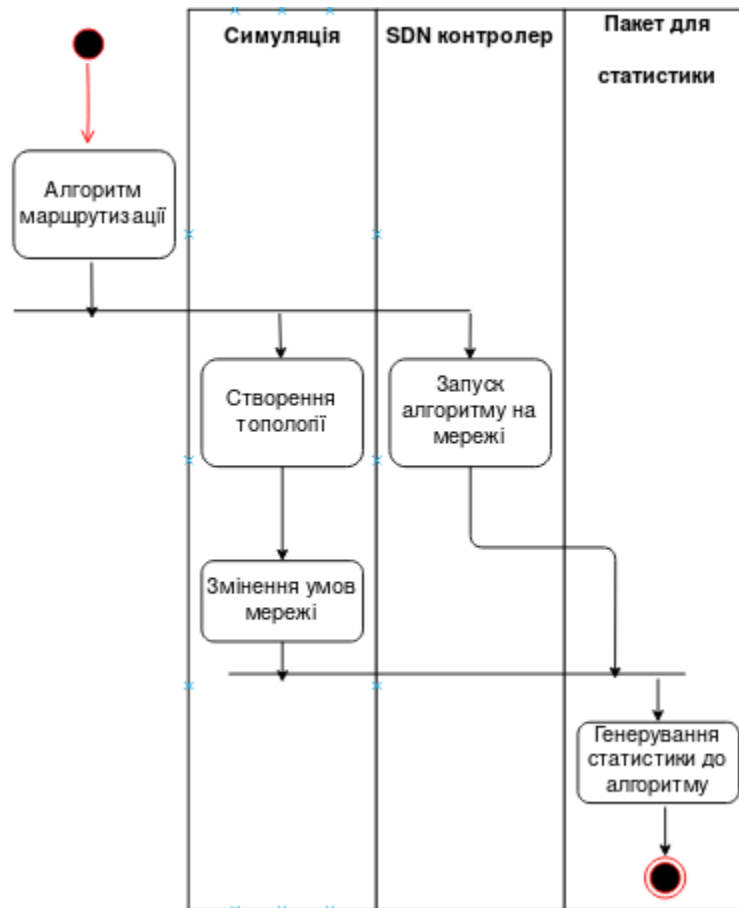


Рисунок 1.4 – Діаграма діяльності модуля тестування Mininet

Діаграма діяльності процесу моделювання маршрутизації наведена у додатку А, плакат 1.

1.2 Опис постановки задачі

Мета створення системи – це створити додатки для вже існуючої системи, що допоможе тестувати та аналізувати нові алгоритми в зручному та простому режимі.

Задачі, що було поставлені:

- аналіз алгоритмів з багашляхової маршрутизації та створення нового алгоритму, що зможе використовувати плюси SDN (розглянуто в наступному розділі);
- додавання SDN контролера до середовища тестування mininet;
- зміна алгоритму маршрутизації що використовується в контролері за замовчуванням;
- можливість змінювати параметри моделюємо мережі;
- можливість аналізувати результати алгоритму при різних умовах.

Для вирішення цих задач, було використано mininet VM з протоколом OpenFlow для створення та симуляції мережі. Було написано скріпти для створення топології та змінення умов мережі. Підключено платформу контролера для тестування мережі, та використано Wireshark для аналізу результатів.

1.3 Рішення з інформаційного забезпечення

Приклад структури JSON файлу для збереження топології з використанням mininet наведено далі.

```
{
  "test1": {
    "hosts": "h1 h2 h3",
    "switches": "s1 s2",
    "controllers": "c1",
    "links": "(h1, s1) (h2, s2) (h3, s1)",
    "metrex": {
      "Type of metrex": "Bandwidth",
      "For links": ""
    }
  }
}
```

В даному файлі для проведення тестів зберігається топологія. В таблиці 1.1 описано дані для mininet.

Таблиця 1.1 — Схема структурна бази даних для mininet

Позначення в файлі	Значення
Hosts	Вузол мережі
Switches	Мережевий комутатор
Controllers	SDN контролер
Links	Зв'язки між об'єктами в мережі

Для бази даних системи розробленої для тестування наведено ER-діаграму на рисунку 1.5.

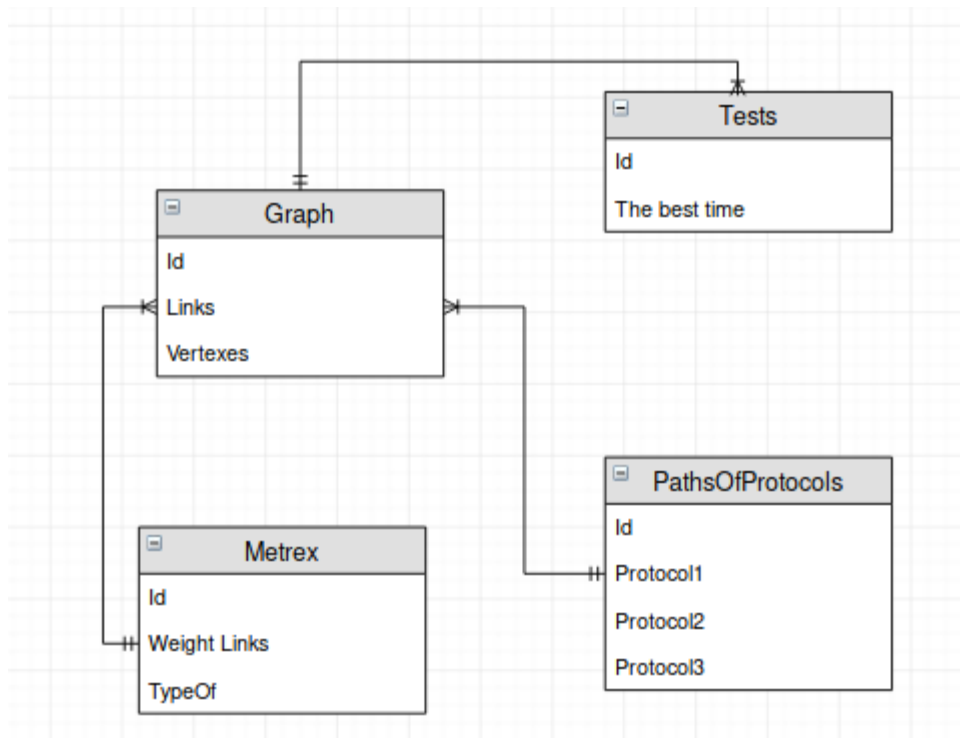


Рисунок 1.5 - ER-діаграма для системи тестування

ER-діаграма бази даних для системи моделювання протоколів маршрутизації наведена у додатку А, плакат 3. В таблицях 1.2 — 1.4 описано дані, що використовуються для системи тестування.

Таблиця 1.2 — Опис таблиці з тестами

Атрибути	Значення
Id	id
The best time	Самий швидкий час для знаходження маршруту

Таблиця 1.3 — Опис таблиці з графами

Атрибути	Значення
Id	id
Links	Зв'язки між вузлами
Vertexes	Вузли

Таблиця 1.4 — Опис таблиці з протоколами

Атрибути	Значення
Protocol1	Список маршрутів з використанням протоколу 1
Protocol2	Список маршрутів з використанням протоколу 2
Protocol3	Список маршрутів з використанням протоколу 3

Висновки до розділу

В даному розділі було розглянуто архітектури SDN та традиційної мережі. Було описано галузі використання SDN мережі та вимоги ефективного алгоритму маршрутизації. Наведено діаграму прецедентів та діаграму діяльності і описано задачі які потрібно вирішити в рамках даного дослідження.

2 МОДЕЛІ ТА МЕТОДИ МАРШРУТИЗАЦІЇ

2.1 Змістовна постановка задачі

Розробити алгоритм маршрутизації, який зробить найкоротший шлях (за метрикою) між початковим вузлом S та кінцевим вузлом F. Так як, алгоритм є багатопшляховим, то в процесі пошуку найкоротшого шляху алгоритм повинен запам'ятовувати вузли, щоб у разі відмови вузла, підібрати новий алгоритм.

2.2 Математична модель

Для розв'язання задач маршрутизації використовується орієнтований граф:

$$G = (V, L),$$

де, V – множина вершин графу;

L – множина дуг.

Дуга l_{ij} - дійсне число d_{ij} , довжина дуги в залежності від метрики протоколу маршрутизації з вузла v_i до вузла v_j .

Найкоротший маршрут $p = (l_{sk}, \dots, l_{zf})$ між вузлами v_s та v_f можна визначити наступним чином:

$$d_p = d_{sk} + d_{yt} + \dots + d_{uf} = \sum_{(i,u)} d_{fu}$$

Є три основних алгоритми пошуку найкоротшого шляху: алгоритм Беллмана-Форда, алгоритм Дейкстри та алгоритм Флойда-Уоршелла. Дані алгоритми використовуються в найпоширеніших протоколів маршрутизації, де довжина дуги визначається за метрикою кожного з протоколів.

Варто зауважити, що алгоритм Флойда-Уоршелла знаходить найкоротший шлях між кожного вузла до кожного, на відміну від інших двох, що знаходять лише найкоротший від стартового вузла до кінцевого. В останньому випадку, останні два

алгоритма можна використовувати для пошуку багатьох маршрутів, якщо запам'ятовувати вузли в процесі конструювання.

2.3 Огляд методів розв'язання

2.3.1 Метрика

Критерієм вибору того чи іншого шляху в мережі між заданою парою вузлів є мінімум або максимум вістані шляху, тобто сума довжин дуг через вузли які є на шляху. Довжина шляху в термінах протоколів маршрутизації називається метрикою протоколу маршрутизації (routing metric).

Метрика може бути обчислювальна в залежності адміністратора мережі або автоматично з розглядом характеристики мережі. Існують наступні метрики для протоколів маршрутизації:

- лічильник вузлів (hop);
- вартість;
- пропускна здатність каналу;
- затримка;
- завантаженість каналу;
- надійність.

2.3.2 Вимоги до сучасних протоколів маршрутизації

Сучасні протоколи маршрутизації повинні забезпечувати наступні характеристики мережі:

- Оптимальність найкращого маршруту в рамках певних критеріїв;
- Мінімальний обсяг створення службового трафіку;
- Масштабованість мережі, тобто здатність протоколу працювати при збільшенні або зменшенні мережі;
- Стійкість до відказів вузлів, кожен вузол може не працювати який-то час.

2.3.3 Проблеми вузлів при використанні алгоритмів маршрутизації

Чорна діра (black hole) – це вузол, де пакети трафіку відхиляються вузлом без інформування вузла, що відправив пакет. Вони можуть бути знайдені тільки моніторингом трафіку відхилених пакетів.

Петлі маршрутизації – це утворений циклічний маршрут передачі пакетів між парою вузлів без відправки його до наступного вузла.

2.3.4 Алгоритми маршрутизації що використовуються в протоколах маршрутизації

Сучасні протоколи маршрутизації використовують два типи алгоритмів:

- дистанційно-векторна маршрутизація. Дані типи протоколів використовуються наступні алгоритми для знаходження коротшого шляху: алгоритм Беллмана-Форда та алгоритм Форда-Фулькерсона;
- протоколи стану каналів. Дані типи протоколу використовують алгоритм Дейкстри.

2.3.5 Алгоритм Беллмана-Форда

Вхідними даними для алгоритму є: граф G , дистанція дуг D , та стартова вершина S . Алгоритм знаходить найкоротші шляхи від початкової вершини S до всіх наступним вершин.

Алгоритм використовує ослаблення дистанції дуги на кожній із фаз.

Вхід: матриця суміжності графу, та початкова вершина.

Вихід: найкоротший маршрут.

Крок 1. Ініціалізація всіх вершин зі значенням ∞ .

Крок 2. Для кожного вузла використати формулу Беллмана-Форда.

Крок 3. Повторювати крок 2 стільки разів скільки є вершин в графі.

Найгірша швидкодія: $O(|V|^3)$,

де, $|V|$ - кількість вузлів..

Даний алгоритм використовується в протоколів дистанційно-векторної маршрутизації, наприклад в RIP протокол. Даний алгоритм є одним з найрозповсюджених алгоритмів маршрутизації, що використовує хопи як метрику.

Обмеженням протоколу, що використовує даний протокол є мала кількість дозволених хопів (всього 15), тому даний алгоритм використовується в мережах маленької ємності.

2.3.6 Алгоритм Дейкстри

Вхідними даними для алгоритму є: граф G , дистанція дуг D , та стартова вершина S . Алгоритм знаходить найкоротші шляхи від початкової вершини S до кінцевої вершини F .

Вхід: матриця суміжності графу, та початкова вершина.

Вихід: найкоротший маршрут.

Крок 1. Ініціалізація всіх вершин зі значенням ∞ . Ініціалізація початкової вершини зі значенням 0.

Крок 2. Помітити початкову вершину як пройдену, всі інші вершини як не пройдені.

Крок 3. Обрати початкову вершину.

Крок 4. Розрахувати тимчасові відстань всіх сусідніх вузлів обраної вершини, підсумовуючи відстань використовуючи вартість ребра.

Крок 5. Якщо розрахована відстань вузла менша за поточне значення, оновить дистанцію.

Крок 6. Обрати вершину з мінімальною дистанцією, помітити цю вершину.

Крок 7. Повторювати крок з 4 до 7 поки не залишиться не помічені вузли.

Найгірша швидкодія: $O(|L| + |V^2|) = O(|V^2|)$.

Де, $|V|$ - кількість вузлів, $|L|$ - кількість дуг.

Даний алгоритм використовується для протоколів стану каналів маршрутизації, наприклад протокол OSPF та IS-IS. Даний протокол дає можливість вибрати тип метрики в залежності від типу послуги.

2.3.7 Алгоритм Флойда-Уоршелла

Вхідними даними для алгоритму є: граф G , дистанція дуг D , та стартова вершина S та кінцева вершина F .

Вхід: матриця суміжності графу, та початкова вершина.

Вихід: масив маршрутів.

Крок 1. Ініціалізація всіх дистанцій зі значенням ∞ . Ініціалізація вершин зі значенням null.

Крок 2. Для кожної вершини k від 1 до $|V|$.

Крок 2.1. Для кожної вершини i від 1 до $|V|$.

Крок 2.2. Для кожної вершини j від 1 до $|V|$, обчислити за формулою 2.1.

В цьому алгоритмі використовується попарне відношення для знаходження коротшого шляху за наступною формулою:

$$G_{ij}^n = \min(D^n - 1_{ik}, D^n - 1_{ik} + D^n - 1_{kj}) \quad (2.1)$$

Найгірша швидкодія: $O(|V|^3)$,

де, $|V|$ - кількість вузлів.

Даний алгоритм використовується в протоколів дистанційно-векторної маршрутизації, наприклад в RIP протоколі. Найбільше використовується з мережами, що належать ISP.

2.3.8 RIP протокол

RIP — це протокол, що використовує вектор відстаней (DV). Даний протокол використовується тільки в мережах не великої розмірності, адже має обмеження числа

пересилань (hops). Для пошуку шляху використовується алгоритм Беллмана — Форда. [9]

RIP протокол (Routing Information Protocol) версії 2 (v2) використовує групову передачу пакетів для обміну інформації маршрутизації.

RIP протокол має наступні характеристики:

Обмеження протоколу. Даний протокол використовує кількість пересилань як метрику. Кількість пересилань (hop count), це кількість вузлів які конструюють маршрут. Два вузли що зв'язані напряду мають метрику 0, а вузли що не зв'язані метрику 16. Тому, даний протокол не є ефективним для використання у великих мережах.

Обновлення маршрутів. RIP протокол відправляє інформацію про маршрутизацію кожні 30 секунд або коли змінюється топологія. Якщо вузол не передає жодну інформацію за 180 секунд даний вузол помічається як не використаний. Метрика збільшується на 1 при кожному вузлі, та вузол записується як наступний. Цей протокол зберігає тільки оптимальний маршрут (тобто з меншою кількість переходів). Якщо в одному з вузлів була відновлена інформація в таблицях маршрутизації, вузол починає відправляти оновлені RIP таблиці для інформування інших вузлів.

Версії протоколу. На сьогодні використовують дві версії даного протоколу:

- RIPv1, який задається стандартом RFC 1058 [13];
- RIPv2, який задається стандартом RFC 2453 [9].

Протокол версії 1, використовує ширококомвну передачу пакетів.

Протокол версії 2, використовує групову передачу пакетів, має функцію автентифікація (для підтвердження що пакети надіслані з авторизованих вузлів), підтримує VLSM (Variable Length Subnet Masking – для рекурсивного ділення порції адресного простору), та має за замовчуванням метод автентифікація як простий текст (що робить його вразливим).

Метрика. RIP протокол використовує лише кількість переходів. Для кожного вузла є значення переходу від джерела до призначеного вузла.

Значення протоколу. В даному протоколі для покращення на різних мережах можна змінювати наступні значення:

- час за яким відправляється оновлення;
- час у секундах, після якого вузол є не працюючим;
- кількість часу у секундах, після якого вузол видаляється з таблиці маршрутизації.

2.3.9 OSPF протокол

OSPF – належить до класу протоколів стану каналів (Link State Protocol), визначає значення метрики від типу послуги TOS (Type of Service). Це найвідоміший протокол серед протоколів внутрішньої маршрутизації створений в 1980 році робочою групою IETF.

В OSPF підтримуються метрики пропускну здатності та затримки. Метрика, що оцінює пропускну здатність каналу, визначається, наприклад, як кількість секунд, необхідних для передачі 100 Мбіт. Метрика затримки — час у мікросекундах, необхідний маршрутизатору для обробки, постановки в чергу та передачі пакетів. Для кожної з метрик протокол OSPF будує окрему таблицю маршрутизації. Стандартний порядок розрахунку метрики, що оцінює показники надійності, затримки й вартості, поки не визначений. Цей порядок визначається адміністратором. [5][6]

Протокол OSPF слухає сусідів та збирає всі дані про стан зв'язку, щоб створити карту топології усіх доступних шляхів у своїй мережі, а потім зберігає інформацію у базі даних стану каналів LSBD (Link-State Database). [7]

Даний протокол створює три таблиці:

- таблиця сусідів – містить всіх знайдених сусідів з якими були обміняні повідомлення;
- таблиця топології – містить всі доступні маршрути та враховує найкращі та альтернативні маршрути;

- таблиця маршрутизації – містить найкращі маршрути, що будуть використовуватися для обміну трафіку між вузлами.

OSPF протокол має наступні характеристики:

Обмеження протоколу. Даний протокол через використання стану каналів погано адаптується до збільшення топології. З використання більшої кількості вузлів, збільшується частота оновлення таблиць маршрутизації та кількість часу для обчислення маршруту. Даний факт створює великі таблиці маршрутизації, використовуючи багато пам'яті, що впливає на час роботи протоколу.

Переваги. Протокол OSPF має наступні переваги:

- масштабованість, цей протокол створений для великих топологій;
- підтримка підмережі;
- hello пакети, використовуються для верифікації каналу без відправлення всієї таблиці маршрутизації;
- ToS маршрутизації, що дозволяє даному протоколу використовувати різні критерії в залежності від типу сервісу (ToS).

Версії протоколу. На сьогодні використовують три версії даного протоколу:

- OSPFv1, який задається стандартом RFC 1131 [14];
- OSPFv2, задається стандартом RFC 1245 [15];
- OSPFv3, задається стандартом RFC 5340 [16].

Версія 3 OSPF протоколу вводить використання IPv6 через високу кількість приладів підключених до Інтернету. Також дана версія використовує VLSM (підтримка масок змінної довжини), автентифікації оновлення маршрутів, зменшення кількості таблиць маршрутизації (route summarization).

OSPF зона. Даний протокол для створення таблиць маршрутизації використовує оновлення стану каналів між вузлами. Кожне оновлення чи зміни топології відправляється кожному вузлу. Тому, для цього протоколу використовуються зони вузлів, тобто для оновлення стану каналів, оновлюється тільки група вузлів що є в даній зоні.

Звісно є ті вузли що належать до кількох зон, тому вони отримують оновлення для всіх зон та називаються ABR (area border routers). Приклад зон OSPF представлений на рисунку 2.1.

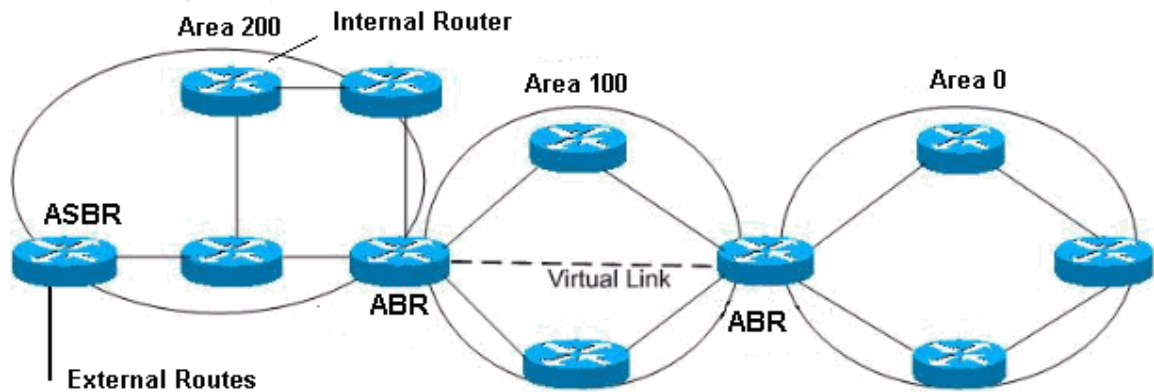


Рисунок 2.1 – Використання зон вузлів

Метрика. Для даного протоколу використовується пропускна здатність каналу та час затримки. Пропускна здатність вимірюється як кількість часу для передачі 100 Мбіт даних. Час затримки вимірюється як кількість часу для передачі 1 біту даних від одного вузла до іншого, та може змінюватися залежно від:

- час для обробки пакету маршрутизатором;
- час прибуття пакету в черзі;
- час для відправки через канал;
- час відповіді про отриманий пакет.

2.3.10 OSPF протокол vs. RIP протокол

Наведемо таблицю з порівнянням OSPF та RIP протоколів (табл. 2.1):

Таблиця 2.1 – Порівняльний аналіз протоколів OSPF та RIP

Характеристика	RIP протокол	OSPF протокол
Обмежена кількість переходів вузлів	Обмеження: 15 переходів (hops count).	∞

Продовження таблиці 2.1

Підтримка VLSM (надає можливість ефективного присвоєння IP-адрес, маючи на увазі обмежену кількість адресів)	RIPv1 не підтримує (але в багатьох вузлах використовується за замовченням), RIPv2 підтримує	Підтримується
Широкомовна передача пакетів (є недоліком для великої топології)	Використовується RIPv1, в RIPv2 вже групова передача	Використовуються hello пакети та групова передача пакетів
Однорідна мережа (є недоліком для швидкості протоколу)	RIP мережі є однорідними	OSPF протокол використовує граничні мережі, що дозволяє використання області мережі
Метрика	Використовує тільки лічильник переходів (з можливістю обмежень таймерів)	Використовує різні типи: пропускна здатність та затримка каналу
Автентифікація	За замовчуванням простий текст, RIPv2 дозволяє використовувати MD5	Простий текст, MD5, key rollover
Складність конфігурації протоколу	Легкий	Складний

2.3.11 IGRP/EIGRP протокол

Протокол IGRP (Interior Gateway Routing Protocol) і його розширена версія EIGRP (Enhanced Interior Gateway Routing Protocol) розроблені в корпорації Cisco. І якщо протокол IGRP — це дистанційно-векторний протокол, то EIGRP — протокол змішаного типу [6].

Балансування навантаження в протоколі IGRP може здійснюватися на шляхах з однаковою та різною вартістю. Сам трафік розподіляється по різних шляхах пропорційно їх вартості. Балансування навантаження на шляхах з однаковою вартістю здійснюється автоматично, а для використання шляхів з різною вартістю необхідна додаткова конфігурація та налаштування обладнання [8, 6].

Протокол для визначення найкоротшого шляху використовує алгоритм Дейкстри. Для обчислення вартості ребра в графі використовується наступна формула:

$$\left(\frac{\left(K_1 * Bandwidth + \frac{K_2 * Bandwidth}{256 - Load} + K_3 * Delay \right) * K_5}{K_4 + Reliability} \right) * 256$$

де ($K_1 - K_5$) змінні, що визначаються адміністратором (за замовчуванням = 1);

Bandwidth– пропускна здатність каналу;

Load– завантаження каналу;

Delay– затримка;

Reliability– надійність.

Опишемо характеристики EIGRP протоколу.

Переваги. Протокол EIGRP має наступні переваги:

- висока швидкість конвергенції, тому що кожен вузол зберігає таблицю маршрутизації для сусідніх вузлів (вузли, що з'єднанні вузлом);
- підтримує VLSM;

- підтримує не повне оновлення для випадків коли змінюється метрика маршрутизації;
- підтримує кілька протоколів мережевого рівня.

Метрика. Протокол EIGRP використовує наступну метрику:

- затримка каналу (мкс);
- надійність каналу (від 1 до 255, де 1 – найменш надійний, по кількості загублених пакетів);
- завантаження каналу (від 1 до 255, де 1 – найменш завантажений, по завантаженості каналу).

Обмеження протоколу. Даний протокол не є пропрієтарне, але використовується тільки з апаратним забезпеченням компанії Cisco [17].

Відмінність алгоритму. Даний протокол є дистанційно-векторним, але на відміну від інших таких протоколів, що використовує алгоритм DUAL (дифузійний алгоритм оновлення), а не алгоритм Форда-Фулькерсона чи Бельмана-Форда.

Таблиці, що використовуються для маршрутизації. Даний протокол використовує наступні таблиці маршрутизації:

- таблиця сусідів містить інформацію (інтерфейс мережі, адреса, лічильник переходів) щодо сусідів вершини;
- таблиця топології містить метрику маршрутів до кожної вершини в AS;
- таблиця маршрутизації містить найкоротших шлях по метриці.

2.3.12 OSPF протокол vs. EIGRP протокол

Наведемо таблицю 2.2 з порівнянням OSPF та EIGRP протоколів:

Таблиця 2.2 – Порівняльний аналіз протоколів OSPF та EIGRP

	OSPF	EIGRP
Тип протоколу	Протокол стану каналів	Протокол дистанційно векторної маршрутизації

Продовження таблиці 2.2

Відмова вузла, що є в маршруті	Протокол запускає алгоритм стану каналів на базі LSA вузлів для знаходження найкращого маршруту	Використовує дифузійний алгоритм оновлення (DUAL)
Ліцензія	В минулому пропріетарне ПО Cisco	Open Standard
Зменшення кількості таблиць маршрутизації (routing summarization)	Можемо тільки в ABR та ASBR	Будь де в топології
Відключення функції routing summarization	Ні	Так
Максимальна кількість вузлів	256 (hop count від 0 до 255)	500
Метрика за замовчуванням	Тільки пропускна здатність каналу	Пропускна здатність каналу та затримка

2.3.13 Створення каналів зв'язку між комутатором та контролером SDN

Під час фази ініціалізації комутатор та контролер обмінюються OpenFlow повідомленнями. Повідомлення можуть бути типу:

- SM – симетричне повідомлення між комутатором та контролером;
- CSM – асиметричне повідомлення між комутатором та контролером.

Комутатор використовує “Hello (SM)” пакети для ідентифікації контролера та використовує пакети для погодження використання версії OpenFlow протоколу для комунікації (рис.2.2).

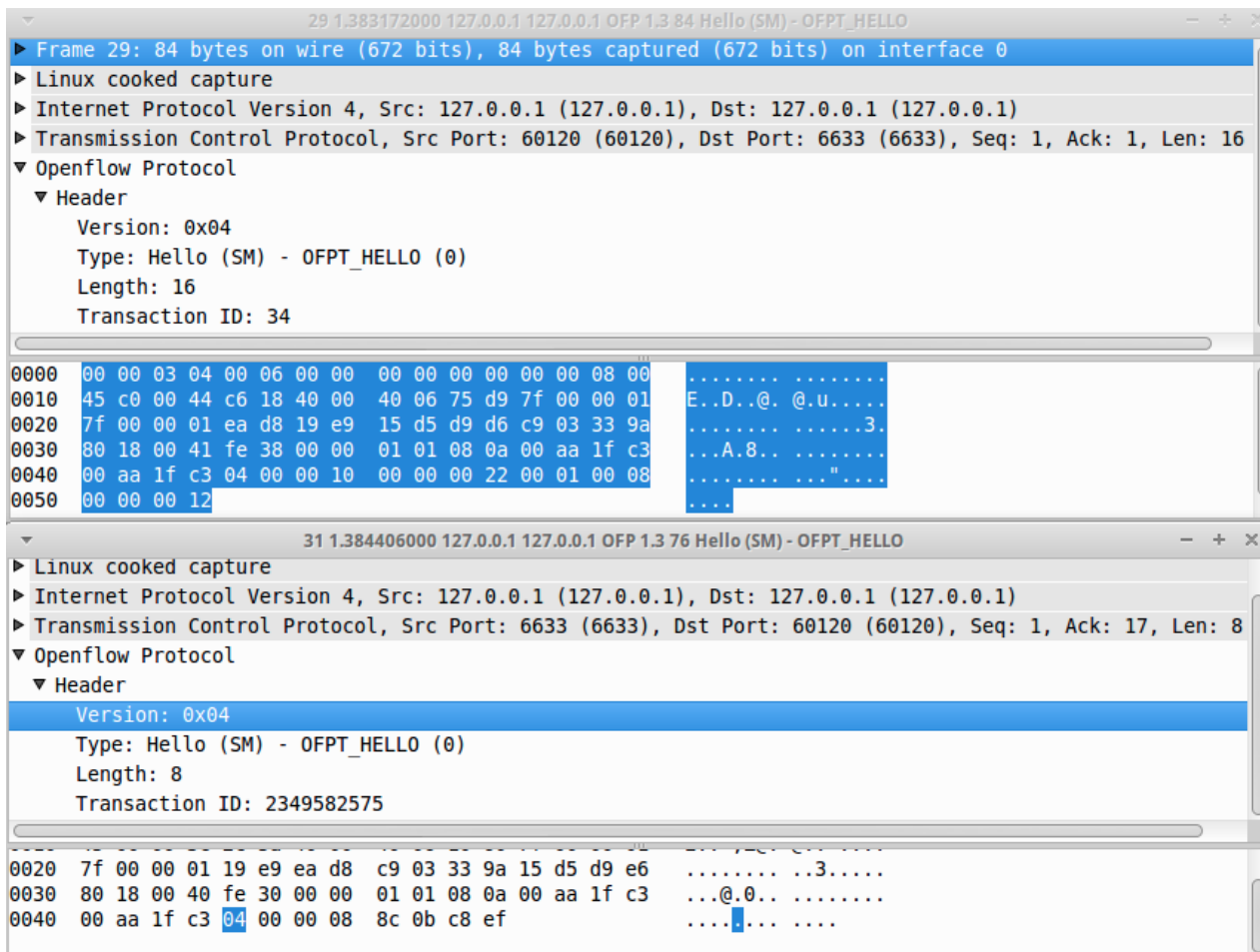


Рисунок 2.2 – Заголовок пакету “Hello (SM)” в WireShark

Процес обміну повідомленнями можна представити як [10]:

Крок 1. Контролер відправляє CSM запит для виявлення вільних портів;

Крок 2. контролер відправляє комутатору запит з параметрами “set config”;

Крок 3. комутатор відправляє контролеру повідомлення (feature reply CSM) з ідентифікацією фізичних портів, пропускної здатності, об’єм буферу та інше.

Повідомлення якими обмінюються комутатор та контролер мають ID транзакції, вони повторюються для надсилання відповіді та запиту. Цей процес повторюється кожні 15 секунд. Всі потоки мають за замовчуванням значення 60 секунд (idle-timeout), тобто якщо нема нового трафіку OpenFlow протокол видаляє потік та інформує про це контролер.

2.4 Удосконалення методу розв'язання задачі

Для підходу даного вирішування задачі, було обрано використовувати SDN, адже це зручно в тестуванні, та моделюванні. Також характеристикою SDN є централізованість мережі, є головний контролер мережі, який має інформацію про вузли. В мережі традиційної архітектури кожен маршрутизатор вирішує маршрутизацію тільки для себе.

Для вирішення задачі маршрутизації можна, маючи контролер, почати пошук одночасно з двох вузлів.

Для розв'язання задач маршрутизації використовується орієнтований граф:

$$G = (V, L)$$

де V – множина вершин графу;

L – множина дуг.

Потрібно знайти найкоротший шлях від вершини S до вершини F . Так як, використовується контролер можна почати одночасно пошук маршруту з вершини S та вершини F . Побудова маршруту буде виконуватися алгоритмом Флойда-Уоршелла, а саме формулу Ч. Коли два маршрути зіткнуться буде знайдено, та створено маршрут від вершини S до вершини F , алгоритм знайшов оптимальний маршрут.

2.5 Розробка алгоритму розв'язання

Даний алгоритм буду розроблятися для платформи розробки на Python, що базується на програмно-конфігурований мережі, з протоколом OpenFlow.

Для нового протоколу маршрутизації буде використовуватися ті метрики для визначення вартості ребра, що і розглянутих вище протоколів, але з іншим алгоритмом:

- а) модифікація протоколу RIP:
 - 1) кількість переходів (hop count);
- б) модифікація протоколу OSPF:
 - 1) пропускна здатність каналу (дуги);

в) модифікація протоколу IEGRP:

- 1) пропускна здатність каналу;
- 2) затримка каналу.

Тому контролер матиме топологію у вигляді наступного графу:

$$G = (V, E, B, T, H),$$

де $V = \{v_1, v_2, \dots, v_n\}$, – множина вершин;

$E = \{e_1, e_2, \dots, e_n\}$ – множина дуг графу;

$B = \{b_1, b_2, \dots, b_n\}$ – множина значень пропускної здатності вершин;

$T = \{t_1, t_2, \dots, t_n\}$ – множина затримки каналів;

$H = \{h_1, h_2, \dots, h_n\}$ – множина кількості переходів.

Так як, розробляється протокол для SDN мережі, де контролер має повну інформацію про вузли, запропоновано починати алгоритм з обох вузлів S (вузол з якого відправляють пакет), F (вузол куди пакет повинен прийти) одночасно [19].

Запропонований алгоритм наведено нижче.

Крок 1. Ініціалізація. Ініціалізація масиву шляхів зі стартової вершини: ребра що починаються з вершини S до сусідів. Відстань до всіх інших вершин графа ∞ . Ініціалізація масиву шляхів з фінальної вершини: ребра що починаються з вершини F . Відстань до всіх інших вершин графа ∞ .

Крок 2. Знаходимо таку вершину (із ще не оброблених), де поточна найкоротша відстань є мінімальною починаючи з вершини S . Та запам'ятовуємо масив вершин з відстанню, до всіх сусідніх вершин. Теж саме, починаючи з вершини F .

Крок 2.1. Обходимо всіх сусідів поточної вершин, і якщо шлях в сусідню вершину через іншу вершину менше за поточний для вершини, запам'ятовуємо новий коротший шлях до сусіда. Зробимо це для всіх вершин що є в масиву вершин найкоротших шляхів. Теж саме, для вершини F .

Крок 3. Порівнюємо поточні вершини з кроків 2 та 2.1. Якщо вершини не співпадають повторюємо кроки починаючи з 2. Якщо вершини співпадають крок 4.

Крок 4. Створюється кінцевий масив найкоротших маршрутів об'єднуючи маршрути побудовані з вершини S та F.

Блок-схема алгоритму наведена на рисунку 2.4.

На початку алгоритму маємо топологію з 9 комутаторами, 12 хостами та одним SDN контролером який наведено на рисунку 2.3.

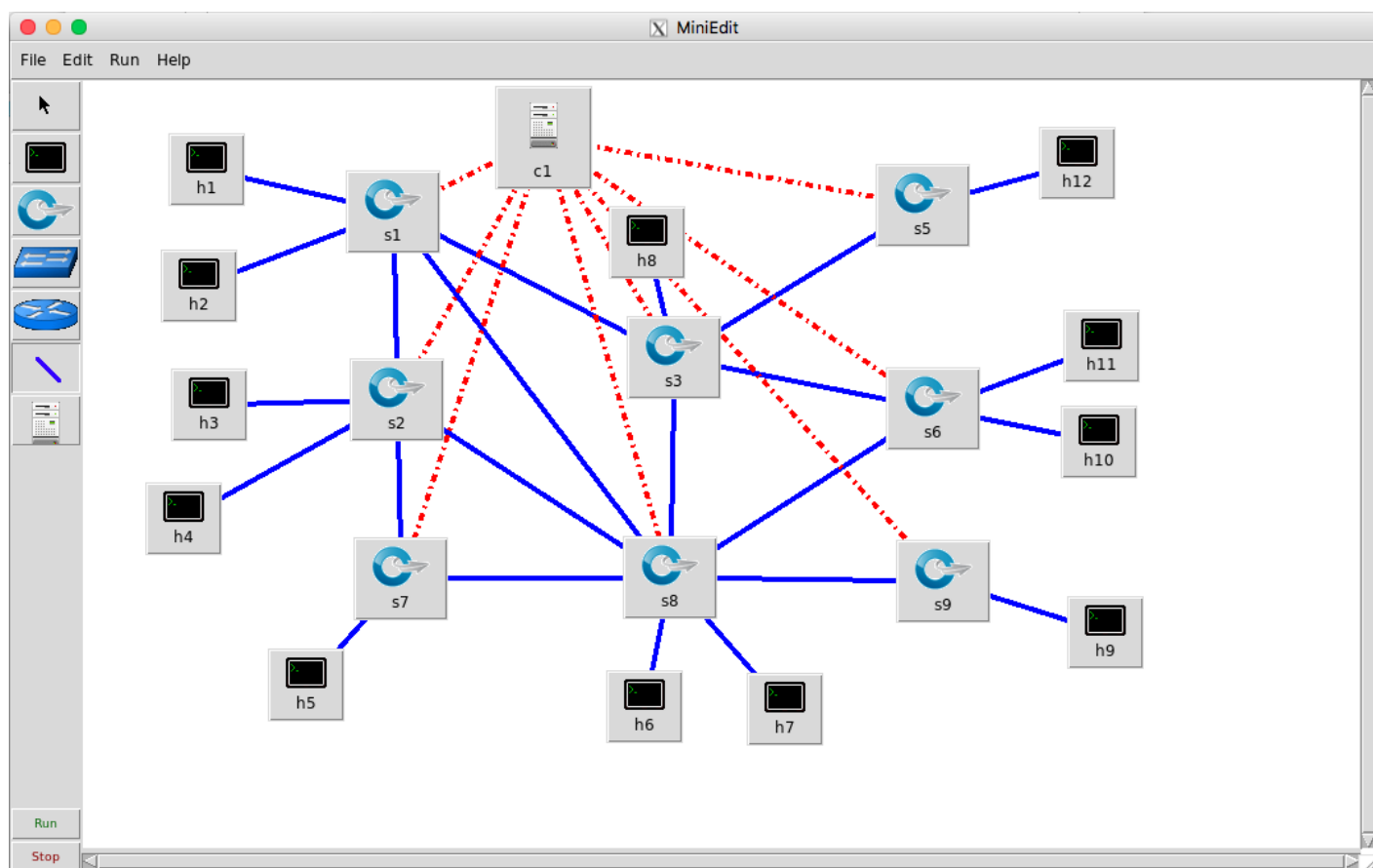


Рисунок 2.3 – Початкова топологія мережі (MinEdit)

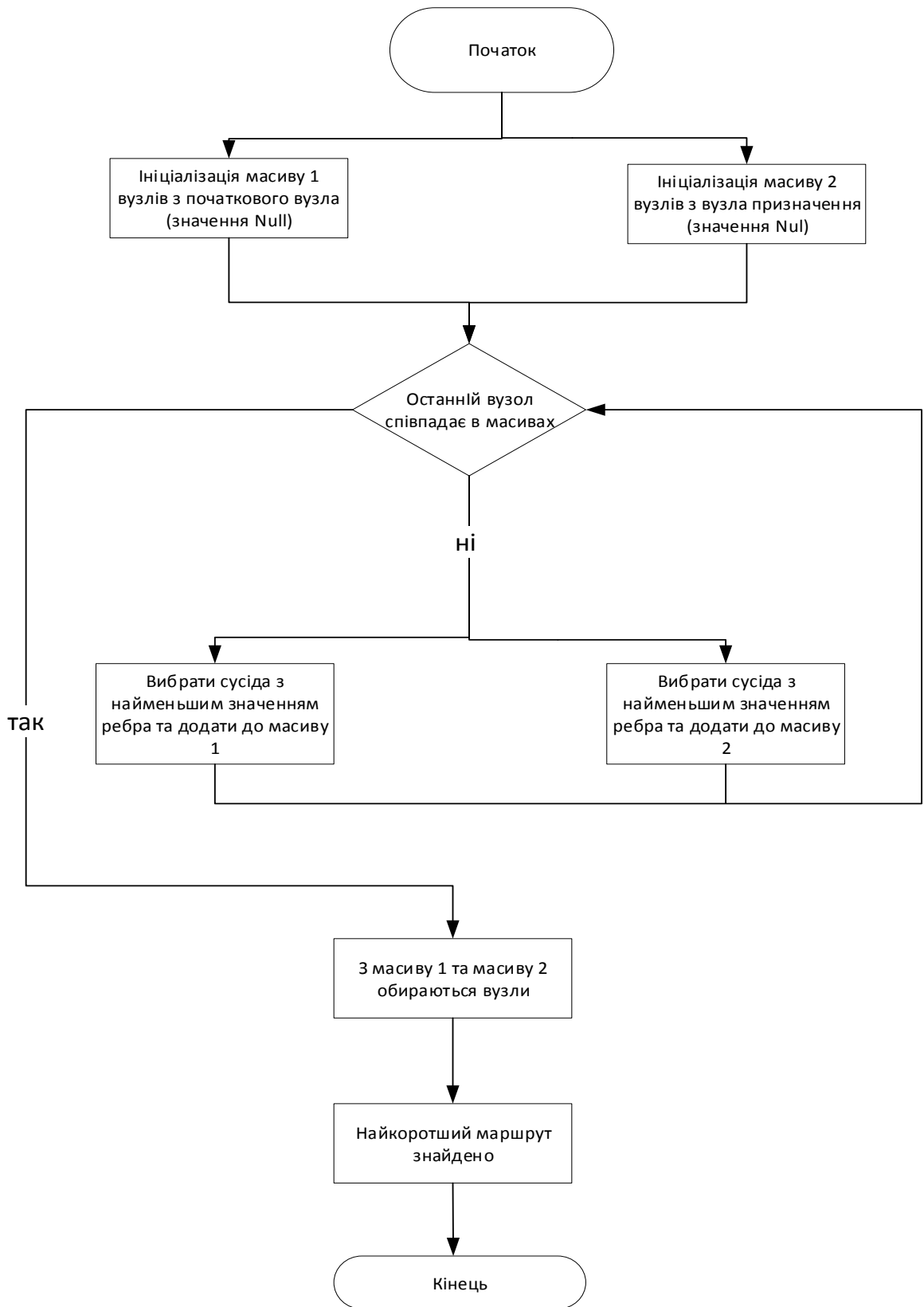


Рисунок 2.4 – Блок-схема алгоритму

2.6 Результати досліджень ефективності методу

Для дослідження ефективності алгоритму буде використовуватися аналіз тестів. Тобто для кількох топологій буде запущено різні алгоритми маршрутизації. Для різних топологій буде змінено умови тестів (таких як, збільшення/зменшення пропускної здатності, відмова деяких каналів зв'язку, максимальна кількість переходів для пристрою). Результати тестів буде занесено до таблиці та проаналізовано.

Висновки до розділу

В цьому розділі було зроблено огляд протоколів маршрутизації та протоколів багатопляхової маршрутизації, алгоритмів що використовуються в протоколах, типи метрик. Було також обґрунтовано використання саме SDN архітектури.

Для кожного алгоритму та протоколу було наведено недоліки, обмеження та відмінності всіх протоколів. Був зроблений порівняльний аналіз між парами протоколів RIP, OSPF та OSPF, EIGRP.

Основні проблеми, що характеризуються описані протоколи є:

- багато протоколів маршрутизації були створені для реалій традиційної архітектури мережі;
- кожен протокол є найбільш ефективний при специфічних типів мереж (наприклад, RIP протокол підходить для малих топологій, де кількість переходів менше 16, OSPF протокол є відкритим стандартом і може використовуватися з різним апаратний забезпеченням);
- деякі компанії створюють пропріетарні протоколи для контролю ринку апаратного забезпечення;
- використання метрик протоколами маршрутизації не завжди є ефективним;
- конфігурація виконується мануально в разі використання традиційної архітектури.

За матеріалами дисертації було опубліковано дві наукові роботи: одна стаття та одні тези доповіді на конференціях [19, 20].

3 ОПИС ПРОГРАМНОГО ТА ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Засоби розробки

Використовуються наступні інструменти для тестування з використання mininet, рох:

Mininet VM (Ubuntu). Дана віртуальна машина допомагає в створенні реалістичної віртуальної мережі. Mininet можна використовувати як консоль, через API або як віртуальну машину. Було обрано використовувати саме віртуальну машину адже вона має різні фреймворки контролерів. Його можна представити як на рисунку 3.1.

OpenFlow протокол забезпечує відкритий інтерфейс для програмно-конфігурованих мереж [12].

Рох, це SDN контролер, що використовується в симуляторі mininet. Ця для цього контролера будуть зроблені алгоритми маршрутизації для даної роботи. Рох контролер на мові Python 2.7.

WireShark, один з найкращих аналізаторів протоколів мереж. За допомогою даного інструменту можна досліджувати протоколи, перехватувати їх в реальному режимі, фільтрувати, зберігати результати, аналізувати різні інтерфейси мережі.

MiniEdit, mininet GUI. За допомогою цієї програми можна візуально будувати топології мережі.

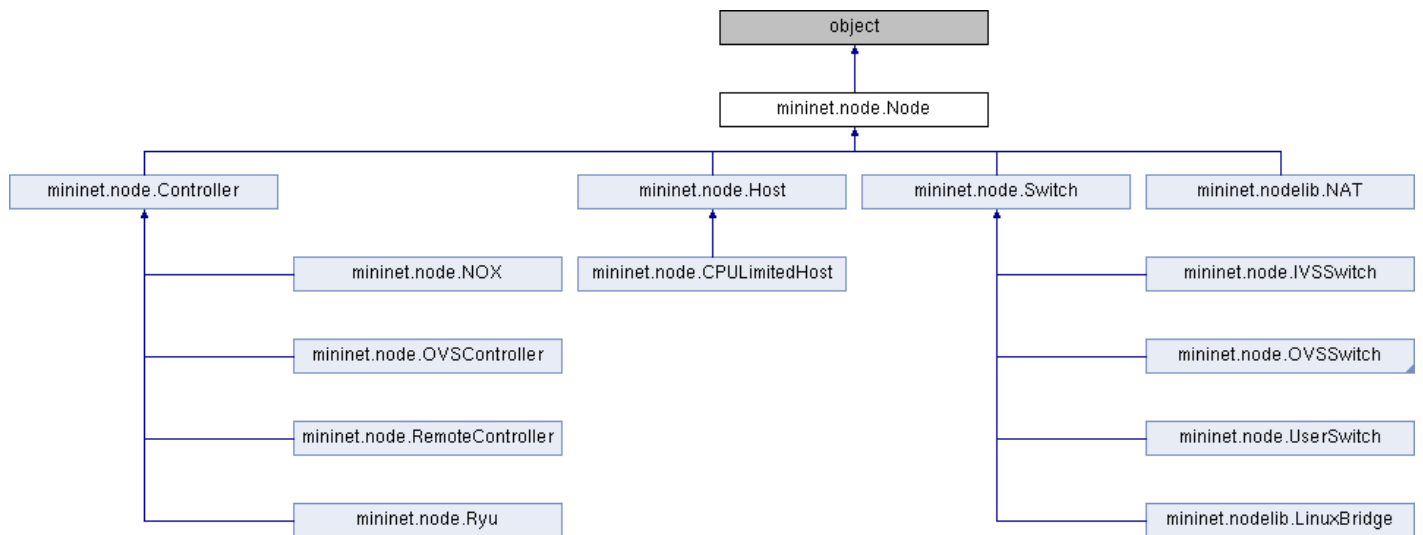


Рисунок 3.1 — Архітектура mininet

Для системи що розроблена для тестування використовується мова Python з модулями:

- Networkx – пакет open-source розроблений на Python для генерації графів, синтетичних мереж;
- WxPython – крос-платформений open-souce набір інструментів для створення GUI;
- SimPy – фреймворк open-source симуляції для Python.

3.2 Архітектура програмного забезпечення

Діаграма класів для системи тестування представлена на рисунку 3.2, діаграма класів для системи моделювання протоколів маршрутизації наведена у додатку А, плакат 4.

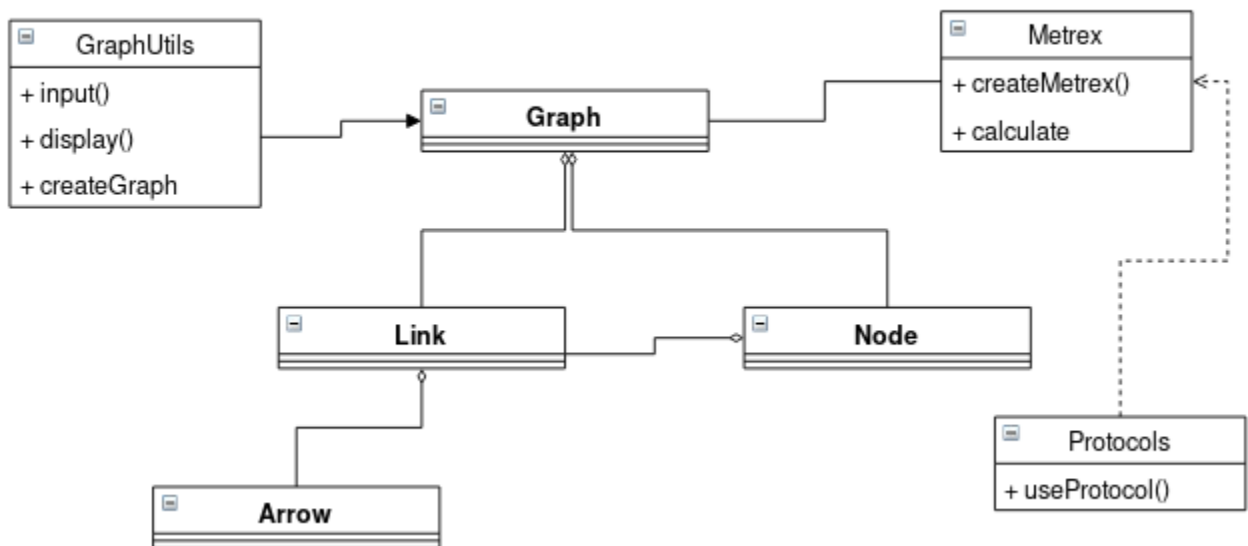


Рисунок 3.2. - Діаграма класів для системи тестування

Використовуються наступні класи:

- а) graphUtils – клас для роботи з графом. Має наступні методи:
 - 1) input() – введення даних для графу;
 - 2) display() – виведення та його на екрані;
 - 3) createGraph() – створення графу;
- б) graph – клас для створення графу (алгоритм його на екран);
- в) link – клас для створення дуг між вузлами;
- г) node – клас для створення вузлів для графу;
- д) metrex – клас для використання метрики на графі. Має наступні методи:
 - 1) createMetrex() – вибір та встановлення тип метрики;
 - 2) calculate() – розрахування ваги дуг з використанням типів метрик;
- е) protocols – клас для використання протоколів на графі. Має наступні методи:
 - 1) useProtocol() – використання алгоритму для протоколу для знаходження маршрутів тесту з використанням ваг, що були розраховані класом Metrex.

3.2 Інструкція користувача

Система що створюється для тестування протоколів надає наступні можливості:

- створити топологію мережі за допомогою даних що ввів користувач;
- вказати тип метрики, що користувач бажає протестувати;
- вказати тип алгоритму, що користувач бажає використовувати;
- запустити тестування, побачити як протокол працює в реальному режимі та отримати результати.
- Для аналізу протоколу багатошляхової маршрутизації за допомогою програми симуляції мережі та контролера на базі Python – рох:
 - створити топологію мережі;
 - обрати та запустити один з протоколів;
 - аналізувати результат за допомогою WireShark.

В розробленій системі для симуляції потрібно створити файл з описаними вершинами, ребрами, пропускної здатності та затримці. Тоді, запустити симуляції що покаже результат та кроки протоколу.

На рисунку 3.3 наведено побудована топологія.

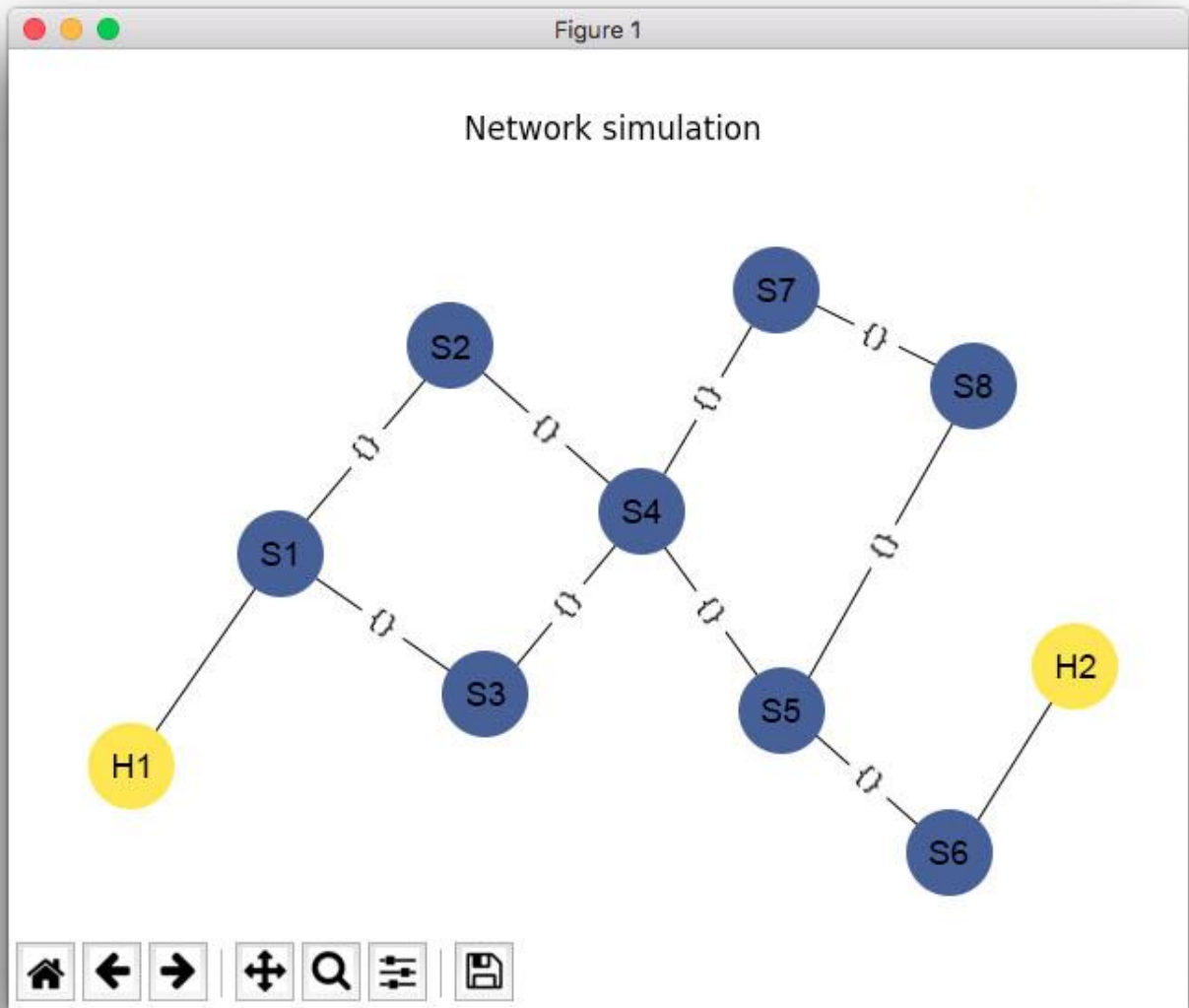


Рисунок 3.3 – побудована топологія

На рисунку 3.4 наведено кінець симуляції.

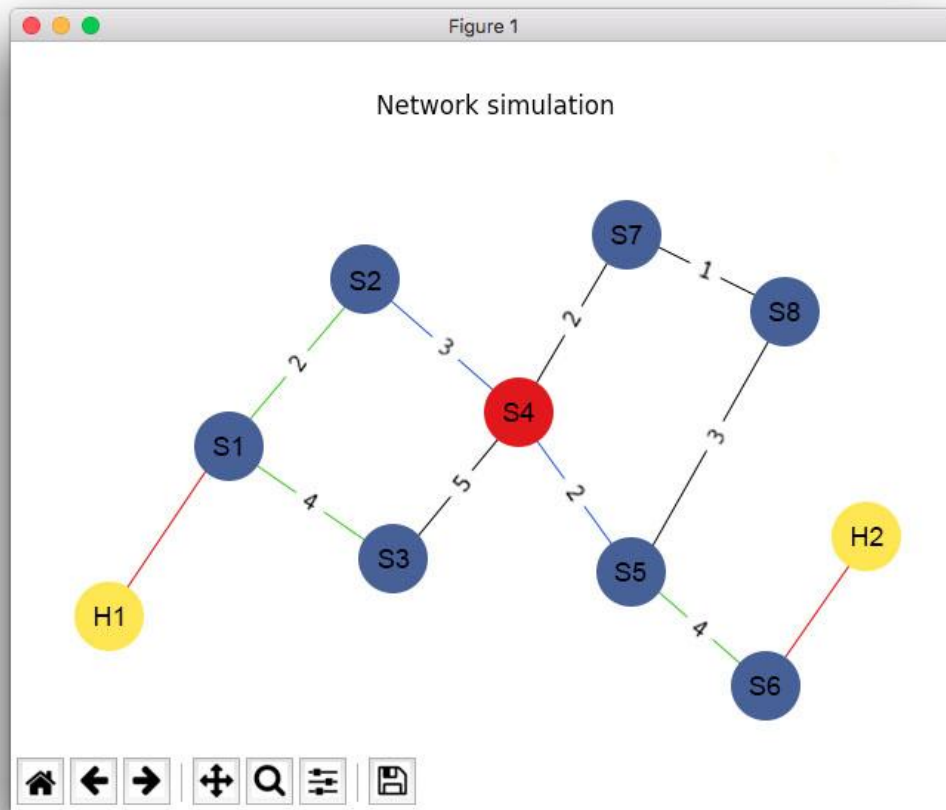


Рисунок 3.4 – Кінець симуляції топології

3.3 Вимоги до апаратно-програмного забезпечення

Вимоги:

- операційна система Window/Mac OS/Linux;
- встановлений Python інтерпретатор;
- встановлений VirtualBox/VmWare з mininet VM.

3.4 Використання mininet для тестування протоколу

Запустимо віртуальну машину через програму віртуалізації (наприклад VirtualBox), та будемо використовувати ssh з хосту для запуску модулів, яким потрібно використовувати сервер X Window Systems. На рисунку 3.5 представлено консоль для початку роботи з mininet VM.

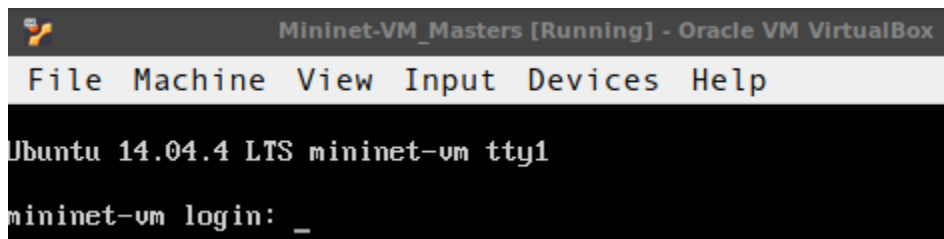


Рисунок 3.5 – Запущена віртуальна машина використовуючи VirtualBox

Для підключення через ssh запускаємо dhclient щоб підключитися через термінал (рис. 3.6).

```

Mininet-VM [Running]
mininet@mininet-vm:~$ sudo dhclient eth1
RTNETLINK answers: File exists
mininet@mininet-vm:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:5a:3f:fa
          inet addr:10.0.3.15  Bcast:10.0.3.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:672 errors:0 dropped:0 overruns:0 frame:0
          TX packets:671 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:71761 (71.7 KB)  TX bytes:57187 (57.1 KB)

eth1      Link encap:Ethernet  HWaddr 08:00:27:3f:db:d3
          inet addr:192.168.56.101  Bcast:192.168.56.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:162315 errors:0 dropped:0 overruns:0 frame:0
          TX packets:87829 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:54371583 (54.3 MB)  TX bytes:29671871 (29.6 MB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:238423 errors:0 dropped:0 overruns:0 frame:0
          TX packets:238423 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:72220507 (72.2 MB)  TX bytes:72220507 (72.2 MB)

mininet@mininet-vm:~$ _

```

Рисунок 3.6 – Запуск dhclient eth1

На рисунку 3.7 наведено ssh вхід до віртуальної машини.

```

[Karinas-MacBook-Pro:~ geshagliuk$ ssh -X mininet@192.168.56.101
[mininet@192.168.56.101's password:
Welcome to Ubuntu 14.04.4 LTS (GNU/Linux 4.2.0-27-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
New release '16.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Wed Dec  5 03:02:47 2018 from 192.168.56.1
[mininet@mininet-vm:~$

```

Рисунок 3.7– З'єднання до віртуальної машини за допомогою ssh.

Для того щоб провести тестування потрібно створити топологію та визначити для каналів наступні характеристики:

- пропускна здатність каналу;
- затримка каналу зв'язку;
- вузол що повинен вийти з строю.

На рисунку 3.8 та 3.9 наведено MinEdit, за допомогою якого створюється топологія (для тестування створимо 16 та 120 вузлів) для SDN мережі та мережі з традиційною архітектурою.

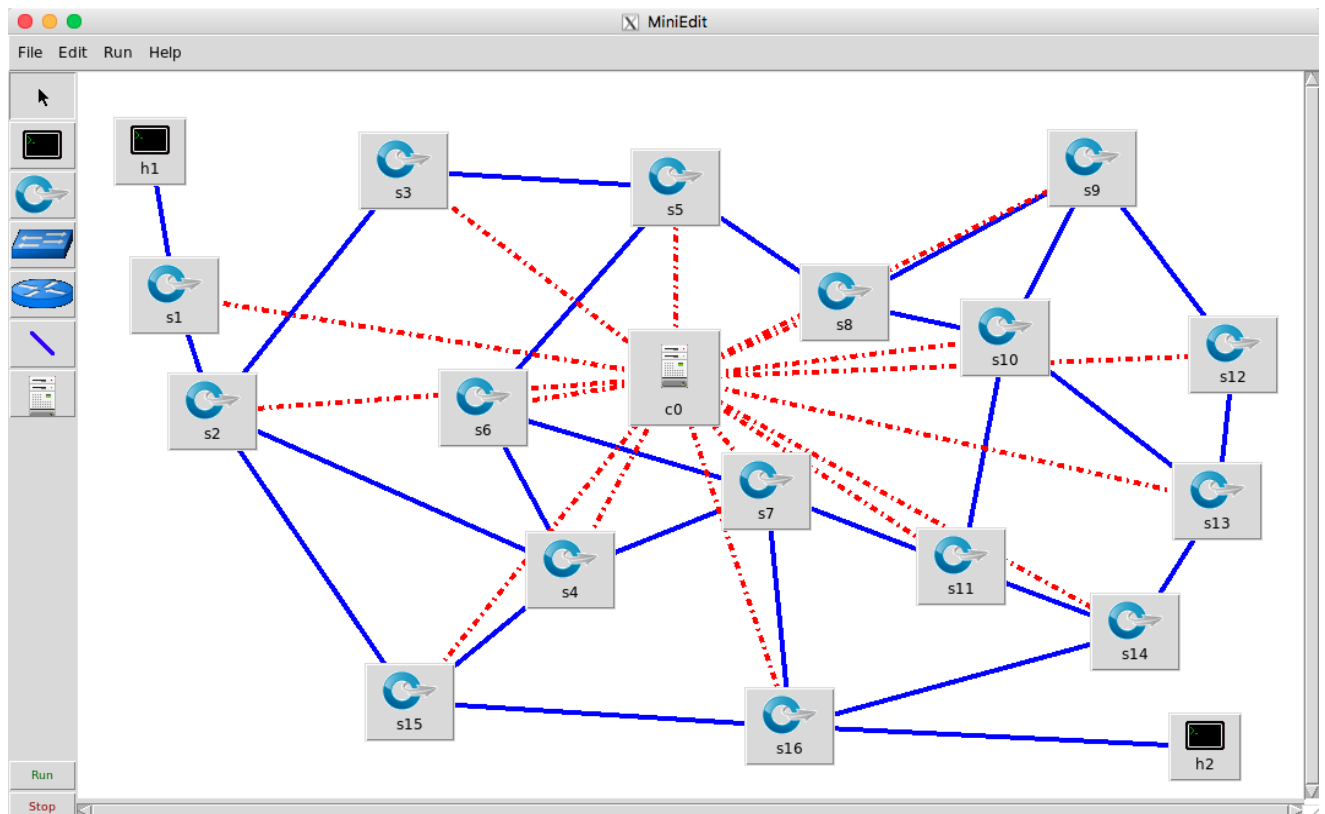


Рисунок 3.8 – Створена топологія для 16 вузлів в SDN

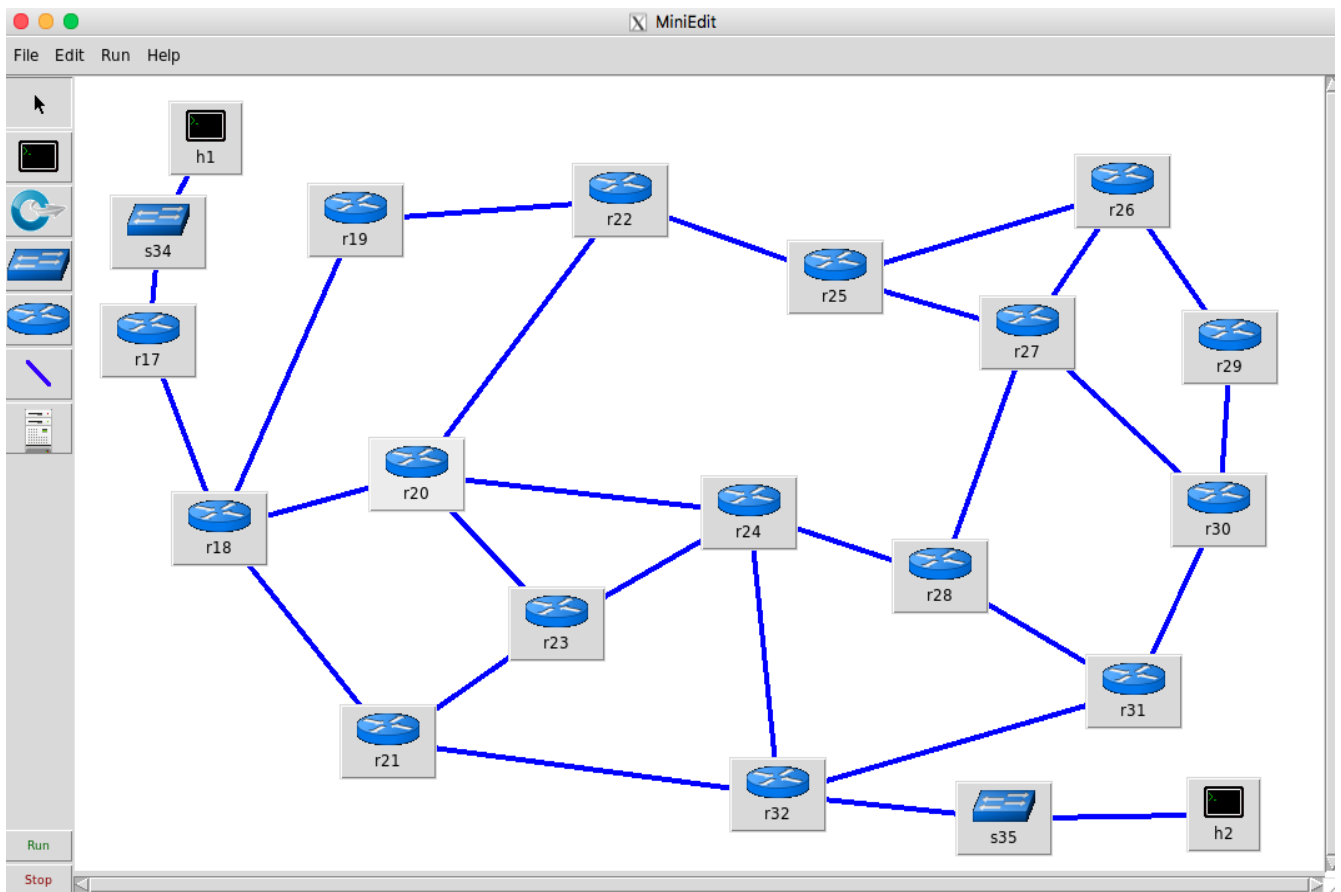


Рисунок 3.9 – Створена топологія для 16 вузлів з традиційною мережі

Для зручності топологія з 120 вузлами буде створюватися за допомогою скрипта, що використовує API mininet.

Для того щоб проаналізувати ефективність розробленого алгоритму, будемо змінювати пропуску здатність та затримку з'єднання в мережі.

Порівнюємо швидкість пересилання в залежності від масштабу мережі. Встановлюємо для всіх каналів пропуску здатність в 10 Мб/с та 2 мс затримки з'єднання як показано на рисунку 3.10.

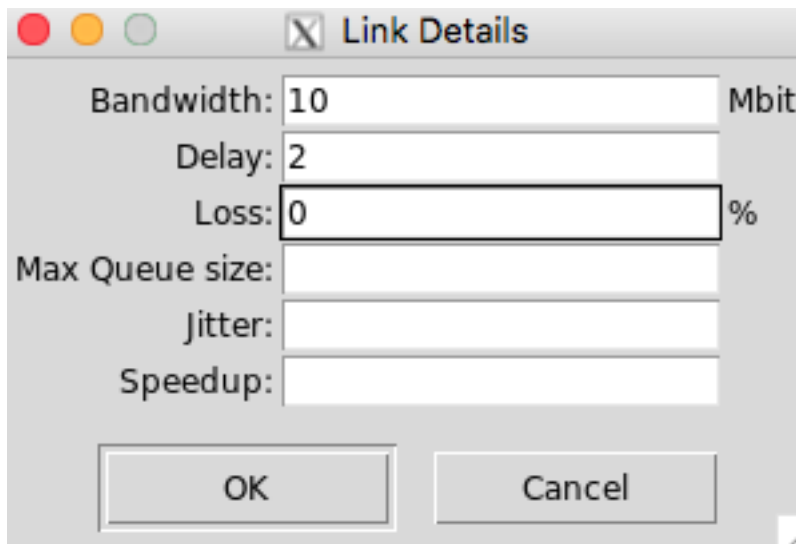


Рисунок 3.10 – Властивості каналу

Також поставимо ступінь втрати пакетів 0. Для отримання результатів тестів використаємо команду ping та pingall між хостами h1 та h2 в мережі з SDN та традиційній. Результати даного тесту можна побачити на рисунку 3.11.

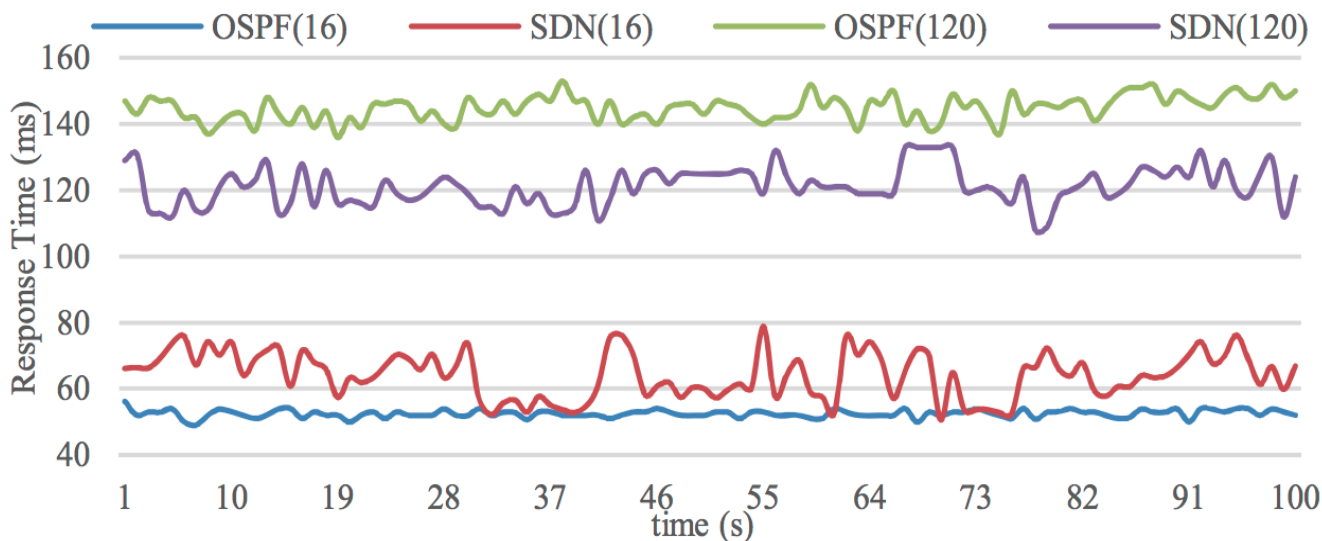


Рисунок 3.11 – Результати тесту

З даного тесту можна побачити, що для великих мереж краще використовувати протоколи в SDN через їх централізованість.

Порівняємо швидкість пересилання в залежності від масштабу мережі та з'єднання з інтернунціями. Для даного тесту встановлюємо для всіх каналів пропускну

здатність в 10 Мб/с, 2 мс затримки та відсоток загублених пакетів 0%. В данному тесті з'єднанні буде мати 15 інтерупцій, тобто 15 разів протокол маршрутизації буде генерувати новий шлях. Результати тесту наведені на рисунку 3.12.

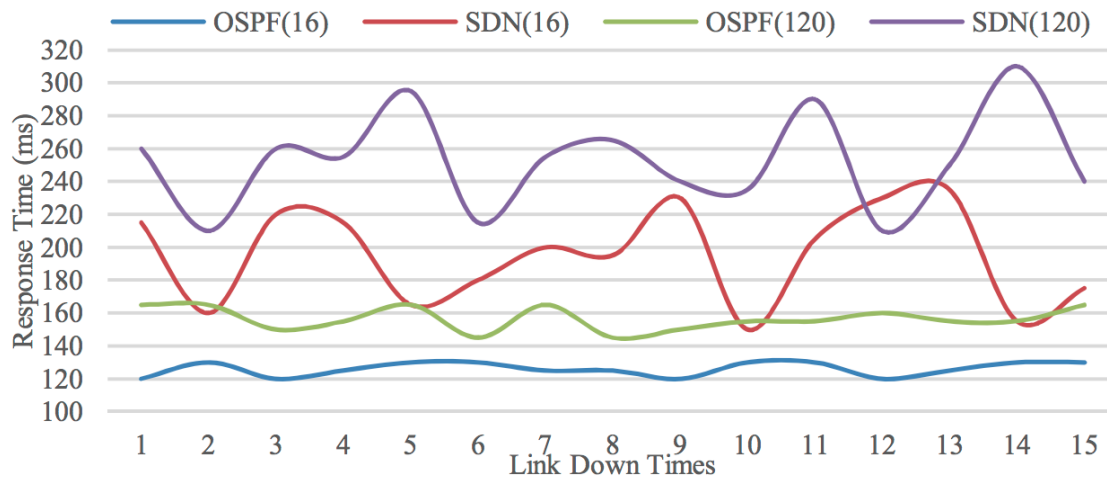


Рисунок 3.12 – Результати тесту

З даного тесту можна побачити, що протокол в SDN значно швидше адаптується до змін топології ніж класичний протокол.

Порівняємо швидкість пересилання в залежності від масштабу мережі, затримці каналу з інтернунціями. Для даного тесту встановлюємо для всіх каналів пропускну здатність в 10 Мб/с, затримки 2 мс, 16 мс, 26 мс, та відсоток загублених пакетів 0%.

З результату тесту (додаток А, плакат 6) можна побачити, що зміна затримки з'єднання не спричинила значного впливу на час пересилання в SDN по відношенню до традиційної мережі.

Висновки до розділу

В даному розділі було розглянуто програмний продукт для симуляції роботи протоколів маршрутизації та конфігурація віртуальної машини для тестування розробленого алгоритму маршрутизації. Наведено обґрунтування використаних технологій. Було розглянуто діаграму класів та архітектуру програмного забезпечення.

Також наведено основні частини інтерфейсу користувача.

4 РОЗРОБКА СТАРТАП-ПРОЕКТУ

У даному розділі описується ідея проекту, аналіз ситуації на ринку та конкурентоспроможності продукту, умови виходу стартап-проекту на ринок, попит та вимоги користувачів, розробляється стратегія поведінки на ринку, виходячи з отриманих висновків.

4.1 Опис ідеї проекту

Спершу необхідно визначити, хто є потенційними користувачами продукту. Продукт створюється для моделювання та аналізу протоколів багатошляхової маршрутизації для контролера програмно конфігурованої мережі, отже, він стане у пригоді розробникам програм для встановлення на контролери.

Запропонований у роботі алгоритм стане цікавим для тих, хто впроваджує програмно конфігуровану мережу як протокол маршрутизації.

У таблиці 4.1 наводиться опис ідеї стартап-проекту, можливі напрями його застосування, та переваги, які отримають користувачі в результаті його використання.

Таблиця 4.1 - Опис ідеї стартап-проекту

Зміст ідеї стартап-проекту	Напрямки застосування	Вигоди для користувача
Створення багатошляхової маршрутизації програмно конфігурованих мереж	1. Впровадження програмно конфігурованої мережі	Ефективність не менша ніж при використанні алгоритмів для класичної архітектури
	2. Модернізація мережі класичної архітектури	Збільшення об'єму трафіку, який можна передавати у мережі

Продовження таблиці 4.1

	3. Збільшення ефективності маршрутизації у мережах великої розмірності	Прискорення маршрутизації та забезпечення стабільної роботи мережі навіть при великих обсягах даних, що передаються
	4. Моделювання роботи мережі із запропонованим алгоритмом	Можливість переглянути побудовані маршрути
	5. Порівняльний аналіз із алгоритмами для класичної архітектури	Наявність значень критеріїв для всіх алгоритмів, що були застосовані до мережі

На даний час для даного стартапу склалися сприятливі умови для виходу на ринок, адже програмно конфігурована мережа ще не була загально впровадженою і Інтернет і досі працює на класичній архітектурі. Лише компанія Google активно використовує нову мережеву парадигму.

У зв'язку із тим, що трафік збільшується у геометричній прогресії, вже виникає потреба у кардинальній зміні структури управління мережею. Але на даний час ще не існує стандарту протоколів для програмно конфігурованих мереж.

За прогнозами компанії Cisco в найближчі 5 років обсяг трафіку збільшиться в 4 рази, причому, мобільний трафік буде подвоюватися щороку. У цей же час мережевий рівень майже не зазнає змін.

Попри те, що загальне впровадження планується на 2020 рік, на даний час немає великої кількості розроблених конкурентних алгоритмів маршрутизації для нового типу мережі.

Відомі протоколи маршрутизації в традиційних мережах неефективні для використання в SDN. Однак, табличні протоколи маршрутизації мають явну перевагу, яка полягає в тому, що коли вузлу потрібно відправити пакет, потрібний маршрут вже побудований в таблицях маршрутизації.

Цей факт є істотним для скорочення часу формування маршруту передачі даних, тому такі алгоритми можуть бути використані в якості базових алгоритмів для розробки протоколів маршрутизації для SDN. До таких відомих табличних протоколів маршрутизації відносяться ті, що широко використовуються в сучасних провідних мережах: протокол RIP (Routing Information Protocol), заснований на дистанційно-векторному алгоритмі, протокол OSPF (Open Shortest Path First), заснований на стані каналу.

Найбільш використовуваним протоколом внутрішньої маршрутизації IGP (Interior Gateway Protocol) є протокол RIP, але він поступово витісняється протоколом OSPF. До нових протоколів зовнішньої маршрутизації EGP (Exterior Gateway Protocol) відноситься протокол BGP і його нова версія - протокол міждоменної маршрутизації IGRP (Enhanced Internet Gateway Routing Protocol), модифікований для використання з OSI адресами замість IP.

В таблиці 4.2 наведено порівняння опису характеристик систем конкурентів та стартап-проекту.

Таблиця 4.2 - Сильні, слабкі та нейтральні характеристики ідеї проекту

№ п/п	Технікоєкономічні характеристики ідеї	(потенційні) товари/концепції конкурентів				W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Мій проект	OSPF	BGP	RIP			
1	Швидкість роботи у мережах великої розмірності	Більша, ніж у алгоритмів класичної архітектури	менша	менша	менша			+
2	Швидкість роботи у мережах малої розмірності	Менша, ніж в алгоритмів класичної архітектури	Більша	Більша	Більша	+		
3	Швидкість відновлення стабільної роботи у разі переривання зв'язку	Інформація знаходиться у контролері, тому - висока	Не висока	Не висока	Не висока			+
4	Ймовірність появи петель маршрутизації	Не виникають	Не знайдено інформації про архітектуру.	Не знайдено інформації	Виникають		+	

Продовження таблиці 4.2

5	Обмеження розміру мережі	Алгоритм показує кращі результати на мережах великої розмірності, на відміну від RIP не обмежує розмір мережі	Сповільнення роботи	Сповільнення роботи	Є обмеження			+
---	--------------------------	---	---------------------	---------------------	-------------	--	--	---

4.2 Технологічний аудит ідеї проекту

В таблиці 4.3 наведений технологічний аудит проекту, який показує список технологій та складових і також їх наявність, за допомогою яких реалізовується ідея проекту.

Таблиця 4.3. - Технологічна здійсненність ідеї проекту

№ п/п	Ідея стартап-проекту	Технології необхідні для реалізації	Наявність цих технологій	Доступність технологій
1	Моделювання мережі	API, побудова графів	Наявна	Доступна
2	Моделювання маршрутизації за умови класичної архітектури	Використання алгоритмів багато шляхової маршрутизації	Наявна	Доступна

Продовження таблиці 4.3

3	Моделювання маршрутизації за умови архітектури програмно конфігурованої мережі	Використання алгоритмів, які враховують особливості нової мережевої парадигми	Потребує розробки	Базується на табличних алгоритмах, які доступні
4	Встановлення параметрів мережі	Інтерфейс взаємодії із користувачем	Потребує розробки	Використовує ті технології, та засоби, які є вільними для використання
5	Аналіз	Використання засобів аналізу	Потребує розробки	Використовує ті технології, та засоби, які є вільними для використання
Обрана технологія реалізації ідеї проекту: Застосування існуючих алгоритмів багато шляхової маршрутизації для мереж класичної архітектури, розробка модифікованого алгоритму для програмно конфігурованої мережі для подальшого аналізу.				

4.3 Аналіз ринкових можливостей запуску стартап-проекту

На даний час ще не існує стандарту протоколів для програмно конфігурованих мереж, тому для даного стартапу склалися сприятливі умови для виходу на ринок, адже програмно конфігурована мережа ще не була загально впровадженою і Інтернет і досі працює на класичній архітектурі. Лише компанія Google активно використовує нову мережеву парадигму.

В таблиці 4.4 наведемо попередні характеристики потенційного ринку.

Таблиця 4.4 - Попередня характеристика потенційного ринку

№ п/п	Показники стану ринку	Характеристика
1	Кількість головних гравців, од	3
2	Загальний обсяг продаж, грн/ум.од	
3	Динаміка ринку (якісна оцінка)	Попит зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Присутні обмеження: 1. Необхідне загальне впровадження нової архітектури – встановлення контролерів 2. Доцільно застосовувати на мережах великої розмірності 3. Відсутність на даний час загального стандарту протоколів програмно конфігурованих мереж
5	Специфічні вимоги до стандартизації та сертифікації	Відсутні
6	Середня норма рентабельності в галузі (або по ринку), %	

За рахунок новизни і ще не повного впровадження технології, є шанс раннього заходу на ринок. Це може стати вирішальним фактором, щоб зайняти цю нішу як один з основних гравців. В таблиці 4.5 наведено характеристику потенційних клієнтів стартап-проекту.

Таблиця 4.5 - Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Потреба у передачі великої кількості трафіку	Будь-які користувачі Інтернет	Залежить від виду діяльності, виду використовуваної техніки	1. Стабільність роботи мережі навіть при збільшенні обсягу обміну інформацією
2.	Розширення мережі але її незмінне функціонування, зміни у мережі, впровадження інновацій	Провайдери	Різна поведінка, яка залежить від географічного положення та пріоритетів	Мінімізація витрат на реконфігурацію мережі

Після формування списку потенційних можливих клієнтів, проведено аналіз ринку, та наведено список факторів, що можуть перешкодити успішному впровадженню системи.

Дані фактори наведено в таблиці 4.6.

Таблиця 4.6 - Фактори загроз стартап-проекту

№ п/п	Фактор загрози	Зміст загрози	Можлива (необхідна) реакція компанії
1	Затягування періоду переходу до іншої архітектури мережі	Необхідно змінити систему, на якій працює всесвітнє павутиння – це вимагає заміни великої кількості устаткування	Показати необхідність переходу до іншої конфігурації мережі – влаштувати презентації великим компаніям
2	Низька довіра та зацікавленість користувачів до нової системи	На момент запуску системи, до того часу, поки вона не проявить себе, як принаймні рівну конкурентам, буде присутня недовіра до нового продукту	Показувати соціальну рекламу
3	Втрата даних	Повне оновлення системи може призвести до втрати інформації	Резервне копіювання

Також складено перелік факторів, що позитивно впливають на можливість успішного впровадження системи (таблиця 4.7).

Таблиця 4.7 - Фактори можливостей стартап-проекту

№ п/п	Фактор можливостей	Зміст можливості	Можлива реакція компанії
1	Неминуче перевантаження і вихід із ладу старої мережі з класичною архітектурою	Зростання трафіку призведе до перевантаження системи	Завчасна підготовка до такої ситуації для подальшого швидкого реагування
2	Маленькі компанії можуть наслідувати перехід великих компаній на нову архітектуру мереж	Успішні компанії вже впроваджують новий тип мережі, менші підприємства повторюватимуть такі дії як шлях до успіху	Покащення роботи алгоритмів для менших мереж
3	Розширення системи	Побудова нової мережі паралельно зі старою	Забезпечення переносу даних

Проведено аналіз поточного ринку, та сформовано риси конкуренції на ринку (таблиця 4.8).

Таблиця 4.8 - Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Тип конкуренції - монополія	Існують чіткі лідери, які представлені на ринку з самого початку роботи класичної мережі	Знайти кардинально новий підхід – наприклад, нова технологія чи архітектура мережі
2. За рівнем конкурентної боротьби - інтернаціональна	Монополісти є інтернаціональними компаніями, грають на міжнародному ринку	Впровадити нове технологічне рішення
3. За галузевою ознакою - міжгалузєва	Інтернет – це те, що існує в усіх галузях	Аналіз еволюції потреб в основних сферах
4. За характером конкурентних переваг - нецінова	Система не передбачає здійснення жодних платежів користувачем, вона існує як єдине ціле і не можна виділити ціну за користування окремими її функціями, хоч провайдери локально і встановлюють ціни на послуги, єдиної вартості Інтернету не існує	Зменшення витрат на впровадження за рахунок програмного керування

Проаналізуємо конкретність в середині галузі за М.Портером. Аналіз наведено в таблиці 4.9.

Таблиця 4.9 - Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
	OSPF RIP BGP	Всі потенційні конкуренти у рівних умовах	Немає	Можуть опиратися введенню нової системи, але це неминуче	Немає
Висновки:	Лише застарілі протоколи	Відсутній стандарт протоколу	Немає	Всі змушені будуть перейти на нову архітектуру мережі	Немає

Не зважаючи на наявну сьогодні конкуренцію ринок є привабливим для впровадження системи та пропонування свого продукту. Проаналізувавши ситуацію, що була наведена в таблиці 4.9, вимоги майбутніх користувачів системи, що наведено у таблиці 4.5, було сформовано набір ключових факторів конкурентоспроможності системи - таблиця 4.10 та обґрунтованої важливості.

Таблиця 4.10 - Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Розширення системи	Новий алгоритм не наводить обмежень на систему
2	Передача великої кількості трафіку	

Продовження таблиці 4.10

3	Забезпечення надійності	У разі втрати зв'язку чи виходження з ладу якогось із вузлів відбувається швидка пербудова топології мержі, що дозволяє у найкоротший час виправити маршрут та доставити повідомлення адресатові
4	Швидкість передачі даних	Швидкість маршрутизації у мережах великої розмірності збільшується, що призводить до менших затрат часу на обмін пакетами
5	Керування програмним стає	Завдяки відділенню функції управління від апаратної частини комутаторів і поява нового пристрою у мережі – контролера , керування мережею спрощується і тепер виконується в управляючому пристрої, а не в кожному комутаторі, як це було раніше
6	Закладена можливість швидкої модифікації системи	Архітектура мережі програмно конфігурованої побудована таким чином, що легко піддається розширенню та масштабуванню.

Сформовані фактори конкурентоспроможності допомагають виявити сильні та слабкі сторони системи стартап-проекту. Аналіз наведено у таблиці 4.11.

Таблиця 4.11 - Порівняльний аналіз сильних та слабких сторін

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з системою рекомендацій на основі поведінкової моделі користувача						
			-3	-2	-1	0	+1	+2	+3
1	Розширення системи	19						+	

Продовження таблиці 4.11

2	Передача великої кількості трафіку	18	+						
3	Забезпечення надійності	14				+			
4	Швидкість передачі даних	20			+				
5	Керування стає програмним	10		+					
6	Закладена можливість швидкої модифікації	19			+				

Щоб оцінити, які складнощі можуть виникнути на ринку необхідно провести додатковий аналіз свого продукту за перевагами та недоліками.

SWOT-аналіз наведено у таблиці 4.12, показує оцінку факторів зовнішнього та внутрішнього походження, що впливають на процес впровадження та виведення продукту на ринок.

Таблиця 4.12. SWOT-аналіз стартап-проекту

Внутрішні фактори	
Сильні сторони	Слабкі сторони
Розширення системи Передача великої кількості трафіку Забезпечення надійності Швидкість передачі даних Керування стає програмним Закладена можливість швидкої модифікації системи	Висока ціна Довгострокові зміни

Продовження таблиці 4.12

Зовнішні фактори	
Можливості	Загрози
Неминуче перевантаження і вихід із ладу старої мережі з класичною архітектурою Маленькі компанії можуть наслідувати перехід великих компаній на нову архітектуру мереж	Затягування періоду переходу до іншої архітектури мережі Низька довіра та зацікавленість користувачів до нової системи Втрата даних

Необхідно розробити алгоритм дій по виводу продукту на ринок. Для цього складемо список заходів, які пришвидшать цей процес. Альтернативи ринкового впровадження стартап-проекту наведено в таблиці 4.13.

Таблиця 4.13. Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Акцентування на випадках виходу з ладу старої системи або будь-яких виявах вузьких місць	Інвестиції гарантовані	4 – 5 років
2	Паралельне створення масштабної мережі з новою архітектурою та приклад застосування програмного продукту для нової мережі	Інвестиції можливі	4 роки

Продовження таблиці 4.13

3	Широкомасштабне впровадження архітектури нової	Ресурси у повному обсязі будуть надходити в результаті повного функціонування системи та за рахунок різних джерел – наприклад, реклама	6 років
---	--	--	---------

4.4 Розроблення ринкової стратегії проекту

Щоб правильно вибрати стратегію завоювання ринку, необхідно точно знати, хто є потенційними користувачами. Для цього проводиться аналіз зацікавлених у продукті людей та утворення цільових груп споживачів. Цільові групи наведені у таблиці 4.14. Серед них необхідно вибрати ті, що є найбільш перспективними для збутку проекту.

Таблиця 4.14 - Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Місцеві провайдери	Низька готовність	Низький попит	-	Висока складність
2	Великі компанії з власними глобальними мережами	Висока готовність	Високий попит		Низька складність

Продовження таблиці 4.14

3	Власники хмарних сховищ	Висока готовність	Середній-вище середнього		Низька складність
4	Користувачі мережі Інтернет	Низька готовність	Низький попит		-
Які цільові групи обрано: Великі компанії з власними глобальними мережами, Власники хмарних сховищ					

Визначаємо базову стратегію розвитку – наведено у таблиці 4.15.

Таблиця 4.15 - Визначення базової стратегії розвитку

Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
Паралельне створення масштабної мережі з новою архітектурою та приклад застосування програмного продукту для нової мережі	Стратегія диференціації	Здатність витримувати великий обсяг трафіку	Розробка алгоритмів для багато шляхової маршрутизації у мережі програмно конфігурованих

Визначаємо базову стратегію конкурентної поведінки – у таблиці 4.16.

Таблиця 4.16 - Визначення базової стратегії конкурентної поведінки

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки*
1	Один із	Залучення нових споживачів паралельно із переведенням існуючих на нову архітектуру мережі	Лише те, що визначатиметься стандартом	Лідерська

Враховуючи вимоги споживачів до продукту, а також обрані стратегії розвитку та конкурентної поведінки розроблена стратегія позиціонування (таблиця 4.17), яка буде притаманна продукту на ринку, для можливості його максимальної реалізації та його однозначної ідентифікації.

Таблиця 4.17 - Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартаппроекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Потреба у передачі великої кількості трафіку	Новий принцип управління мережею	Оптимізація	Новий принцип управління мережею Отримання релевантних рекомендацій

Продовження таблиці 4.17

2	Збереження усіх даних старої мережі	Резервне копіювання	Збереження	Збереження усіх даних старої мережі Розширення мережі але її незмінне функціонування, зміни у мережі, впровадження інновацій
3	Розширення мережі але її незмінне функціонування, зміни у мережі, впровадження інновацій	Маршрутизація	Розвиток	

4.5 Розроблення маркетингової програми стартап-проекту

Маркетингова програма проекту буде направлена на те, що перехід на нову технологію є неминучим. Стара архітектура мережі скоро не зможе витримувати навантаження за рахунок збільшення обсягу трафіку, а альтернатив програмно конфігурованій мережі ще не розробили.

Тобто, на даний момент не існує навіть теоретичних розробок іншої архітектури такого масштабу.

Цей факт свідчить про те, що усі користувачі мережі Інтернет скоро будуть змушені перейти на нову архітектуру, а провайдери в свою чергу будуть змушені забезпечити її впровадження.

Провівши аналіз потреб потенційних користувачів продукту, було відібрано ключові - у таблиці 4.18. Вони є необхідними до врахування для впровадження проекту та забезпечення конкурентноспроможності.

Таблиця 4.18 - Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Потреба у передачі великої кількості трафіку	Немає обмежень щодо розміру мережі	Ефективне управління
2	Розширення мережі але її незмінне функціонування, зміни у мережі, впровадження інновацій	Програмне керування	Швидкість
3	Недороге впровадження	Програмні зміни а не апаратні	Зміни можна впровадити за допомогою коду

Виконаємо опис трьох рівнів моделі товару (таблиця 4.19).

Таблиця 4.19 - Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові
I. Товар за задумом	Програма моделювання багато шляхової маршрутизації у мережі що є програмно конфігурованою

Продовження таблиці 4.19

II. Товар у реальному виконанні	Властивості/характеристики		
	Моделювання маршрутизації у мережі, Програмне управління за допомогою контролера		
	Додається документація. Було проведено тестування		
	Програма, що встановлюється на контролер		
III. Товар із підкріпленням	Програмне забезпечення		

Визначення цінових меж стартап-проекту показано у таблиці 4.20. Схема отримання прибутку складна і залежить від багатьох факторів. Тому оцінити межі ціни продукту достатньо складно.

Таблиця 4.20 - Визначення меж встановлення ціни

№ п/п	Рівень цін на товаризамінники	Рівень цін на товарианалог	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	невідомо	невідомо	Доходи підприємців	На даний момент не можна оцінити верхню та нижню границі цін

За проаналізованою ситуацією на ринку та потенційними користувачами було сформовано систему збуту вказану у таблиці 4.21 та концепцію маркетингових комунікацій – у таблиці 4.22.

Таблиця 4.21 - Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Оплата послуг провайдеру	Провайдер надає доступ до мережі	Провайдер-користувач	Програма

Таблиця 4.22 - Концепція маркетингових комунікацій

Специфіка поведінки обраних цільових клієнтів	Канали комунікацій, що використовують цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
Мають побоювання щодо переходу до нової системи	Мережа Інтернет	Необхідність переходу до нової мережі, наслідки, якщо цього не станеться	Обґрунтування важливості та необхідності переходити на нову технологію	Показати збільшення трафіку за останні роки

Висновки до розділу

Розроблюваний проект має ряд істотних переваг на ринку у порівнянні із представленою давно класичною архітектурою мережі. По-перше, спрощується сам процес створення маршрутів. На відміну від стандартної мережі, де маршрутизація - це розподілений ітеративний процес, при якому робоча топологія мережі «обчислюється» спільно всіма пристроями, в SDN - це програма моделювання мережі з заданими параметрами. Використання цієї моделі відкриває нові можливості для створення мережі з вимогами, неможливими в рамках традиційного інжинірингу трафіку і з використанням стандартних протоколів маршрутизації.

По-друге, значно збільшуються можливості для інновації. У традиційній розподіленій моделі необхідність приведення функціональності до спільного знаменника для можливості взаємодії між незалежними пристроями визначає істотну консервативність системи по відношенню до нововведень. У SDN інновація - це питання написання нового додатка.

По-третє, замість складних і дорогих маршрутизаторів можна використовувати більш прості пристрої.

Впровадження продукту значно збільшує можливості для інновації та розвитку мереж. У традиційній розподіленій моделі необхідність приведення функціональності до спільного знаменника для можливості взаємодії між незалежними пристроями визначає істотну консервативність системи по відношенню до нововведень. У SDN інновація - це питання написання нового додатка.

Саме простота модифікації мережі, збільшення її можливостей та надійності – основні фактори для виходу на ринок та закріплення там серед конкурентів.

Завдяки новизні технології всі знаходяться у рівних умовах конкуренції, тому вихід у лідери залежить виключно від якості розробленого продукту.

Також варто зазначити, що перехід на нові технології є неминучим, тому що старі мережі не витримують прогнозованого збільшення трафіку.

Продукт є конкурентоспроможним.

ВИСНОВКИ

Розвиток різних технологій призвели до того, що процес контролю мереж стає дедалі складніше. Хмарні технології, мобільні додатки, IoT, великі дані повинні процвітати з архітектурою що була створена тридцять років тому, адже традиційна архітектура мереж не була призначена для такої кількості сервісів. Тому, і було вибрано тему дослідження нової архітектури мережі та мережеві протоколи.

У першому розділі було розглянуто архітектуру мережі що використовується сьогодні, і які вона має недоліки. Було наведено нові типи архітектури мереж, та чому саме SDN архітектура була використана в рамках даної роботи. Було описано головні задачі протоколу маршрутизації, та задачі що були поставлені для створення протоколу маршрутизації. Також було представлено діаграму діяльності та діаграму прецедентів системи симуляції протоколів.

У другому розділі було наведено основні алгоритми що використовуються протоколами маршрутизації, як вони працюють, та зроблено порівняльний аналіз. Було аналізовано літературні джерела на наступні теми: протоколи маршрутизації, алгоритми маршрутизації, запропоновані модифікації. Також була розглянута змістовна постановка задачі та зроблено удосконалення методу розв'язання даної задачі. Було наведено результати досліджень ефективності методу.

У третьому розділі наведено засоби розробки що використовувалися для симуляції протоколів маршрутизації та дослідження їх ефективності. Було наведено архітектуру програмного забезпечення та зроблена інструкція користувача.

З результатів проведених експериментів виявлено залежність традиційної та SDN архітектури від масштабності топології. Наведено також графіки з використанням традиційних протоколів маршрутизації та удосконаленим алгоритмом в SDN.

У четвертому розділі була описана ідея проекту, зроблено аналіз ринкової ситуацій та конкурентоспроможності продукту. Використана в даній роботі архітектура має ряд переваг у порівнянні з класичною архітектурою мережі. Було зазначено що перехід на

нові технології є неминучим. Адже, кожен день збільшується кількість і різноманітність пристроїв комп'ютерної мережі, і різні технології що впроваджуються мають обмеження від загально використаної традиційної мережі.

Сьогодні, найбільший виклик – швидкість змін. Адже на даний момент на даному ринку, немає консенсусу, тому що немає стандартів.

Також за матеріалами дисертації було опубліковано дві наукові роботи: одна стаття та одні тези доповіді на конференції [19, 20].

ПЕРЕЛІК ПОСИЛАНЬ

1. CERN Accelerating science – Information Management: A proposal by Tim Berners-Lee, CERN, March 1989, May 1990 [Електронний ресурс] – Режим доступу до ресурсу: <https://cds.cern.ch/record/369245/files/dd-89-001.pdf> (дата звернення 07.12.2018).
2. M. Beck. “On the Hourglass model”. CoRR. 2016. Abs/1607.07183.
3. L. Meinan, L. Ruiying. “Modeling End-to-End Delay of Network in Accelerated Reliability Testing”. 2013 Third International Conference on Instrumentation, Measurement Computer, Communication and Control. Shenyang, China. 21-23 Sept. 2013. pp. 1371-1374.
4. WANSPEAK new wan edge articles and insights – SD-WAN Economics: What Will You Do With a 5% Increase in Budget? [Електронний ресурс] – Режим доступу до ресурсу: <https://blog.silver-peak.com/sd-wan-economics-what-will-you-do-with-a-5-increase-in-budget> (дата звернення 07.12.2018).
5. M. A. Alqarni. “Benefits of SDN for Big data applications”. 2017 14th International Conference of Smart Cities: Improving Quality of Life Using ICT & IoT (HONET-ICT). Irbid, Jordan. 9-11 Oct. 2017. pp 74-77.
6. Поповський В.В, Лемешко О.В., Ковальчук В.К., Плотніков М.Д., Картушин Ю.П., Попонін О.М., Агєєв Д.В., Сабурова С.О., Олійник В.Ф., Персіков А.В., Лошаков В.А., Селіванов К.О. Телекомунікаційні системи та мережі. Структури й основні функції. Том 1. 2018. Р. 10.
7. IETF Tools – OSPF Version 2 [Електронний ресурс] – Режим доступу до ресурсу: <https://tools.ietf.org/html/rfc2328> (дата звернення 07.12.2018).
8. IETF Tools – Cisco’s Enhanced Interior Gateway Routing Protocol (EIGRP) [Електронний ресурс] – Режим доступу до ресурсу: <https://tools.ietf.org/html/rfc7868> (дата звернення 07.12.2018).

9. IETF Tools – RIP Version 2 [Електронний ресурс] – Режим доступу до ресурсу: <https://tools.ietf.org/html/rfc2453> (дата звернення 07.12.2018).
10. D. Teare, R. Graziani, B. Vachon. “OSPF Implementation”. Cisco Press. 2015. С. 155.
11. S. Sharma, D. Staessens, D. Colle, M. Pickavet, P. Demeestr. “Automatic bootstrapping of OpenFlow networks”. 2013 19th IEEE Workshop on Local & Metropolitan Area Networks (LANMAN). Brussels, Belgium. 10-12 Apr. 2013. pp. 1-6.
12. ONF – OpenFlow Switch Specification Version 1.5.1 (Protocol version 0x06) [Електронний ресурс] – Режим доступу до ресурсу: <http://www.opennetworking.org/wp-content/uploads/2014/10/openflow-switch-v1.5.1.pdf> (дата звернення 07.12.2018).
13. IETF Tools – Routing Information Protocol [Електронний ресурс] – Режим доступу до ресурсу: <https://tools.ietf.org/html/rfc1058> (дата звернення 07.12.2018).
14. IETF Tools – The OSPF Specification [Електронний ресурс] – Режим доступу до ресурсу: <https://tools.ietf.org/html/rfc1131> (дата звернення 07.12.2018).
15. IETF Tools – OSPF protocol analysis [Електронний ресурс] – Режим доступу до ресурсу: <https://tools.ietf.org/html/rfc1245> (дата звернення 07.12.2018).
16. IETF Tools – OSPF for IPv6 [Електронний ресурс] – Режим доступу до ресурсу: <https://tools.ietf.org/html/rfc5340> (дата звернення 07.12.2018).
17. Cisco Systems – Cisco ASA 5500 Series Configuration Guide using the CLI. 2014. С. 23-1.
18. H. Y. Lee, A. Nakao. “Improving Routing Table Lookup in Software Routers”. IEEE Communications Letters. 01 Apr 2015. Vol. 19. pp 957-960.
19. Кулаков Ю. А. Формирование множества непересекающихся путей в компьютерных сетях с применением алгоритма «обратной волны» / Кулаков Ю. А., Коган А. В., Диброва М. А., Чхаидзе Д. М. Проблеми інформатизації та управління: збірник наукових праць. – К.: НАУ, 2015. – Вип. 4 (52). – С.68-73.

20. Пастрелло Н., Коган А. В., Кулаков Ю. О., Павленкова Є. В., Мачехин М. “Метод теорії прийняття рішень орієнтований на маршрутизацію в програмно-конфігурованій мережі”. Східно-Європейський журнал передових технологій (подано до друку).
21. Пастрелло Н., Коган А.В. “Спосіб організації багатошляхової маршрутизації в SDN”. Всеукраїнська наукова-практична конференція молодих вчених та студентів «Інформаційні системи та технології управління» (ІСТУ-2018). м. Київ. НТУУ «КПІ ім. Ігоря Сікорського», 29-30 грудня 2018 р. С. 205-210.

Додаток А

Графічний матеріал

СТРУКТУРНА СХЕМА ДІЯЛЬНОСТІ

СТРУКТУРНА СХЕМА ВАРІАНТІВ ВИКОРИСТАННЯ

СХЕМА СТРУКТУРНА БАЗИ ДАНИХ

СТРУКТУРНА СХЕМА КЛАСІВ

БЛОК-СХЕМА АЛГОРИТМУ

Результати експериментальних досліджень

КОПІЇ ЕКРАННИХ ФОРМ