Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет навчально-науковий центр заочної форми навчання
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

АТЕСТАЦІЙНА РОБОТА

Пояснювальна записка

Рівень вищої освіти другий (магістерський)

Модифікований квест-метод аналізу оверлейних мереж з невизначеною кількістю оверлеїв

(тема)

Виконав:
студент II курсу, групи СПзм-18-2
Артьомов С.В.
(прізвище, ініціали)

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

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

Освітня програма
Системне програмування
(повна назва освітньої програми)

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

Допускається до захисту

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

2020 р.

Харківський національний університет радіоелектроніки

Факультет навчально-науковий центр заочної форми навчання

Кафедра електронних обчислювальних машин

Рівень вищої освіти другий (магістерський)

Спеціальність 123 – Комп'ютерна інженерія
(код і повна назва)

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

Освітня програма Системне програмування
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
"____" _____ 20___ р.

ЗАВДАННЯ

НА АТЕСТАЦІЙНУ РОБОТУ

студентові Артьомову Сергію Віталійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Модифікований квест-метод аналізу оверлейних мереж з невизначеною кількістю оверлеїв

затверджена наказом по університету від "30" березня 2020 р. № 43Стз

2. Термін подання студентом роботи до екзаменаційної комісії 18 травня 2020 р.

3. Вхідні дані до роботи обчислювальна станція SR-205 – 5 шт.;
фрагмент комп'ютерної мережі кафедри ЕОМ;
програмне середовище Cisco Packet Tracert;
програмне середовище Network Simulator;
програмне середовище NET-Simulator;
програмне середовище Graphical Network Simulator 3.

4. Перелік питань, що потрібно опрацювати в роботі
Вступ.
1. Особливості побудови динамічних оверлейних мереж.
2. Методи аналізу оверлейних мереж з невизначеною кількістю оверлеїв.
3. Модифікований квест-метод аналізу оверлейних мереж з невизначеною кількістю оверлеїв.
4. Дослідження ефективності роботи системи аналізу оверлейних мереж
Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) Демонстраційні матеріали. Плакати – 12 арк. ф. А4

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.	Огляд прототипів рішення.	31.03.20-13.04.20	
2.	Розробка алгоритмів.	14.04.20-20.04.20	
3.	Моделювання.	21.04.20-29.04.20	
4.	Проведення експериментальних досліджень.	30.04.20-03.05.20	
5.	Складання висновків.	04.05.20-05.05.20	
6.	Оформлення пояснювальної записки		
	та презентаційних матеріалів.	06.05.20-11.05.20	
7.	Подання атестаційної роботи керівникові та її		
	попередній захист	12.05.20-13.05.20	
8.	Подання атестаційної роботи на рецензування	14.05.20-15.05.20	

Дата видачі завдання 30 березня 2020 р.

Студент _____
(підпис)

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

РЕФЕРАТ

Пояснювальна записка атестаційної роботи: 79 с., 13 рис., 1 табл., 2 дод., 26 джерел.

ОВЕРЛЕЙНА МЕРЕЖА, АНОНІМНА МЕРЕЖА, КВЕСТ-МЕТОД, ПРОТОКОЛ, СЕРВЕР.

Метою атестаційної роботи є модифікація квест-методу аналізу оверлейних мереж з динамічною структурою та невизначеною кількістю оверлеїв для оцінки рівня анонімності в мережі.

У ході виконання атестаційної роботи було проаналізовано особливості побудови динамічних оверлейних мереж, розглянуто методи аналізу оверлейних мереж з невизначеною кількістю оверлеїв, зокрема контурний метод та методи, що використовуються в анонімних мережах, розроблено модифікований квест-метод аналізу оверлейних мереж з невизначеною кількістю оверлеїв, створено програмний продукт та проведено дослідження ефективності роботи системи аналізу оверлейних мереж з невизначеною кількістю оверлеїв.

ABSTRACT

Master’s thesis: 79 pages, 13 figures, 1 tables, 2 appendices, 26 sources.

OVERLAY NETWORK, ANONYMOUS NETWORKS, QUEST-METHOD, PROTOCOL, SERVER.

The major goal of this thesis is to modify the quest method for analyzing overlay networks with a dynamic structure and an indefinite number of overlays to assess the level of anonymity in the network.

In order to the features of building dynamic overlay networks were analyzed, methods for analyzing overlay networks with an indefinite number of overlays were considered, in particular, the contour method and methods used in anonymous networks, a modified quest method for analyzing overlay networks with an undetermined number of overlays was developed, and a software program was created The product has been tested for the effectiveness of the analysis system of overlay networks with an undetermined number of overlays.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ 7
ВСТУП 8
1 ОСОБЛИВОСТІ ПОБУДОВИ ДИНАМІЧНИХ ОВЕРЛЕЙНИХ МЕРЕЖ 10
1.1 Принципи побудови та функціонування оверлейних мереж 10
1.2 Особливості динамічних оверлейних мереж 21
1.3 Постановка задачі 24
2 МЕТОДИ АНАЛІЗУ ОВЕРЛЕЙНИХ МЕРЕЖ З НЕВИЗНАЧЕНОЮ КІЛЬКІСТЮ ОВЕРЛЕЇВ 25
2.1 Контурний метод аналізу оверлейної мережі 25
2.2 Аналіз анонімних оверлейних мереж 34
3 МОДИФІКОВАНИЙ КВЕСТ-МЕТОД АНАЛІЗУ ОВЕРЛЕЙНИХ МЕРЕЖ З НЕВИЗНАЧЕНОЮ КІЛЬКІСТЮ ОВЕРЛЕЇВ 40
4 ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ РОБОТИ СИСТЕМИ АНАЛІЗУ ОВЕРЛЕЙНИХ МЕРЕЖ З НЕВИЗНАЧЕНОЮ КІЛЬКІСТЮ ОВЕРЛЕЇВ 50
ВИСНОВКИ 60
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ 61
ДОДАТОК А Графічний матеріал атестаційної роботи 65
ДОДАТОК Б Фрагменти коду програмного забезпечення для аналізу динамічної оверлейної мережі 72

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

КОД – конвеєрна обробка даних

КСПОД – комп’ютерна система з покроковою обробкою даних

DDR – подвоєна передача даних (англ., Double data rate)

FIFO – перший прийшов перший вийшов (англ., first in, first out)

NP – недетермінований багаточлен (англ., non-deterministic polynomial)

MIMD – множинний потік команд (англ., multiple instruction, multiple data)

SSD – твердотілий накопичувач (англ., solid-state drive)

ВСТУП

Велика кількість задач, відомі як задачі організації покрокової обробки інформації, характеризуються сукупністю практичних застосувань, які постійно доповнюються, займаючи важливе місце в розробці проблем комбінаторної оптимізації. Дана прикладна задача в широкому сенсі є задачею поточного планування, в процесі якого вибираються дані і визначаються траєкторію їх передачі між компонентами оверлейної мережі. В умовах кожної задачі міститься опис оверлейної мережі, що визначає сукупність можливих шляхів проходження до кінцевих вузлів всіх даних. Як правило, структурні параметри мережі залишаються незмінними протягом усього процесу рішення задачі.

В галузі комп’ютерної інженерії існують напрями, що об’єднані задачею організації поетапної обробки даних. Можна виділити найбільш цікаві з них. Наприклад, в процесі аналізу даних мережі важливим кроком є попередня обробка даних. Вона складається з сукупності послідовних процедур: очищення, нормалізація, тощо.

При проектуванні оверлейної мережі для виконання задачі попередньої обробки даних існує два напрями. Перший полягає у створенні багатофункціональних систем обробки даних. Кожна з систем виконує повний цикл обробки кожної з порцій даних. Тобто порція даних, при надходженні до такої обчислювальної мережі монопольно займає підсистему на весь час її обробки.

Другий напрямок полягає у створенні однофункціональних систем, що характерно для хмарних архітектур. Іншими словами, певний етап обробки даних може виконувати певна кількість систем. При надходженні даних в приграничну підсистему – вона обробляється і далі переміщується до наступної системи. Наприклад, при використанні цибульної схеми, кожен шар може відповідати за власний етап обробки даних.

Виходячи з вищенаведених різновидів задачі організації покрокової обробки даних, цю задачу можна охарактеризувати як модифікацію квест-методу, де таких потоків даних може бути кілька. При цьому ці порції даних характеризуються часом аналізу у вузлі оверлейної мережі. А метою є мінімізація часу такого аналізу даних мережі.

Таким чином, мета даної атестаційної роботи полягає в модифікації квест-методу аналізу оверлейних мереж з динамічною структурою та невизначеною кількістю оверлеїв для оцінки рівня анонімності в мережі.

1 ОСОБЛИВОСТІ ПОБУДОВИ ДИНАМІЧНИХ ОВЕРЛЕЙНИХ МЕРЕЖ

1.1 Принципи побудови та функціонування оверлейних мереж

Сучасні мережні технології – це собою складний набір взаємозв’язаних протоколів, інтерфейсів і алгоритмів взаємодії різних пристроїв, правильне спільне застосування яких дозволяє вирішувати широке коло завдань щодо забезпечення інформаційного обміну. Топології мереж, сформовані в результаті тривалого історичного розвитку шляхом нашарування різних технологій, відрізняються складністю і різноманітністю [1].

Взаємодію вузлів комп’ютерної мережі прийнято описувати за допомогою багаторівневих моделей, таких як OSI (Open System Interconnection – модель взаємодії відкритих систем). [2] Дані моделі описують узгоджений набір протоколів, званий стеком протоколів, спільне застосування яких забезпечує передачу як призначених для користувача даних, так і службових мережних повідомлень, необхідних для належного функціонування мережі.

Найважливішим протоколом стека TCP/IP є протокол IP (Internet Protocol), на якому побудовано все взаємодія як в інтернеті, так і в локальних мережах. Він не був розрахований на роботу в мережах такого масштабу, який має сьогодні Інтернет, і на певному етапі розвитку зіткнувся з наступними проблемами:

- лавиноподібне зростання складності маршрутизації великого числа мереж;
- залежність адреси від провайдера, складність масового зміни адрес;
- вичерпання IP-адрес.

Для вирішення останньої проблеми найбільш ефективним, і як наслідок найбільш поширеним, засобом є трансляція мережевих адрес. Дана технологія дозволяє замінювати адреси великого числа комп’ютерів в

локальній мережі на одну адресу шлюзу у зовнішній мережі, якої, як правило, є інтернет. Крім економії IP-адрес, використання NAT також призводить до підвищення безпеки за рахунок приховування інфраструктури внутрішньої мережі, однак дана технологія має важливий недолік – внутрішня мережа, яка перебуває за пристроєм NAT, виявляється ізольованою від зовнішніх з'єднань і повністю «невидима» з Інтернету. Така ситуація є неприйнятною для територіально розподілених організацій, чиї інформаційні ресурси розосереджені по декількох внутрішніх мережах, тому для забезпечення зв'язку між ними створюють оверлейні мережі, в основі яких лежать дві технології: шифрування і тунелювання.

Завдання тунелювання – процес, в ході якого створюється логічне з'єднання між двома кінцевими точками за допомогою інкапсуляції різних протоколів. Інкапсуляція – це процес передачі даних з верхнього рівня додатків вниз по стеку протоколів до фізичного рівня. При просуванні пакету даних за рівнями зверху вниз кожен новий рівень додає до пакету свою службову інформацію у вигляді відповідних заголовків.

В процесі інкапсуляції, виконуваної при тунелюванні, виділяють наступні типи протоколів [5]:

- транспортний протокол, тобто той, чиї дані потрібно передати через тунель;

- протокол інкапсуляції, який виконує службові функції передачі параметрів тунелю і позначення самого факту тунелювання;

- несе протокол, призначений для передачі всієї перерахованої вище інформації з питань зовнішньої мережі.

Від звичайних багаторівневих мережевих моделей (як OSI або TCP/IP) туннелирование в оверлейних мережах відрізняється тим, що транспортний протокол відноситься до того ж або більш низького рівня, ніж використовуваний в якості тунелю протокол (рисунок 1.1).

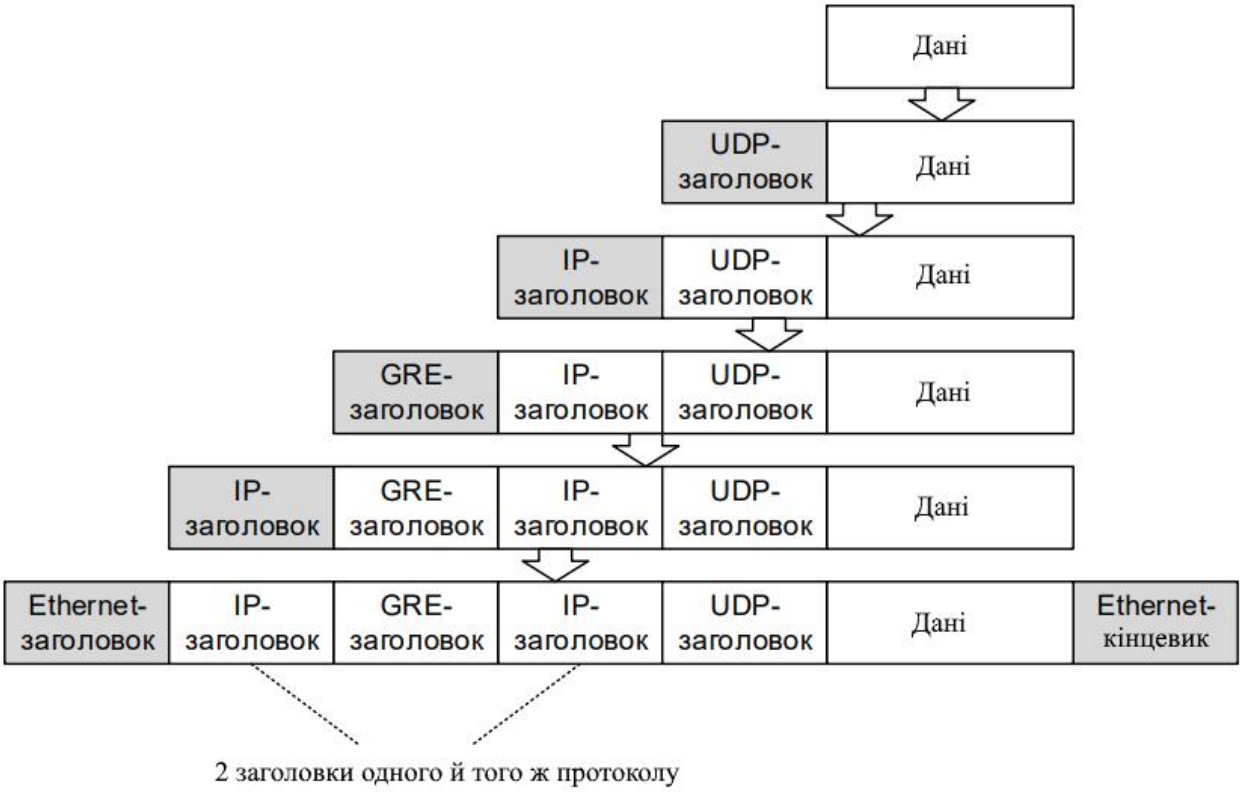

Рисунок 1.1 – Інкапсуляція пакетів при тунелюванні

На рисунку 1.1 транспортними протоколами є UDP і IP, протоколом інкапсуляції – GRE, а несучим протоколом, також як і транспортним, – IP. Описаний підхід є стандартним, внаслідок чого легко розпізнається системами аналізу трафіку, а в разі відсутності додаткового захисту піддається подальшому аналізу для вилучення корисної інформації. У зв’язку з цим перспективним може виявитися інший підхід, заснований на принципах мережної стеганографії – сукупності методів прихованої передачі даних з використанням особливостей роботи мережних протоколів [6].

Всі мережні протоколи умовно можна розділити на дві групи: протоколи, призначені для передачі корисного навантаження (як правило призначених для користувача даних) і службові протоколи, використовувані тільки для обміну інформацією про стан мережі, її топології, зіставлення адрес, а також виконують велику кількість інших функцій, безпосередньо не пов’язаних з передачею даних користувача, але необхідних для коректної роботи мережі. До першої групи належать всі протоколи прикладного рівня,

а також протоколи TCP, UDP, IP і Ethernet, до другої – ICMP, ARP, DHCP, DNS, всі протоколи маршрутизації і також багато інших менш поширених протоколів.

Як правило, під час налаштування систем аналізу трафіку, для службових протоколів використовують менш строгі правила ніж для транспортних, а іноді і зовсім ігнорують їх, що пов’язано з економією обчислювальних ресурсів, а також уявленням про те, що в службових протоколу не буде призначених для користувача даних, що представляють інтерес для аналізу [7].

Найбільш простим для інкапсуляції оверлейних даних є протокол ICMP, в пакетах якого передбачено місце для розміщення будь-яких даних, безпосередньо не використовуваних самим протоколом [8, 9].

Іншим поширеним службовим протоколом є ARP (Address Resolution Protocol), призначений для зіставлення IP і MAC адрес. Його пакети часто передаються по мережі, не мають чітких ознак, що дозволяють відрізнити реальний пакет від підробленого і не блокуються засобами захисту мереж. Не зважаючи на те, що в даному протоколі не передбачено окреме поле для записі даних, можна використовувати для цих цілей поля його заголовка, призначені для одного з адрес відправника.

У разі, коли в мережі застосовується динамічна маршрутизація, пакети використовуваного протоколу маршрутизації можна застосовувати для транспортування повідомлень корисувачів. Такі пакети містять велику кількість даних, наприклад, про топології мережі, для протоколу ICMP, або про вже наявних маршрутах, для протоколу RIP. Довжина цих даних не обмежена, а їх зміст не може бути перевірено проміжним вузлом, що здійснює аналіз трафіку, в результаті чого можливе додавання в пакет значних обсягів даних не є маршрутної інформацією.

Незалежно від обраного несучого протоколу, використання принципів стеганографії для впровадження даних в службові мережеві пакети зводиться до визначення потенційно надлишкової інформації в структурі пакета, її

компресії і записи в місце, що вивільнилося, необхідної інформації [10]. Даний принцип показаний на рисунку 1.2.

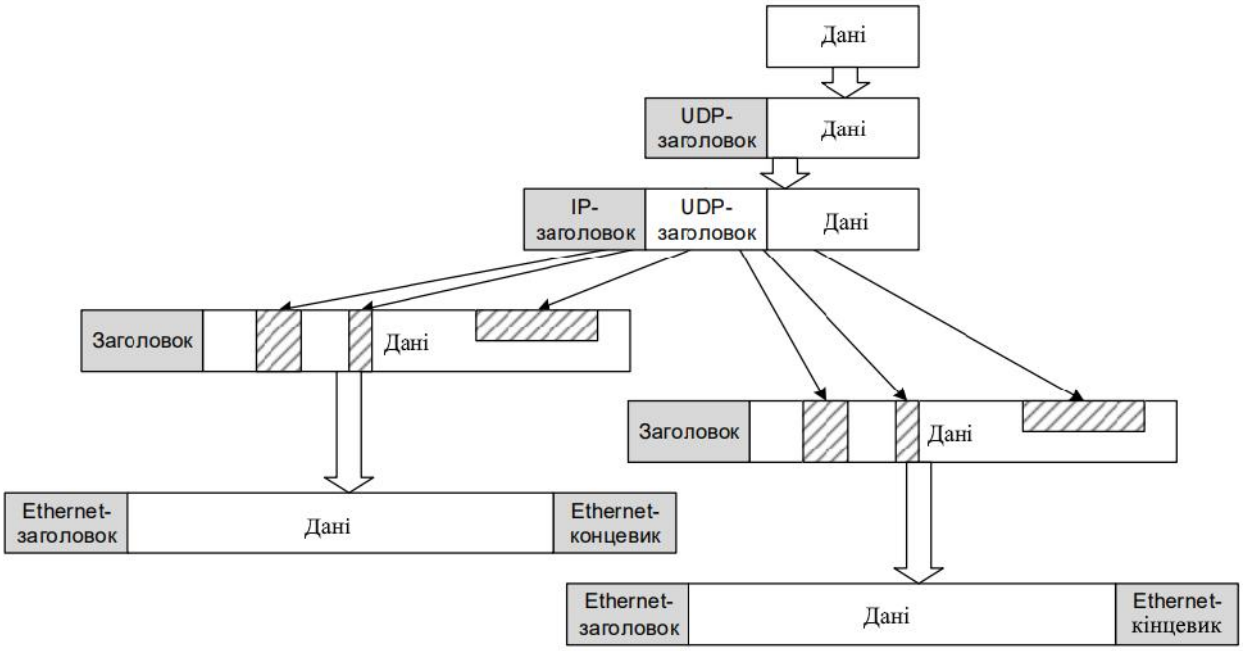

Рисунок 1.2 – Стенографічна інкапсуляція даних в оверлейній мережі

Ще одним різновидом оверлейних мереж є програмно-реконфігуровані мережі. Поняття ПКМ – це загальна концепція; конкретні реалізації ПКМ можуть значно відрізнятися за своєю архітектурою. Можна виділити три напрямки реалізацій ПКМ:

- ПКМ, створювані на основі первісної, оригінальної версії, запропонованої фахівцями Стенфордського університету (США) в 2007 р .;
- ПКМ, створювані на основі існуючих API;
- ПКМ, створювані на основі оверлейних мереж і гіпервізора.

Перший напрямок, тобто реалізація ПКМ в оригінальній версії, передбачає переміщення функціоналу управління мережею від мережних пристроїв до централізованого контролера з використанням протоколу OpenFlow [7]. Така ПКС повинна володіти наступними п’ятьма фундаментальними властивостями: мати розділені рівні управління, використовувати більш прості мережеві пристрої, здійснювати

централізоване управління мережею, використовувати автоматизацію і віртуалізацію втратити зв’язок із мережею, бути відкритою для фахівців, розробників. З’явився спеціальний термін – «Open SDN» для мереж, що мають перераховані властивості [8]. Обов’язковим елементом такої ПКМ є наявність контролера, що працює з мережею через інтерфейс OpenFlow.

У другому напрямку при створенні ПКМ використовуються функції API, які можуть бути викликані віддалено, як правило, за допомогою традиційних механізмів, таких як SNMP, CLI або при використанні більш нових, гнучких механізмів, типу RESTful API.

Реалізація ПКМ в третьому випадку не залежить від мережної інфраструктури: ПКМ накладається поверх існуючої фізичної мережі. І другий, і третій варіанти створення ПКМ є альтернативними варіантами щодо Open SDN, але в деяких випадках є кращими при створенні оверлейних комп’ютерних мереж.

Розглянемо доцільність вибору того чи іншого варіанту ПКМ при створенні оверлейной мережі.

Завдяки зняттю з комутаторів завдань управління просуванням даних відбувається прискорення переміщення трафіку, що істотно підвищує продуктивність. При цьому за рахунок віртуалізації управління мережею знижуються витрати на їх побудову і супровід. На централізованому контролері Open SDN системний адміністратор може спостерігати всю мережу в єдиному поданні, за рахунок чого підвищуються зручність управління, безпека і спрощується виконання ряд інших завдань. Теоретично необмежені можливості мереж Open SDN до розширення дозволяють будувати реальні хмари, масштабовані в залежності від розв’язуваних завдань. При цьому мережа володіє необхідною «інтелектуальністю», необхідної, зокрема, для оркестровки роботи великих груп комутаторів [9].

Але незабаром з’явилося чимало критики, пов’язаної і з архітектурою Open SDN, і з питаннями практичного впровадження таких мереж. Найбільш гучними критиками є виробники мережного обладнання, які стурбовані

занадто радикальними, на їх погляд, змінами, які несе з собою нова мережна технологія. Це і значна вартість нового обладнання, тобто комутаторів з підтримкою інтерфейсу OpenFlow, і ризики, що виникають із-за недостатнього тестування нового обладнання, яке повинно в масовому порядку впроваджуватися у клієнтів.

Найбільш серйозним аргументом критиків Open SDN з точки зору її архітектури є вразливість такої мережі через наявність в ній єдиної точки відмови. На рисунку 1.3 показана схема просування даних в мережі Open SDN, керованої єдиним контролером. При виході з ладу цього контролера відбувається втрата працездатності всієї мережі. Тобто зображений контролер ПКМ є єдиною точкою відмови.

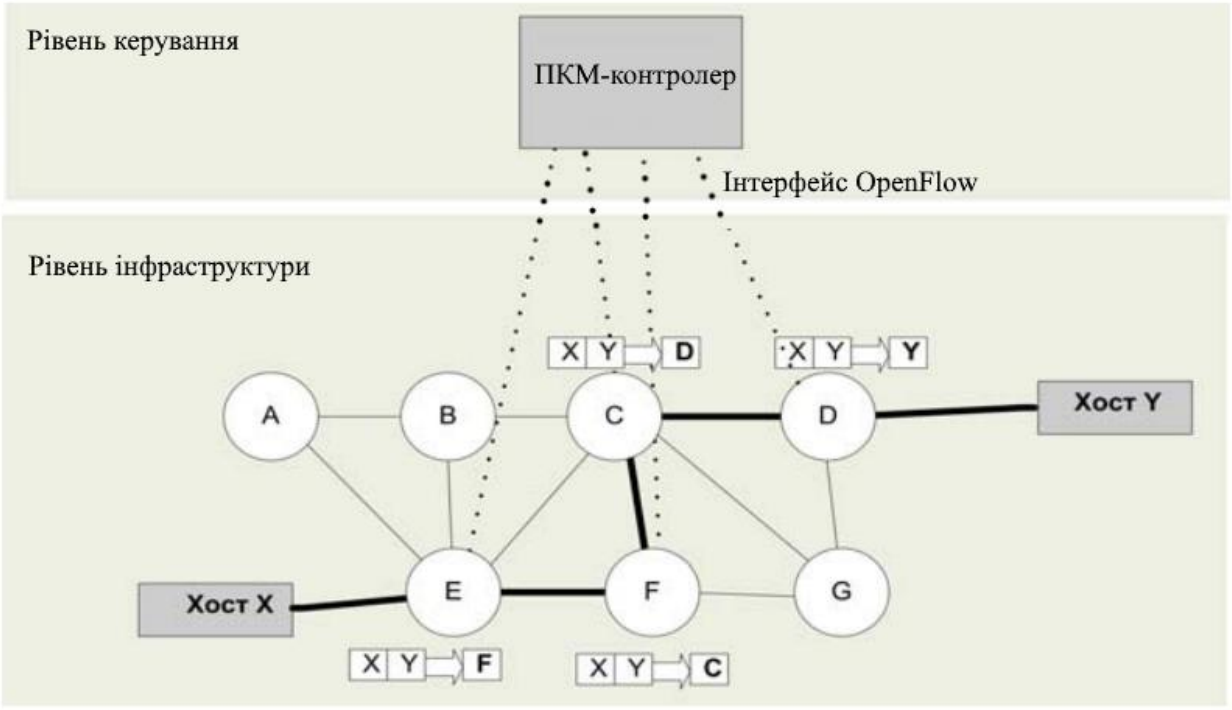

Рисунок 1.3 – Переміщення даних в мережі згідно архітектури OpenSDN

Щоб подолати зазначений недолік, необхідно мати в мережі кілька контролерів, взаємодіючих між собою за допомогою особливо надійних ліній зв’язку. Це ускладнює і здорожує архітектуру Open SDN.

Далі розглянемо ПКМ, створюване на основі існуючих API. Якщо забезпечити мережні пристрої можливістю розпізнавання більш широкого набору команд API, за допомогою яких контролер зможе гнучко управляти пристроями і всієї мережею, то це і буде ПКМ, реалізованою через існуючі API (рисунок 1.4).

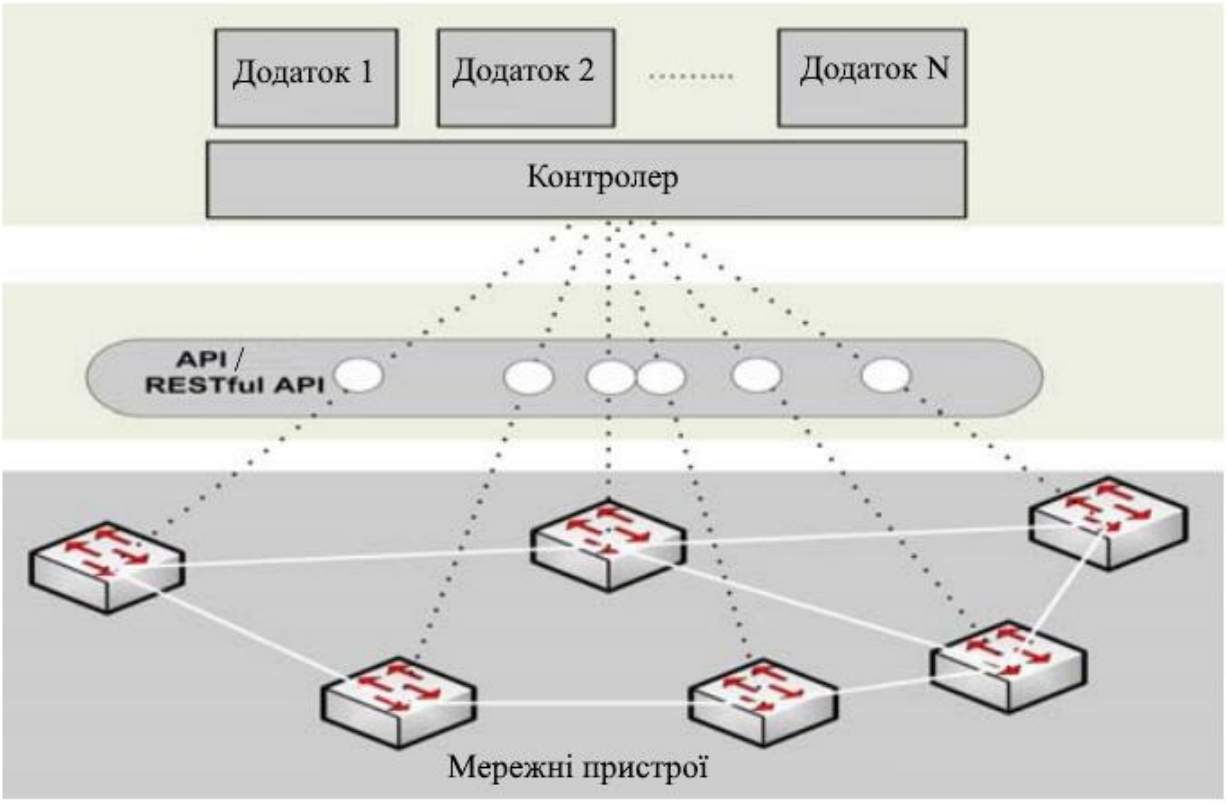

Рисунок 1.4 – ПКМ, створена на основі інсуючих API

Для створення ПКМ за такою схемою деякі виробники обладнання модернізують існуючі API на пристроях. Наприклад, замість традиційних CLI і SNMP впроваджується RESTful API [10]. Механізми CLI і SNMP давно розроблені і використовуються при виконанні мережних налаштувань. Але в даний час, коли необхідно оперативно, динамічно управляти великою мережею або центром обробки даних, ці механізми є занадто громіздкими і незручними. Їм на зміну прийшов новий механізм - RESTful API. В останні роки цей механізм став найбільш поширеним при передачі API-запитів по мережі. Технологія RESTfull API працює з використанням протоколу

передачі гіпертексту HTTP, який в загальному випадку використовується для передачі WEB-трафіку. Технологія RESTful API є відносно простою і легко розширюється. Вона зручна в застосуванні, так як використовує в роботі стандартний TCP-порт, що не вимагає спеціальних налаштувань брандмауера для проходження через нього API-запитів. У ПКМ, створених на основі існуючих API, є ряд переваг. Одним з очевидних переваг є те, що вони працюють зі звичайними, не модернізованими комутаторами. Тобто не потрібно впровадження комутаторів нового типу, з підтримкою стандарту OpenFlow.

Ще одна перевага такого підходу полягає в тому, що в певній мірі підвищується гнучкість управління мережею. Наявні API спрощують написання програмного забезпечення для оркестровки подій в мережі, тобто для швидкого і автоматичного реагування на зміни в мережі, таких, як динамічні переміщення віртуальних машин в центрах обробки даних. Наступна перевага полягає в тому, що використання наявних API дозволяє побудувати мережу з централізованим в певних межах управлінням. Це веде до більшої відкритості в мережевих архітектурах, так як виробники змушені відкривати специфікації інтерфейсів свого фірмового обладнання. Останнє необхідно для розробки і нормальної експлуатації додатків сторонніми розробниками.

ПКМ, створювані на основі існуючих API, мають і свої недоліки. Перш за все, в таких мережах в більшості випадків контролер зовсім відсутній. Тому мережний програміст змушений програмувати безпосередньо кожен комутатор. Але якщо навіть є контролер, у програміста відсутный загальний, стандартний механізм взаємодії з пристроями мережі. Тобто програміст змушений знати технічні характеристики інтерфейсу кожного комутатора. Ще одним недоліком є те, що програмне забезпечення такої ПКМ буде працездатний тільки для конкретної конфігурації мережі. Це і зрозуміло, так як API пристроїв різних виробників не підпадають під загальний стандарт (на противагу протоколу OpenFlow). Тому ПКМ такого типу зможе працювати з

обладнанням певного виробника або невеликої групи виробників сумісного обладнання.

Наступний недолік пов’язаний з тим, що наказують в архітектурі ПКМ переміщення функцій управління від комутатора до контролера мало на меті створення простих, менш дорогих комутаторів. Цю задачу вирішити не вдалося: ПКС створюються на основі тих же складних і дорогих комутаторів. Нарешті, хоча ПКС, створені на основі існуючих API, і дозволяють в певній мірі керувати пересиланням даних, зокрема, при побудові VLAN і VPN, але вони не можуть забезпечити більш тонкий контроль кожного інформаційного потоку, як це можливо при використанні протоколу OpenFlow.

ПКМ, створені на основі існуючих API, можна розглядати в якості практичного розширення поточної функціональності оверлейної комп’ютерної мережі, коли більш радикальне рішення на основі OpenFlow ще не створено або недоцільне з яких-небудь причин. В цілому, створення ПКМ на основі існуючих API є кроком у правильному напрямку, рухом в напрямку створення оверлейної мережі з повністю централізованим програмним управлінням.

Розглянемо ПКМ, створювані на основі віртуальних мереж і гіпервізора. Альтернативним і більш інноваційним способом створення ПКМ є використання накладених мереж з керуванням через гіпервізор [11]. У тому випадку фізична мережа залишається як є, без змін. Але над цією фізичною мережею створюються накладені віртуальні мережі з гіпервізорами. Прикладні системи на вузлах мережі взаємодіють з цими віртуальними мережами, нічого не знаючи про фізичні характеристики мережі, через яку відбувається передача даних (малюнок 1.5).

Оскільки віртуальні мережі знаходяться над фізичною інфраструктурою, ними можуть керувати системи (або пристрої), розміщені на кінцевих вузлах мережі. У центрах обробки даних такими системами є гіпервізор віртуальних машин, які присутні на кожному сервері. Передача трафіку в віртуальних мережах здійснюється за допомогою тунелювання, з

використанням інкапсуляції. Тобто при надходженні пакета на вузол віртуальної мережі для передачі мережеве пристрій (зазвичай це гіпервизор) інкапсулює цей пакет в інший кадр. ПКМ, створені на основі накладених мереж і гіпервізора, добре працюють в ЦОД, в яких вже встановлено програмне забезпечення для серверної віртуалізації. Вони допомагають усунути ряд проблем, що виникають при роботі ЦОД.

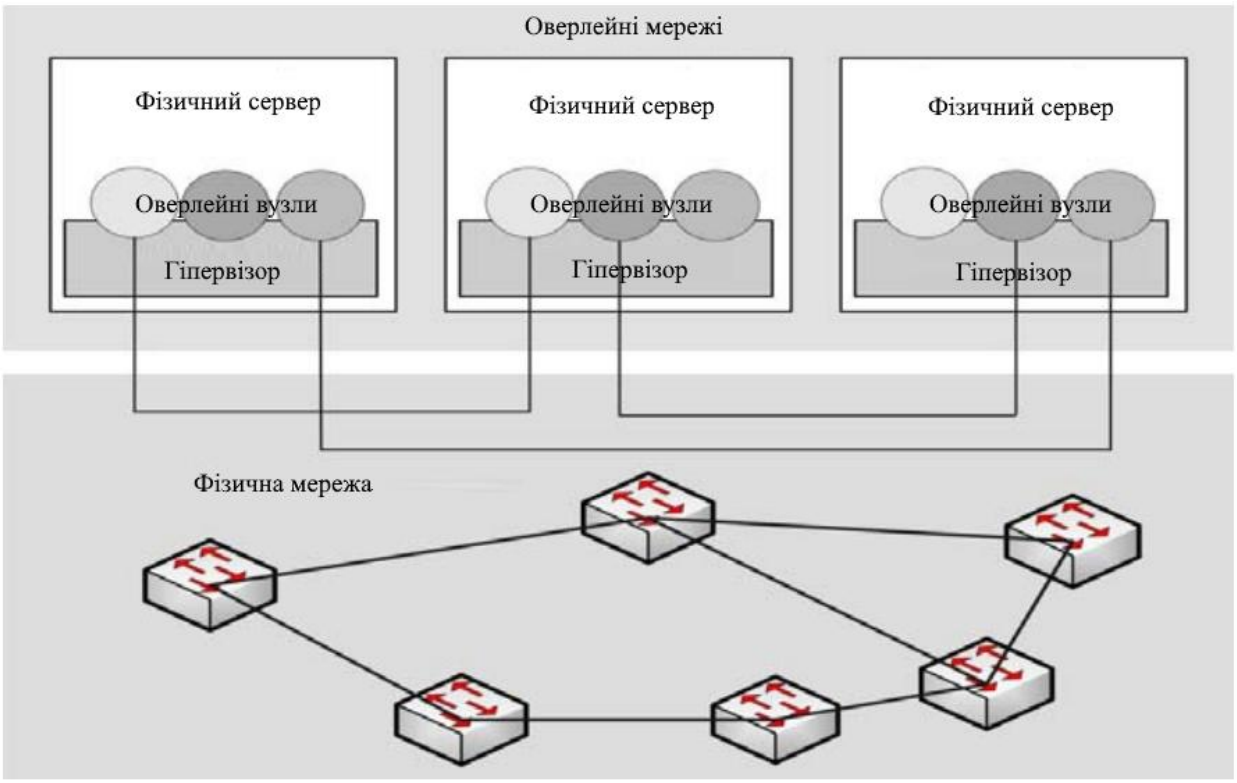

Рисунок 1.5 – Віртуальні мережі, накладені на фізичну мережу (оверлейна інфраструктура)

По-перше, усувається вибухове зростання MAC-адрес вузлів, так як тепер, при використанні ПКМ, MAC-адреси приховані в інкапсульованих кадрах. По-друге, відомі обмеження на кількість підтримуваних VLAN в локальних мережах вже не мають значення, так як тут використовується тунелювання (замість VLAN) для поділу трафіку численних інформаційних потоків. По-третє, побудова ЦОД на основі ПКМ дозволяє дуже гнучко і оперативно змінювати характеристики мереж, що беруть участь в

обчислювальних процесах ЦОД, за рахунок можливості централізованого програмного управління віртуальними мережами ПКМ.

ПКМ, створені на основі накладених мереж і гіпервізора, не вирішують всі проблеми. Зокрема, фізична мережева інфраструктура все так же вимагає ручного налаштування і обслуговування. Це стосується, наприклад, протоколів QoS, STP. Іншим недоліком є те, що в таких ПКС, як і в попередньому варіанті, мережні пристрої залишаються незмінними, у той час як оверлейна мережа може бути динамічною, зі змінною кількістю вузлів.

1.2 Особливості динамічних оверлейних мереж

Динамічна комп’ютерна мережа - це така мережа, яка характерізується постійною Перебудова маршутів передачі даних через включення або віключення з неї учасников передачі даних. Прикладом таких мереж є MANET, FANET тощо.

Організація мережевої структури в умовах високої інтенсивності руху її учасників є специфічним і важкоздійснюваним завданням. Функціонування мережних структур з багаторівневою ієрархією націлене на забезпечення обміну великими обсягами даних зі стаціонарними вузлами, або вузлами з невеликою швидкістю пересування – є різновидом самоорганізуючих мереж передачі даних, зі змінною топологією і які не мають постійної структури і призначені для зв’язку між рухомими об’єктами.

Кожен учасник такої мережі має бути як передавачем власних повідомлень, так і ретранслятором інформації від інших вузлів. Таким чином, вузли взаємодіють в межах мережного середовища, якщо ж потрібно встановити обмін даними поза цим радіуса, то створюються маршрути передачі даних.

Якщо на таку мережу накладається віртуальна (оверлейна) мережа, то організація надійної маршрутизації забезпечується спеціалізованими алгоритмами маршрутизації. У свою чергу, протоколи маршрутизації в

динамічних оверлейних мережах класифікуються наступним чином:

- за запитом (встановлення маршруту відбувається за "бажанням" вузла);

- табличні (маршрути постійно оновлюються і зберігаються на кожному вузлі)

- гібридні (об’єднання технік за запитом і табличних),

- багатошляхові (створюється множина маршрутів для підвищення надійності передачі інформації) [14].

Найчастіше, для забезпечення функціональних вимог, що покладаються на оверлейні мережі, розглядаються спеціальні протоколи маршрутизації, тому що вузли в мережі ведуть постійний обмін даними про своє місцезнаходження і напрямок руху.

Для забезпечення підвищення рівня безпеки передачі даних, а також забезпечення своєчасної ситуаційної обізнаності інформація від вузла до вузла повинна бути доставлена негайно. До того ж, система повинна постійно підлаштовуватися в умовах постійної зміни конфігурації вузлів. Рівень контролю доступу до середовища призначений для організації багатостанційного доступу, тобто забезпечення ефективного розподілу канального ресурсу між вузлами, охочими вести передачу даних. MAC рівень відповідає канальному рівню еталонної моделі взаємодії відкритих систем OSI. Техніка, яка використовується на рівні багатостанційного доступу, є найважливішим аспектом при проектуванні і розробці комп’ютерної системи, тому потрібно виділити основні властивості протоколів і визначити їх важливість для систем ІТС:

- уникнення колізій – основна властивість цих протоколів. Протокол повинен надавати можливість уникати виникнення ситуацій з одночасною передачею інформації;

- адаптивність – протокол повинен підлаштовуватися під зміни в системі. Цей параметр важливий, тому що динаміка мережі приводить до перебудови топології мережі;

- пропускна здатність – відсоток успішно прийнятих пакетів по каналу зв’язку. Важливий параметр, тому що пропускна здатність повинна зберігатися при збільшенні числа користувачів;

- затримка – в даному випадку, час між спробою отримання доступу до каналу і моментом отримання доступу. Як вказувалося вище – функціонування мережі оверлейної мережі є чутливим до часу доставки інформації.

- ефективність використання каналу – важливий параметр, що впливає на показники затримок і пропускної здатності.

У стандарті IEEE 802.11p використовується метод множинного доступу із контролем несучої і униканням колізій (CSMA/CA). Цей стандарт розроблявся спеціально для забезпечення зв’язку між мобільними вузлами та підходить при побудові оверлейної мережі над існуючою. Протокол CSMA досить поширений і не вимагає жорсткої синхронізації між вузлами, тому дуже часто застосовується при розробці систем, що самоорганізуються – ad hoc і сенсорних мереж. CSMA в стандарті 802.11p реалізований наступним чином:

- кожен вузол при включенні прослуховує канал зумовлений період, для міжкадрового вирішення конфліктів – TAIFS;

- якщо підключення до найближчого вузла є або з’являється під час періоду прослуховування, то вузол починає процедуру відстрочки передачі - backoff (на випадковий період часу);

- якщо підключень немає, то вузол здійснює передачу по закінченню періоду прослуховування.

Параметр backoff визначається наступним чином: спочатку проводиться вибір випадковим чином цілого числа з інтервалу [0; CW] (CW - contention window, параметр впливає на рівень конкуренції в каналі), отримане число перемножується на час тривалості слоту Tslot. При відстроченні передачі процедура прослуховування повторюється знову і знову. Якщо при прослуховуванні стан каналу ідентифікується, як вільний,

то параметр backoff зменшується на час тривалості слота. По досягненню backoff = 0, вузол веде передачу. Тоді починають вибудовуватися віртуальні канали оверлейної мережі.

Збільшення параметра CW призводить до погіршення ефективності використання каналу, але зменшує число колізій. Відомо, що при CSMA, коли число учасників руху велике, можуть виникати колізії між поруч розташованими вузлами. При завантаженості мережі в 100 % частка втрачених пакетів становить 30 %. Ці фактори мають великий вплив на ефективність роботи 802.11p для додатків безпеки, тому в якості альтернативного методу забезпечення багатостанційного доступу до середовища розглядається система з тимчасовим поділом каналу – STDMA.

Self-organized time division multiple access [9] – це детермінований протокол багатостанційного доступу до середовища без конкуренції. У STDMA канальний час ділиться на кадри і тимчасові слоти, тривалість яких визначається реалізацією системи. Кожен вузол, що використовує певний алгоритм, має можливість самому вибрати слот, в якому він буде вести передачу. Це робить звязки в оверлейній мережі, що вибудовується більш стійкі. Мається на увазі, що в кожному слоті проводиться передача одного повідомлення в якому є навігаційна інформація та інформація для управління оверлейної мережі.

1.3 Постановка задачі

Таким чином, з метою підвищення надійності функціонування оверлейної мережі, надбудованої над динамічно змінюваній комп’ютерній мережі, в якій кількість вузлів є нестабільним числом, необхідно модифікувати існуючий квест-метод аналізу мережі [17] та провести відповідні дослідження ефективності його застосування з метою забезпечення анонімізації користувачів мережі.

2 МЕТОДИ АНАЛІЗУ ОВЕРЛЕЙНИХ МЕРЕЖ З НЕВИЗНАЧЕНОЮ КІЛЬКІСТЮ ОВЕРЛЕЇВ

2.1 Контурний метод аналізу оверлейної мережі

Вихідними даними для аналізу оверлейної мережі служить топологія мережі провайдера, матриця запитів між окремими вузлами, потоки між ними, як правило, задаються у вигляді матриці запитів, що становить собою матрицю розмірності *DxD*, де елемент d_{ij} показує інтенсивність потоку від *i*-го вузла до *j*-го вузла однієї мережі.

Характеристиками, що аналізуються, є потоки трафіку і завантаження каналів зв’язку при обмеженнях параметрів якості обслуговування. В результаті аналізу будуть запропоновані маршрути проходження трафіку між кожною парою оверлеїв. Нехай топологія мережі представлена у вигляді орієнтованого графа *G(N,A)*, де кожне ребро графа визначає односпрямований канал зв’язку.

Основною ідеєю контурного методу є те, що всі топології, що містять однакове число гілок, пов’язані між собою тензором перетворення, в ролі якого можуть виступати матриця лінійно незалежних розрізів, або матриця лінійно незалежних контурів, або об’єднана матриця лінійно незалежних контурів і розрізів. Оскільки всі мережі з топологією, що складається з однакового числа гілок, пов’язані між собою тензором перетворення, то серед множини проекцій можна виділити так звану примітивну мережу [3].

Примітивна мережа складається з такої ж кількості гілок, як і досліджувана мережа, але кількість непов’язаних компонент в ній також рівне числу гілок, в зв’язку з чим потоки в кожній гілці примітивної мережі виявляються незалежними.

Якщо визначити математичну модель найпростішого елемента мережі, яким є одноканальна схема, то згідно постулату узагальнення Крона

математична модель примітивної мережі має тривіальний вид і показує зв’язок контурних інтенсивностей з контурними часом обслуговування і контурними завантаженнями в матричної формі:

$$\begin{bmatrix} \rho_1 \\ \rho_2 \\ \vdots \\ \rho_n \end{bmatrix} = \begin{bmatrix} t_1 & 0 & \cdots & 0 \\ 0 & t_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & t_n \end{bmatrix} \times \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} \quad (2.1)$$

де: $\lambda_i(i=1...n)$ – інтенсивність надходження, що надходить на вхід, елемента мережі;

$t_i(I=1...n)$ – середній час обслуговування;

$\rho_i(i=1...n)$ – завантаження i-го елемента.

Якщо в якості базисних елементів для кожної топології будуть використовуватися лінійно незалежні контури, то тензор перетворення встановлює зв’язок між контурними інтенсивностями однієї мережі з контурними інтенсивностями іншої мережі. Так, зв’язок контурних інтенсивностей примітивної мережі пов’язана з контурними интенсивностями досліджуваної мережі тензором перетворення.

Кількість контурних інтенсивностей аналізованої мережі визначається цикломатичним числом, рівним: $p=n-m+1$.

Тензор перетворення C визначається як матриця лінійно незалежних контурів. Є різні методи отримання матриці контурів: граф-комбінаторні і аналітичний. Граф-комбінаторні методи засновані на таких алгоритмах, як пошук в глибину і пошук в ширину. Слід зазначити, що складність таких алгоритмів дорівнює $O(N)$, тобто зростає лінійно зі збільшенням числа гілок мережі. Аналітичний спосіб припускає знаходження матриці контурів як матрицю, ортогональную матриці інцидентності, так як матриця інцидентності є також однією з множин матриць розрізів, але з одним лінійно залежним рядком. Іншими словами, добуток матриці інцидентності без

одного довільного рядка на матрицю лінійно незалежних контурів дає матрицю з нульовими елементами. Використовуючи це правило, можна знайти матрицю лінійно незалежних контурів.

У загальному випадку значення елементів матриці контурів не має значення, головне, щоб контури, що знаходяться в кожному стовпці, були лінійно незалежними. Але для розрахунків зручним виявляється використання матриці не просто з лінійно незалежними контурами, а з фундаментальними циклами, саме така матриця і обрана в якості тензора перетворення.

Саме матриці фундаментальних циклів, на відміну від просто матриць лінійно незалежних контурів, явно показують, що якщо задані потоки в контурах примітивної мережі, номери яких відповідають номерам рядків, в яких всі елементи крім одного дорівнюють нулю, то після перетворення примітивної мережі в досліджувану потоки в останніх гілках будуть однозначно визначені як лінійна комбінація заданих потоків. Якщо ж вибрати довільну матрицю лінійно незалежних контурів, то такий зв’язок буде неочевидний.

Зв’язок контурних інтенсивностей примітивної мережі з досліджуваною можна, як уже було сказано, записати наступним чином:

$$\lambda^{\tilde{i}} = C_i^{\tilde{i}} \lambda^i, \qquad (2.2)$$

де $\lambda^{\tilde{i}}$ – контурні інтенсивності примітивної мережі;

$C_i^{\tilde{i}}$ – тензор перетворення контурних інтенсивностей досліджуваної мережі в контурні інтенсивності примітивної мережі;

λ^i – значення контурних інтенсивностей досліджуваної мережі.

За допомогою нескладних матричних перетворень можна показати, що вектор контурних завантажень досліджуваної мережі може бути записаний, як: $\rho_i = C_i^{\tilde{i}} \rho_{\tilde{i}}$ і матриця контурних тривалостей обслуговування досліджуваної

мережі визначена у вигляді:

$$t_{ji} = t_{\tilde{j}\tilde{i}} C_j^{\tilde{j}'} C_i^{\tilde{i}'}, \tag{2.3}$$

де *C'* – тензор перетворення без лінійно залежних рядків. На підставі отриманих значень контурних завантажень і контурних часів обслуговування значення контурних інтенсивностей досліджуваної мережі можна записати, як:

$$\lambda^i = \left(t_{ji}\right)^{-1} \rho_j . \tag{2.4}$$

А значення потоків в кожній гілці можна визначити наступним чином:

$$\lambda_{гілок}^{\tilde{i}} = C_i^{\tilde{i}} \lambda^i . \tag{2.5}$$

Оскільки для кожної оверлейної мережі відомі свої матриці запитів, проводячи тензорний аналіз характеристик для кожної оверлейної мережі окремо, тим самим отримавши стільки систем рівнянь типу (2.4), скільки існує оверлейних мереж, потім підсумувавши однойменні рядки всіх рівнянь, отримуємо фінальну систему рівнянь, що відображає залежність інтенсивностей потоків в кожному каналі зв'язку від завантажень створюваної кожної окремої мережі. Таким чином, якщо число мереж рівне *N*, то остаточна система рівнянь, що визначає потоки всіх оверлейних мереж в кожній гілці досліджуваної мережі, буде такою:

$$\lambda_{гілок_загальне}^{\tilde{i}} = C_i^{\tilde{i}} \sum_{q=1}^{N} \sum_{p=1}^{K} \lambda_{qp}^i \tag{2.6}$$

Система рівнянь (2.6) містить n рівнянь, де n – число гілок в аналізованій мережі, а число невідомих дорівнює $S \cdot p$, де S – загальна кількість оверлеїв всіх оверлейних мереж. Отже, система рівнянь (2.6) має нескінченне число рішень. Це пов’язано з тим, що кожне рішення системи рівнянь визначає будь-якої маршрут проходження трафіку між оверлеями мереж, а оскільки варіантів проходження маршрутів існує безліч, то і система рівнянь має нескінченну кількість рішень. Систему рівнянь (2.6) можна використовувати для аналізу трафіку двома способами.

Перший варіант випливає з того, що в системі (2.6) потоки в кожній гілці виражаються через лінійно незалежні завантаження, тому визначивши потоки/завантаження тільки в лінійно незалежних гілках, автоматично визначаються і потоки/завантаження в гілках, що залишилися.

Другий варіант використання системи рівнянь (2.6) полягає в знаходженні будь-якого рішення, а не в підборі значень лінійно незалежних компонент, при цьому отримане рішення буде описувати маршрути між оверлеями, але при такому підході система рівнянь (2.6) повинна вирішуватися з рядом обов’язкових обмежень в вигляді системи нерівностей.

Обов’язковими обмеженнями для системи проектування рівнянь (2.6) є: невід’ємність потоків створюваної окремим оверлеєм q-ї мережі, значення сумарного потоку в каналі зв’язку не повинно перевищувати величину пропускної здатності даного каналу, величина потоку в області n від p-го оверлею не повинна перевищувати величину потоку, що створюється р-м оверлеєм оверлейної мережі, а також необхідно забезпечити відсутність появи петльових маршрутів.

Додатковими обмеженнями, що накладаються на систему рівнянь (2.5), можуть бути обмеження наскрізної затримки або ймовірності втрат кожного типу трафіку, створюваного оверлеями віртуальних мереж.

Таким чином, вирішуючи систему рівнянь (2.6) з урахуванням невід’ємності потоків або іншими обмеженнями, що накладаються на якість обслуговування потоків оверлейних мереж, а також з огляду на значення

матриць запитів, можна однозначно визначати маршрути проходження трафіку між всіма оверлеями мереж. У свою чергу якщо рішення системи рівнянь (2.6) спільно з накладеними обмеженнями знайти не вдається, то це означає, що дана мережа характеризується наявністю невідомою кількістю оверлеїв.

Проаналізуємо чисельні результати. Як приклад проведемо аналіз оверлейної мережі, наведеної на рисунку 2.1. На рисунку 2.1 під умовним позначенням ВК показані вузли комутації, в ролі яких, як правило, виступають віртуальні маршрутизатори або комутатори. Згідно [14], математична модель досліджуваної мережі може бути представлена в вигляді мережі масового обслуговування (рисунок 2.3).

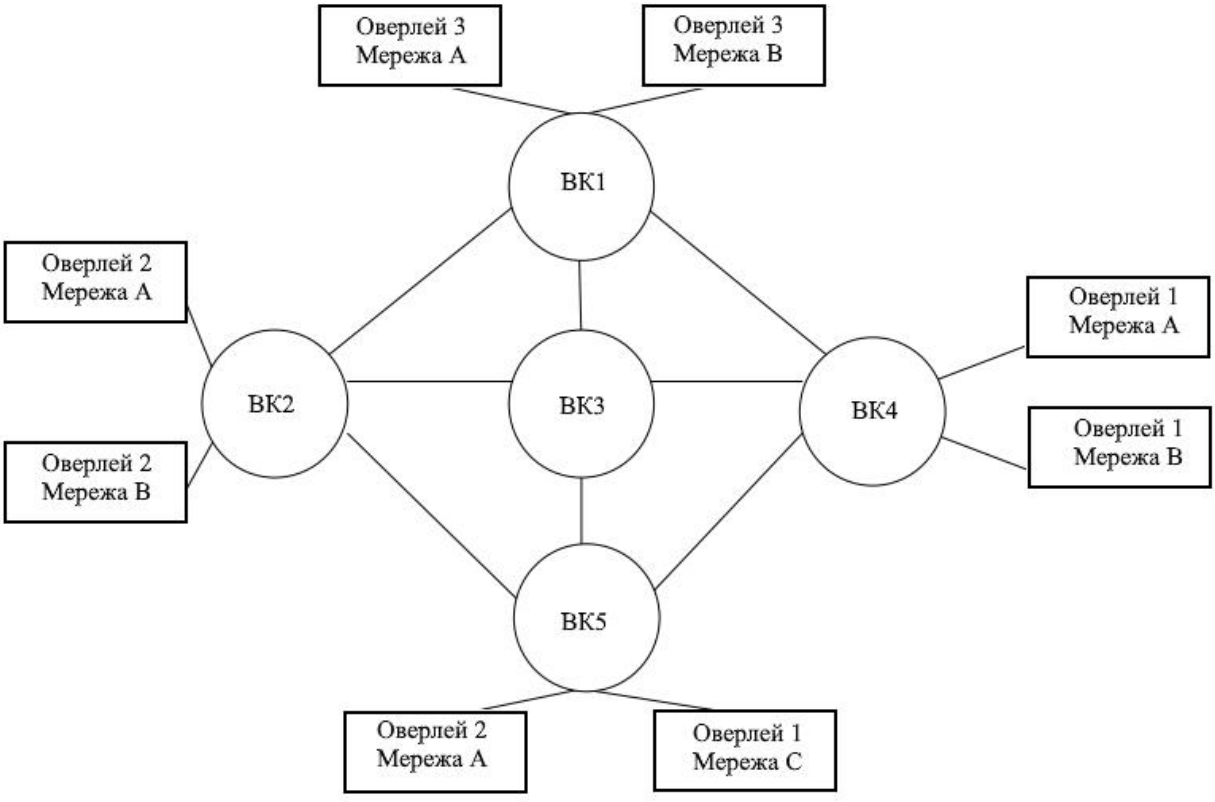

Рисунок 2.2. – Приклад організації оверлейної мережі

Відповідно до контурного методу аналізу, вузли 1’, 2’, 4’, 5’, 7 ’, 8’, 10’, 11’ об’єднуються в один вузол, після – визначається тензор перетворення від структури досліджуваної мережі до структури примітивної мережі, який

пов’язує контурні інтенсивності досліджуваної мережі з контурними интенсивностями примітивною. Потім по формулам (2.2) – (2.3) визначаються контурні завантаження і контурні часи обслуговування. Після чого визначаються контурні інтенсивності аналізованої мережі за формулою (2.4), потім за формулою (2.5) визначаються потоки в кожній гілці, створювані р-м вузлом q-ї мережі, потім по формулі (2.6) визначаються потоки в кожній гілці, створювані всіма оверлеями.

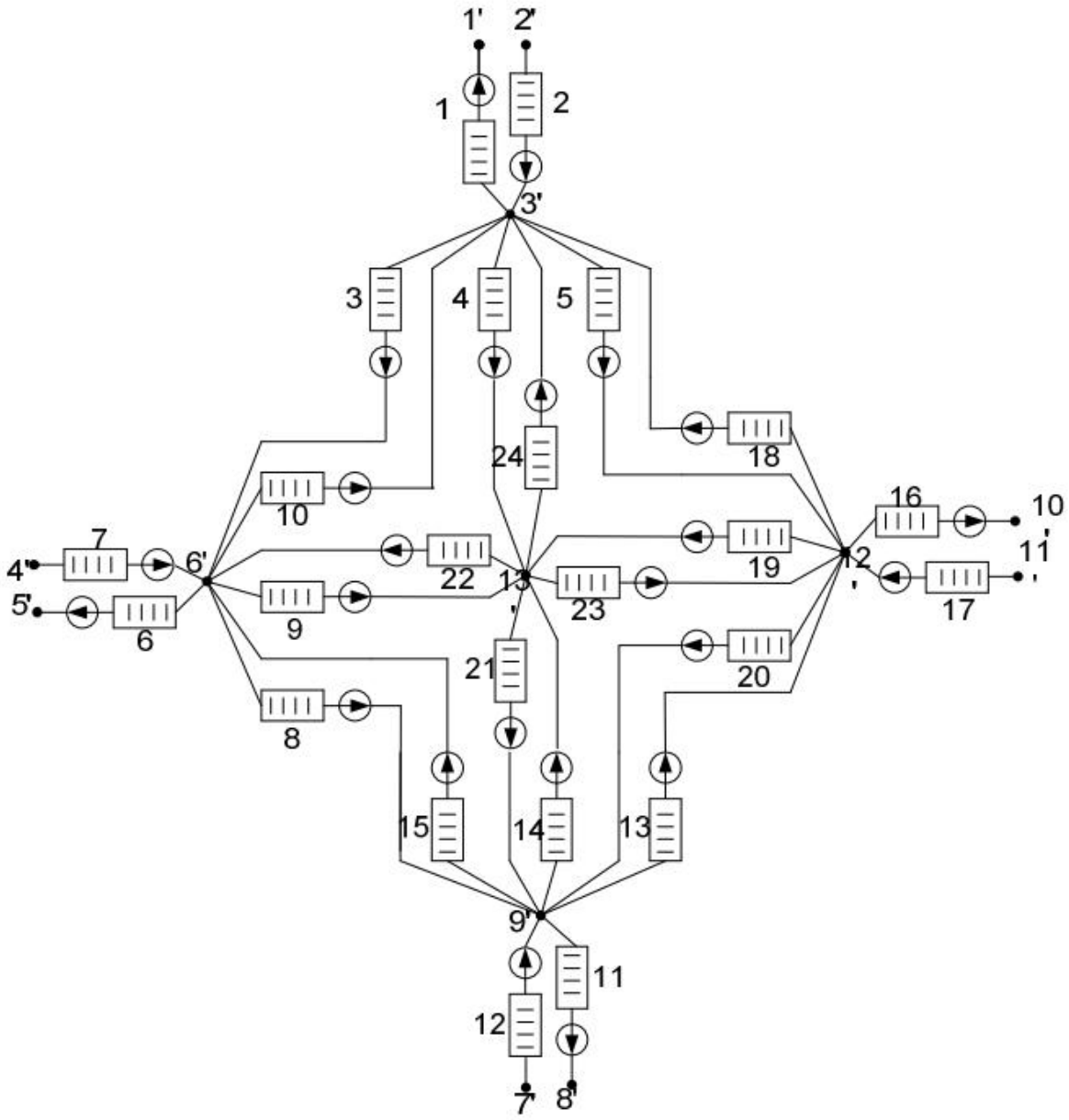

Рисунок 2.3 – Схема масового обслуговування для досліджуваної оверлейної мережі з невідомою кількістю оверлеїв

В якості вихідних даних визначені матриці запитів оверлеїв D_A, D_B, D_C у вигляді: $D_A = \begin{Vmatrix} 0 & 5 & 6 \\ 7 & 0 & 3 \\ 1 & 2 & 0 \end{Vmatrix}$, $D_B = \begin{Vmatrix} 0 & 8 & 6 \\ 10 & 0 & 3 \\ 15 & 9 & 0 \end{Vmatrix}$, $D_C = \begin{Vmatrix} 0 & 9 \\ 10 & 0 \end{Vmatrix}$. Елементи матриць показують, скільки одиниць інформації в секунду віддається від *i*-го оверлею до j-му в рамках однієї оверлейної мережі.

Так, елемент матриці D_A, що знаходиться в першому рядку і другому стовпці, показує, що з першого сайту в другий вузол мережі *A* передаватиметься 5 од.інф./с. час обслуговування однієї одиниці інформації приято як 0,01 с, що має на увазі швидкість 100 одиниць в секунду. При таких вихідних даних одним з рішень системи рівнянь (2.6) відповідає наступний розподіл:

$\Lambda_{1A} = [6\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 5\ 0\ 0\ 0\ 0\ 0\ 11\ 6\ 0\ 5\ 0\ 0\ 0\ 0]$
$\Lambda_{2A} = [3\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 10\ 7\ 3\ 0\ 7\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 3]$
$\Lambda_{3A} = [0\ 3\ 0\ 2\ 1\ 0\ 0\ 0\ 0\ 0\ 2\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 2\ 0\ 0\ 0]$
$\Lambda_{1B} = [6\ 0\ 0\ 0\ 0\ 8\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 14\ 6\ 8\ 0\ 0\ 8\ 0\ 0]$
$\Lambda_{2B} = [3\ 0\ 0\ 0\ 0\ 0\ 13\ 0\ 10\ 3\ 0\ 0\ 0\ 0\ 0\ 10\ 0\ 0\ 0\ 0\ 0\ 0\ 10\ 0]$
$\Lambda_{3B} = [0\ 24\ 15\ 0\ 9\ 15\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 9\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$
$\Lambda_{1C} = [0\ 0\ 0\ 0\ 0\ 0\ 10\ 10\ 0\ 0\ 10\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$
$\Lambda_{2C} = [0\ 0\ 0\ 0\ 0\ 9\ 0\ 0\ 0\ 0\ 0\ 9\ 0\ 0\ 9\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$

Кожен вектор показує, яка інтенсивність в гілках створюється *p*-м оверлеєм мережі_q. Сумарний вектор інтенсивностей для кожної гілки представлений в таблиці 2.1.

Таблиця 2.1 – Вектори інтенсивностей для гілок

Номер гілки	1	2	3	4	5	6	7	8	9	10	11	12
Інтенсивність входу	18	27	15	2	10	32	23	10	10	3	17	19
Номер гілки	1	2	3	4	5	6	7	8	9	10	11	12
Інтенсивність входу	7	3	9	27	25	12	8	5	2	8	10	3

Також при вирішенні даної задачі в якості обмежень виступало те, що значення елементів векторів Λ_{ij} і гілок Λ повинні знаходитися в діапазоні [0; 100). Таким чином, отримане рішення задовольняє поставленим обмеженням, при цьому необхідно відзначити, що це одне можливе з множини рішень.

Таким чином, у даному підрозділі проаналізовано метод дослідження оверлейних мереж с використанням контурного методу аналізу. У зв’язку з тим що мережна структура багатьох інфраструктур будується саме на базі оверлейних мереж, провайдери, що надають даний вид послуг, повинні забезпечувати гарантії за якістю обслуговування потоків трафіку для кожної мережі. Велика кількість підтримуваних оверлеїв і розгалужена інфраструктура мережі провайдера робить задачу управління потоками трафіку досить рутинною, в свою чергу проаналізований метод аналізу дуже добре описує розподіл трафіку в великих системах завдяки добре формалізованому матричному математичному апарату, який залишається інваріантним до розміру досліджуваної топології.

Також необхідно відзначити, що арифметичні операції, що виконуються над матрицями, є добре розпаралелюваними обчислювальними операціями, що дозволяє на багатопроцесорних системах скоротити час аналізу мереж. Перевагами даного підходу в порівнянні з існуючими методами є простота отримання математичної моделі мережі з невідомою кількістю оверлеїів, складність обчислення маршрутів між вузлами зростає лінійно зі збільшенням числа каналів в мережі провайдера, в разі якщо необхідно провести аналіз можливості організації маршруту по конкретним областям мережі. Але якщо стоїть задача в пошуку довільного маршруту, який буде рішенням системи рівняння (2.6), то накладається обмеження на відсутність маршрутних петель в кожному маршруті між оверлеями збільшує складність завдання.

Також необхідно відзначити, що вид тензора перетворення контурних інтенсивностей повністю визначається матрицею лінійно незалежних циклів досліджуваної мережі, тому в результаті досліджень оверлейних мереж з

різною структурою контурним методом можна сформулювати кілька вимог до вибору матриці лінійно незалежних циклів, з метою зменшення витрат часу на обчислення.

У якості тензора перетворення краще вибирати матрицю фундаментальних циклів або матрицю лінійно незалежних циклів, де кожен цикл включає в себе гілку джерела і гілка одержувача.

2.2 Аналіз анонімних оверлейних мереж

Механізми анонімізації в мережі Інтернет дозволяють користувачам використовувати можливості оверлейної мережі для багатьох видів діяльності. Користувачі використовують Tor, I2P і інші оверлейні мережі для анонімної взаємодії і різних анонімізуючих дій в сфері комп’ютерної інформації [11].

Для прикладу розглянемо особливості функціонування анонімної мережі Tor. Функціонування забезпечується мережею проксі-серверів в Інтернеті, які здійснюють передачу даних в зашифрованому вигляді. Доступ в мережу надається за допомогою вільно поширюваного програмного забезпечення.

Багаторівневе шифрування, застосування цибульної маршрутизації і розподілений характер мережі роблять практично марним перехоплення трафіку. Відсутність логічного зв’язку між відправником і повідомленням гарантує надійну анонімність. Це сприяло виникненню в мережі Tor глибинних підмереж з неявною кількістю оверлеїв.

Крім того, дані мережі використовуються для організації спільних розподілених обчислень. Додаткові можливості користувачам надає пірінгова електронна підтримки роботи оверлеїв.

Незважаючи на ряд досліджень, в даний час не існує ефективних технічних рішень для деанонімізація користувачів в анонімних розподілених комп’ютерних мережах. Для аналізу даних в даних мережах вдаються до

методів соціальної інженерії, впроваджуючи в мережі свої віртуальні майданчики, приховані сервіси і проміжні маршрутизатори, переводячи тим самим мережі під свій контроль. При цьому процедуру блокування також можна вважати одним з найбільш ефективних методів протидії [15]. Можна виділити наступні види блокування неявних оверлеїв:

- блокування веб-сайтів, що поширюють програмне забезпечення Tor (якщо поле «Host» в заголовку HTTP = «torproject.org»);
- блокування кореневих серверів (directory servers), що поширюють базу доступних маршрутизаторів (onion routers);
- блокування IP-адрес всіх доступних маршрутизаторів (onion routers) з бази кореневого сервера;
- виявлення і блокування трафіку мережі Tor.

Для обходу зазначених механізмів блокування користувачі підключаються через спеціалізовані Tor-мости – вузли, які функціонують як вхідні маршрутизатори, адреси яких не поширюються за допомогою кореневих серверів, а передаються в приватному порядку. При цьому адреси Tor-мостів поширюються через різні підмережі і є неявнимии. Для блокування Tor-мостів застосовується механізм перевірки з’єднання (connection probe). Міжмережевий екран «заморожує» все підозрілі з’єднання і виробляє пробне підключення по протоколу Tor для перевірки зафіксованих IP-адрес.

Розробниками Tor розроблений механізм опосередкованої передачі трафіку через легальні сервіси (pluggable transports). З цією метою потік Tor-трафіку трансформується для передачі між клієнтами мережі та Tor-мостом в трафік легальних мережних додатків (наприклад, Skype), що дозволяє вводити в оману механізм глибинного аналізу мережі [16].

Запущені на Tor-мостах служби виробляють зворотну трансформацію трафіку, після чого дані клієнтів потрапляють в мережу Tor. Застосовуваний метод аналізу даних визначається використовуваними модулями подібних сервісів, що додатково ускладнює завдання для механізмів виявлення. Слід

зауважити, що структура такого трафіку не завжди повністю відповідає меті анонімізації, що може бути використано для його виявлення [17].

Незважаючи на що застосовується маскування і шифрування, трафік Tor і інших анонімних розподілених комп'ютерних мереж схильний прихованим залежностям і закономірностям [18]. Є можливим використання методів математичного та імітаційного моделювання, алгоритми і методи теорії розпізнавання образів і статистичного аналізу для вирішення задачі виявлення трафіку анонімних розподілених комп'ютерних мереж (рисунок 2.4).

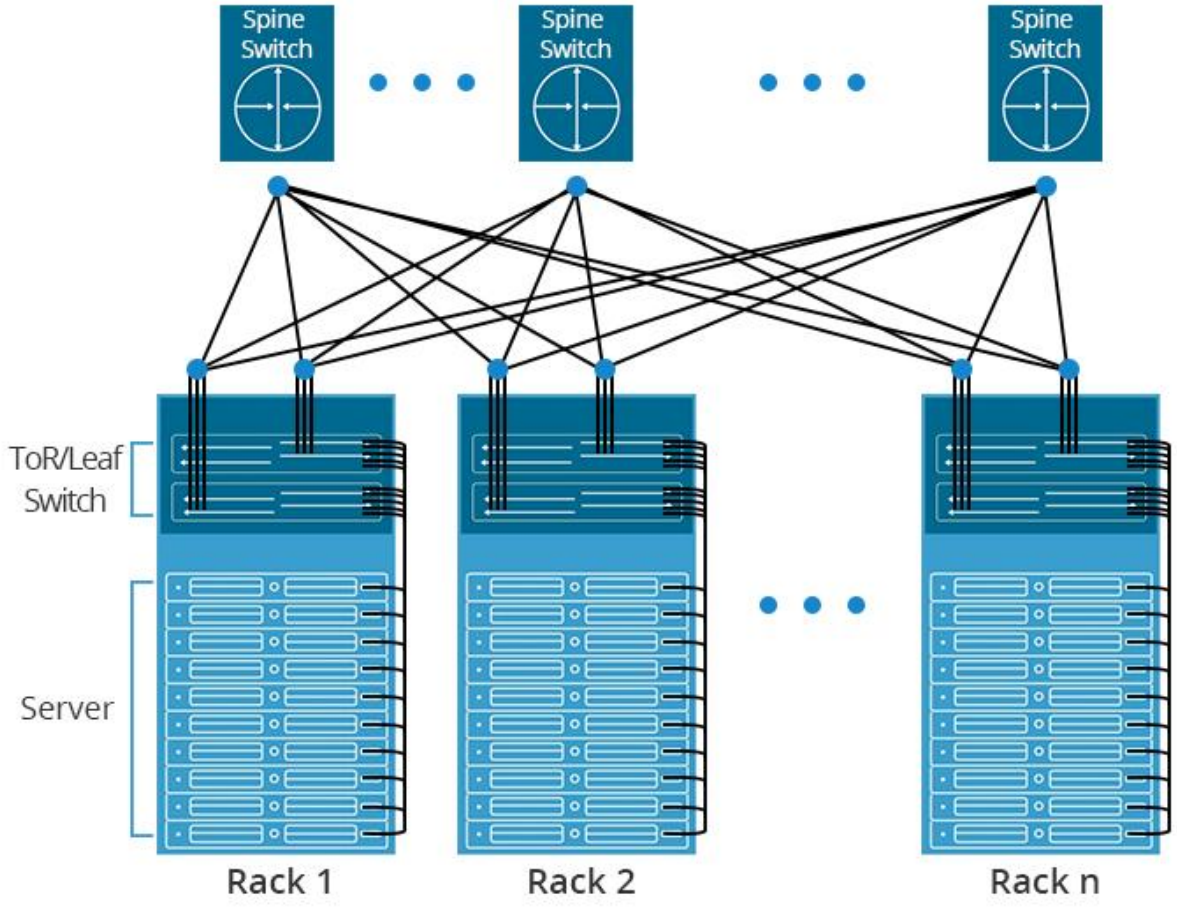

Рисунок 2.4 – Приклад застосування TOR-комутації в комп'ютерній мережі дата-центру

Проблема ідентифікації інтернет-трафіку досліджується вже більше 20 років. Традиційний підхід до ідентифікації, заснований на номерах портів,

закріплених за відомими мережними службами, залишається найбільш простим і швидким способом ідентифікації трафіку. Разом з тим низька ефективність даного підходу зумовила виникнення сигнатурного підходу до аналізу трафіку. Даний підхід заснований на синтаксичному аналізі потоку трафіку з метою пошуку відомих бітових послідовностей, що генеруються різними мережними додатками.

В результаті еволюції зазначеного підходу був запропонована ефективна технологія ідентифікації мережевих пакетів по їх вмісту та виявлення неявних вузлів анонімної мережі, включаючи дані прикладного рівня. Разом з тим аналізато трафіку має ряд обмежень. Зазначені проблеми не дозволяють обмежитися використанням аналізатору для ідентифікації Tor-трафіку.

Всі якісні і кількісні характеристики віртуального трафіку мережі Tor (або інший анонімної мережі) можна умовно розділити на дві групи. Перша група включає характеристики (використовувані номери портів, термін дії TLS-сертифікатів тощо), які явно вказують на приналежність трафіку до мережі Tor (або іншим анонімним мереж). На основі даної групи характеристик необхідно розробити загальну базу сигнатур трафіку анонімних розподілених мереж. Однак слід враховувати, що програмна реалізація Tor має відкритий код, а розробники системи мають можливість оперативної модифікації параметрів трафіку, що впливають на його виявлення за допомогою сигнатур. Таким чином, необхідно передбачити можливість автоматизованої актуалізації бази сигнатур з метою симетричної відповіді та адаптації до реакції розробників і зловмисників.

В ході наступного етапу досліджень повинна бути виділена група якісних і кількісних характеристик трафіку (тимчасові затримки відповідей, які використовуються алгоритми шифрування і створення загального ключа, кількість пакетів), побічно вказують на можливість приналежності трафіку до Tor.

Дані характеристики, побічно вказують на можливість приналежності

трафіку до Tor, не забезпечують максимальної точності виявлення Tor-трафіку. Разом з тим їх сукупне застосування дозволяє суттєво підвищити достовірність і надійність його виявлення.

Перевага характеристик другої групи також полягає в тому, що вони в меншій мірі схильні до впливу розробників. Наприклад, не існує технічної можливості знизити тимчасові затримки, пов’язані з багаторівневим шифруванням. Для сукупного застосування цих характеристик пропонується розробити композитну метрику захищеного трафіку (обчислюється на основі зазначених характеристик). Розробка метрики здійснюється на основі збору і статистичного аналізу великих масивів даних (включаючи дані мережі Tor). Аномальні значення даної метрики з певною ймовірністю (в залежності від значення) можуть свідчити про приналежність трафіку до неявних оверлейних вузлів Виходячи з експериментальних спостережень, для побудови бази сигнатур і композитної метрики TLS-трафіку був виділений набір наступних його характеристик:

- період актуальності сертифіката TLS (термін дії сертифікатів Tor зазвичай значно коротше частовживаних сертифікатів);
- значення поля Common Name сертифіката TLS (в сертифікатах Tor використовується випадковий рядок символів);
- порядок вибору застосовуваних криптографічних алгоритмів;
- глибина ланцюжка сертифікатів (для підключень Tor застосовуються самопідписні сертифікати);
- тимчасовий період між запитом і відповіддю в рамках одного TLS-підключення (багаторівневе шифрування, що застосовується в Tor, призводить до його суттєвого збільшення).

Є можливим здійснення генерації сигнатур Tor-трафіку для регулярної автоматизованої актуалізації бази сигнатур, виходячи з результатів виявлення аномалій оверлейного трафіку. Таким чином, комплексна взаємодія двох зазначених підходів дозволяє поліпшити відомі результати в даній області (знизити ймовірність помилок виявлення першого і другого

роду). Для вирішення зазначеного завдання існує багато алгоритмів актуалізації бази сигнатур трафіку анонімних розподілених комп'ютерних мереж на основі методів машинного навчання.

Для подолання механізмів прихоування трафіку потрібна розробка алгоритму виявлення трафіку анонімних розподілених мереж в загальному потоці трафіку мережних додатків на основі методів теорії розпізнавання образів.

Сучасний підхід до аналізу і ідентифікації трафіку ґрунтується на застосуванні алгоритмів розпізнавання образів із застосуванням технологій машинного навчання. Даний підхід, на відміну від класичного аналізу даних, дозволяє аналізувати і ідентифікувати навіть зашифрований трафік. При цьому ефективність коректної ідентифікації трафіку, і зокрема трафіку анонімних розподілених мереж, сильно залежить від обраного набору критеріїв і повноти опису ідентифікованого типу трафіку.

Застосування такого навчання без учителя допомагає адаптуватися до постійних змін внутрішньої структури мережевого трафіку при великій кількості неявних вузлів оверлейної мережі. Також додатково досліджуються методи класифікації трафіку (дерево класифікації, метод опорних векторів) для оцінки їх застосовності до класифікації прихованого трафіку анонімної мережі Tor.

Очевидно, що для ефективного виявлення трафіку від невідомих оверлеїв анонімних розподілених комп'ютерних мереж потрібно сукупне застосування всіх зазначених підходів. Розробка методів виявлення зашифрованого і прихованого трафіку анонімних розподілених комп'ютерних мереж дозволяє отримати відносну оцінку частки даного трафіку в загальному обсязі призначеного для користувача інтернет-трафіку і абсолютну оцінку числа користувачів даних оверлейних мереж.

3 МОДИФІКОВАНИЙ КВЕСТ-МЕТОД АНАЛІЗУ ОВЕРЛЕЙНИХ МЕРЕЖ З НЕВИЗНАЧЕНОЮ КІЛЬКІСТЮ ОВЕРЛЕЇВ

Оверлейні мережі з динамічною топологією мережі як правило реалізуються на платформі MANET (Mobile Ad hoc Networks). Основне призначення подібних мереж – організація безпечного зв’язку між рухомими об’єктами – людьми, автомобілями, залізничним і водним транспортом.. Причиною, що викликає неможливість передачі трафіку реального часу з великою швидкістю в таких мережах – є особливості роботи протоколів канального і мережного рівнів, викликані сильним впливом формального уявлення топологічних змін MANET-систем, які використовуються в програмних комплексах, призначених для дослідження оверлейних мереж з невизначеною кількістю оверлеїв. Основним математичним апаратом для подання топологічних змін до оверлейних мережах стала теорія випадкових графів, згідно з якою переміщення вузлів мережі представляється у вигляді броунівського руху, при цьому детермінованість траєкторії руху вузлів не враховується, що некоректно при розгляді в ролі мережних вузлів об’єктів транспорту.

У зв’язку з вищевикладеним атестаційних досліджень є коригування формальної інтерпретації топологічних змін, розробка математичної моделі процесу оцінки якості роботи оверлейних мереж з невизначеною кількістю оверлеїв, розробка модифікованого квест-методу аналізу таких оверлейних мереж.

В якості основи формального уявлення топологічних змін доцільно скористатися моделлю «мерехтливого графа» [18], суть якої зводиться до подання динамічної топології мережі у вигляді еволюціонуючого графу, який дискретно змінює свій стан:

$$\{G(\Delta t)\} = G(\Delta t_0) \to G(\Delta t_1) \to ...G(\Delta t_k), \qquad (3.1)$$

де $G(\Delta t)$ – граф, який описує топологію протягом часу стаціонарного часу Δt.

Граф мережі, що знаходиться в стані *(k+1)*, відрізняється від графа, що знаходиться в стані *k*, на деякий приріст *ΔG*. *ΔG* показує кількість доданих або вилучених ребер з топології в процесі переходу системи зі стану *k* в стан *(k+1)*:

$$G(\Delta t_{k+1}) \to G(\Delta t_k) \cup \Delta G. \qquad (3.2)$$

У розглянутому поданні, на відміну від відомої моделі дискретного часу [19, 20], допускається, що *Δt≠const*. Це дозволяє поширювати його на оверлейні мереж з невизначеною кількістю оверлеїв, в яких відсутній період повторення топологічних станів. Узагальнення виразів (3.1), (3.2) дозволяє на понятійному рівні представити процес перетворення топологічних змін в просторі (коли топологія мережі змінюється внаслідок зміни взаємного розташування вузлів мережі відносно один одного) до топологічних змін в часі (коли топологія мережі змінюється внаслідок відкриття або закриття каналів зв’язку між вузлами).

Значення часу «відкриття» і «закриття» каналів зв’язку між вузлами мережі визначаються з умов руху вузлів і параметрів обладнання. Аналіз часу відкриття або закриття каналів зв’язку дозволяє судити про пов’язаності топології в комп’ютерній системі в процесі переміщення мережних вузлів і, як наслідок, приймати рішення про принципову працездатність мережі. Подібне прийняття рішення дозволяє при автоматизованому проектуванні відсівати свідомо непрацездатні топології.

В цілому процес оцінки працездатності проектованої мережі доцільно здійснювати в два етапи: на рівні аналітичного моделювання і на рівні імітаційного моделювання, за результатами дослідження виробляється рішення про придатність або непридатність заданої топології для надання заданих сервісів.

Формально процес оцінки можна представити у вигляді функціоналу:

$$A(t) = \Psi[\, f(\{x_i\},\{h\},t), \xi(\{v_i\},t)] \tag{3.3}$$

де функціонал $f(\{x_i\},\{h\},t)$ – це аналітична складова моделі, за допомогою якої визначається час взаємодії мережних вузлів;

$\{x_i\}$ – множина параметрів рухомого i (траєкторія і швидкість руху, параметри обладнання, що відповідають за дальність встановлення з'єднання при заданих якісних показниках трафіку);

$\{h\}$ – параметри статичних вузлів, імітаційна складова моделі показана у вигляді функціоналу $\xi(\{v_i\},t)$ та призначається для оцінки роботи системи зв'язку з урахуванням трафіковий навантаження, алгоритмів роботи використовуваних мережевих протоколів та інших імовірнісних характеристик мережі, представлених у вигляді множини $\{v_i\}$.

Виходом аналітичної складової моделі є множина значень часу «відкриття» і «закриття» каналів між вузлами-учасниками оверлейної мережі:

$$f(\{x_i\},\{h\},t) = \begin{bmatrix} \left\{T_{1,1}^{відк}\right\},\left\{T_{1,1}^{закр}\right\} & \dots & \left\{T_{1,m}^{відк}\right\},\left\{T_{1,m}^{закр}\right\} \\ \dots & \dots & \dots \\ \left\{T_{n,1}^{відк}\right\},\left\{T_{n,1}^{закр}\right\} & \dots & \left\{T_{n,m}^{відк}\right\},\left\{T_{n,m}^{закр}\right\} \end{bmatrix}, \tag{3.4}$$

де $\left\{T_{n,m}^{відк}\right\}$ – безліч значень часу відкриття каналу між заданою парою вузлів;

$\left\{T_{n,m}^{закр}\right\}$ – безліч значень часу закриття каналу між заданою парою вузлів.

Елементи множин $\left\{T_{n,m}^{відк}\right\}$, $\left\{T_{n,m}^{закр}\right\}$ визначаються за допомогою математичних моделей руху вузлів мережі і величини зони покриття кожного вузла (детально методика розрахунку часу взаємодії двох рухомих вузлів

розглянута в [18, 19]). Сутність методики зводиться до того, що на підставі даних про умови переміщення вузлів і даних про розміри зони покриття складаються рівняння, рішеннями яких є комплексні числа виду $t = a + ib$, з їхнього вигляду можна судити про характер взаємодії вузлів. Основні випадки можна описати таким чином:

- якщо $b \neq 0$, то канал зв'язку між вузлами не відкривався (з позиції теорії графів це означає, що в топології відсутні шляхи першого рангу між заданими парами вузлів);
- якщо $a \geq 0$, $b = 0$ та $a_n < a_{n+1}$, то a_n – час початку взаємодії і a_{n+1} – час закінчення взаємодії двох вузлів;
- якщо $a \leq 0$ і $b = 0$, то об'єкти почали взаємодіяти на момент початку експерименту.

Аналіз значень матриці, показаної в виразі (3.4), дозволяє судити про пов'язаності топології (графа) мережі в процесі переходу вузлів із зони дії один одного. У спрощеному вигляді умова, за яким здійснюється оцінка пов'язаності топології мережі, виглядає наступним чином:

$$t_{закр_n} > t_{откр_{n+1}}. \qquad (3.5)$$

Умова означає, що час закінчення взаємодії з попереднім вузлом має наступати дещо пізніше, ніж час початку взаємодії з подальшим вузлом. Умова (3.5) є необхідною для забезпечення пов'язаності мережі з динамічною топологією, достойність (визначення різниці $t_{закр_n} - t_{откр_{n+1}}$) визначається швидкістю роботи керуючих систем комп'ютерного обладнання, що знаходиться на вузлі, яке, згідно з [19], визначається зі співвідношення

$$t_{кер} = t_{дост} + t_{пку}, \qquad (3.6)$$

де $t_{дост}$ – час, необхідний для достовірної оцінки рівня сигналу,

необхідного для передачі трафіку з заданими параметрами якості, по каналу між рухомими вузлами;

$t_{пку}$ – час, необхідний обладнанню для ухвалення рішення про відкриття нового каналу.

З фізичної точки зору процедура передачі рухомого вузла між сусідніми рухомими або стаціонарними вузлами в цілому аналогічна процедурі естафетної передачі в існуючих системах рухомого зв'язку.

Залежно від способу управління мережею рішення про передачу вузлів може прийматися або безпосередньо на самому вузлі, при децентралізованому управлінні, або в центрі управління при централізованому управлінні. В якості основного контрольованого параметра, на підставі значення якого приймається рішення про «закриття» або «відкритті» каналу з іншими вузлами, найбільш доцільно використовувати рівень потужності, при досягненні певного порогового значення якого $P_{пор}$ включається процедура попередження, а в разі досягнення деякого допустимого значення $P_{доп}$ відбувається прийняття рішення або про перехід на роботу з іншим вузлом мережі, або про припинення передачі інформації, що фактично призводить до втрати пов'язаності топології мережі.

Оскільки виразами (3.1)-(3.3) в цілому допускається відсутність детермінованості топологічних змін, то процес зміни каналу доцільно представити за допомогою ланцюга Маркова, як це було вже зроблено в [22] стосовно до процедури квест-передачі між базовою станцією і оверлеєм (рисунок 3.1).

На рисунку 3.1 прийняті наступні позначки:

U_1 – оптимальний рівень прийому на V_j ($V_{j+2},...V_{j+m}, m$ – парне число, $m \neq \infty$);

U_2 – сигнал про зростання інтенсивності передачі даних між оверлеями, наближення до значення $P_{пор}$;

U_3 – стан закриття каналу зв’язку між заданою парою оверлеїв;

U_4 – процедура зміни сусіднього оверлею;

U_5 – приближення до порогового значення $P_{пор}$ на V_{j+1} (V_{j+3},...V_{j+n}, n – парне число, $n \neq \infty$).

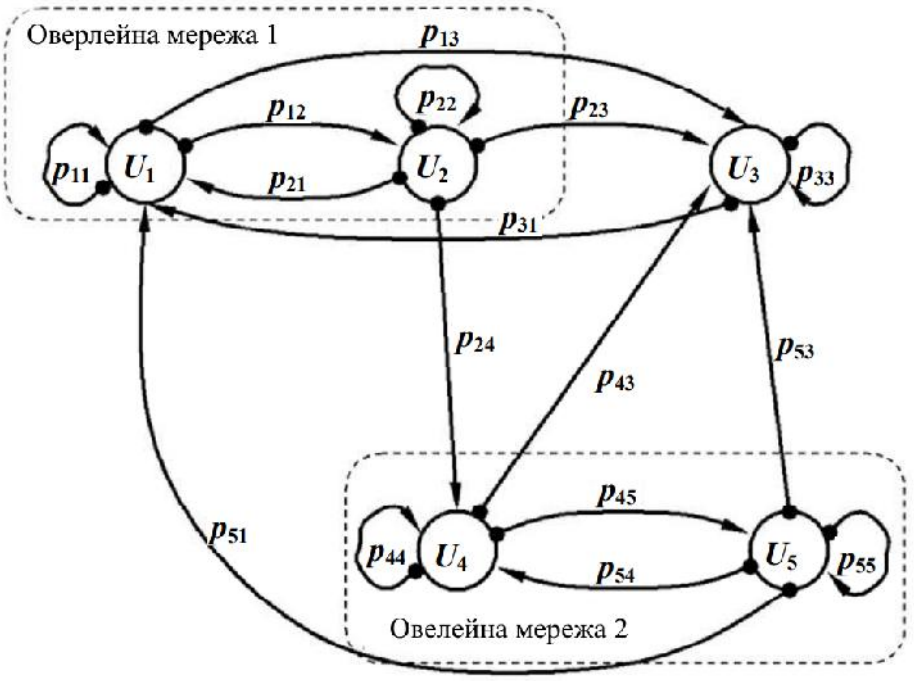

Рисунок 3.1 – Послідовність подій при зміні каналу між рухомими або рухомим і стаціонарними оверлейними мережами

Грунтуючись на тому, що процедури зміни стану системи події несумісні, а також на теоретичних положеннях, наведених в [23], умови переходу станів системи можна представити у вигляді наступних виразів (3.7)-(3.13).

Масив формул (3.7)-(3.13) з формальної позиції описує процес прийняття рішення про «закриття» або «відкриття» каналів зв’язку між рухомими вузлами мережі, що необхідно як при написанні алгоритмів роботи програмних комплексів по моделюванню оверлейних мереж з невизначеною кількістю оверлеїв (з метою підвищення достовірності результатів

моделювання за рахунок обліку умов поширення даних), так і при створенні систем автоматизованого управління оверлейними мережами з невизначеною кількістю оверлеїв.

$$p_{11}=\begin{cases}1, P_{\text{тек}}^{i+m} > P_{\text{пор}} \\ 0\end{cases},\ p_{12}=\begin{cases}1, P_{\text{пор}} > P_{\text{тек}}^{i+m} > P_{\text{доп}} \\ 0\end{cases},\ p_{13}=\begin{cases}1, P_{\text{тек}}^{i+m} < P_{\text{доп}} \\ 0\end{cases};\qquad (3.7)$$

$$p_{22}=\begin{cases}1, P_{\text{тек}}^{i+m} = P_{\text{доп}} \\ 0\end{cases},\ p_{21}=\begin{cases}1, P_{\text{тек}}^{i+m} \geq P_{\text{пор}} \\ 0\end{cases},\ p_{23}=\begin{cases}1, P_{\text{тек}}^{i+m} < P_{\text{пор}} \\ 0\end{cases},\qquad (3.8)$$

$$p_{24}=\begin{cases}1, P_{\text{тек}}^{i+n} > P_{\text{пор}} \\ 0\end{cases};\qquad (3.9)$$

$$p_{33}=\begin{cases}1, P_{\text{тек}}^{\forall i} < P_{\text{пор}} \\ 0\end{cases};\qquad (3.10)$$

$$p_{43}=\begin{cases}1, P_{\text{тек}}^{i+n} < P_{\text{доп}} \\ 0\end{cases},\ p_{44}=\begin{cases}1, P_{\text{тек}}^{i+n} \approx P_{\text{тек}}^{i+m} \approx P_{\text{доп}} \\ 0\end{cases},\ p_{45}=\begin{cases}1, P_{\text{пор}} > P_{\text{тек}}^{i+m} \geq P_{\text{доп}} \\ 0\end{cases};\qquad (3.11)$$

$$p_{51}=\begin{cases}1, P_{\text{тек}}^{i+m} > P_{\text{доп}} \\ 0\end{cases},\ p_{53}=\begin{cases}1, P_{\text{тек}}^{i+m} < P_{\text{доп}} \\ 0\end{cases},\ p_{54}=\begin{cases}1, P_{\text{тек}}^{i+m} > P_{\text{пор}} \\ 0\end{cases};\qquad (3.12)$$

$$p_{55}=\begin{cases}1, P_{\text{тек}}^{i+m} < P_{\text{пор}} \\ 0\end{cases}.\qquad (3.13)$$

Після позитивних результатів перевірки топології мережі на зв’язаність, згідно з умовами (3.5), (3.6), проводиться дослідження роботи оверлейних мереж з невизначеною кількістю оверлеїв для надання необхідної номенклатури сервісів. В результаті досліджень виходять множини параметрів оцінки якості мультисервисного трафіку (затримка передачі, ймовірність втрати джиттер-пакетів), які порівнюються з нормованими значеннями, наприклад, рекомендованими концепцією QoS (Quality of Service). Формально цей процес можна представити у вигляді:

$$A(t) = \begin{cases} \{\chi\} \subset \{X\}, 1, \\ \{\chi\} \supset \{X\}, 0, \end{cases} \quad (3.14)$$

де $\{\chi\}$ – множина показників роботи оверлейних мереж з невизначеною кількістю оверлеїв, отриманих в ході імітаційного моделювання; $\{X\}$ – множина нормованих параметрів якості роботи системи.

Якщо *A(t)=1* – топологія системи визнається придатною для надання заданої номенклатури сервісів, і *A(t)=0* – в іншому випадку.

В алгоритмічній вигляді процес роботи програмного комплексу, що описується за допомогою виразів (3.3)-(3.14), представлений на рисунку 3.2.

В даний час аналітична частина моделі частково реалізована у вигляді програми (додаток Б), а імітаційна реалізована на базі середовища моделювання компюетрних систем Network Simulator [25]. Спільне використання двох програмних продуктів дозволяє скоротити загальний час дослідження оверлейних мереж з невизначеною кількістю оверлеїв в 2,5 рази в порівнянні з часом використання виключно засобів імітаційного моделювання.

Коригування формальної інтерпретації топологічних змін дозволила перевести процес зміни топології в просторі (коли топологія мережі змінюється внаслідок зміни взаємного розташування вузлів мережі відносно один одного) до топологічних змін в часі (коли топологія мережі змінюється внаслідок відкриття або закриття каналів зв’язку між вузлами), що дозволяє розглядати динамічні і статичні складові загальної топології мережі як єдине ціле.

Розроблена математична модель процесу оцінки якості роботи оверлейних мереж з невизначеною кількістю оверлеїв з урахуванням проведеної коригування дозволяє проводити дослідження топології мережі в два етапи: на першому етапі перевіряється зв’язаність.

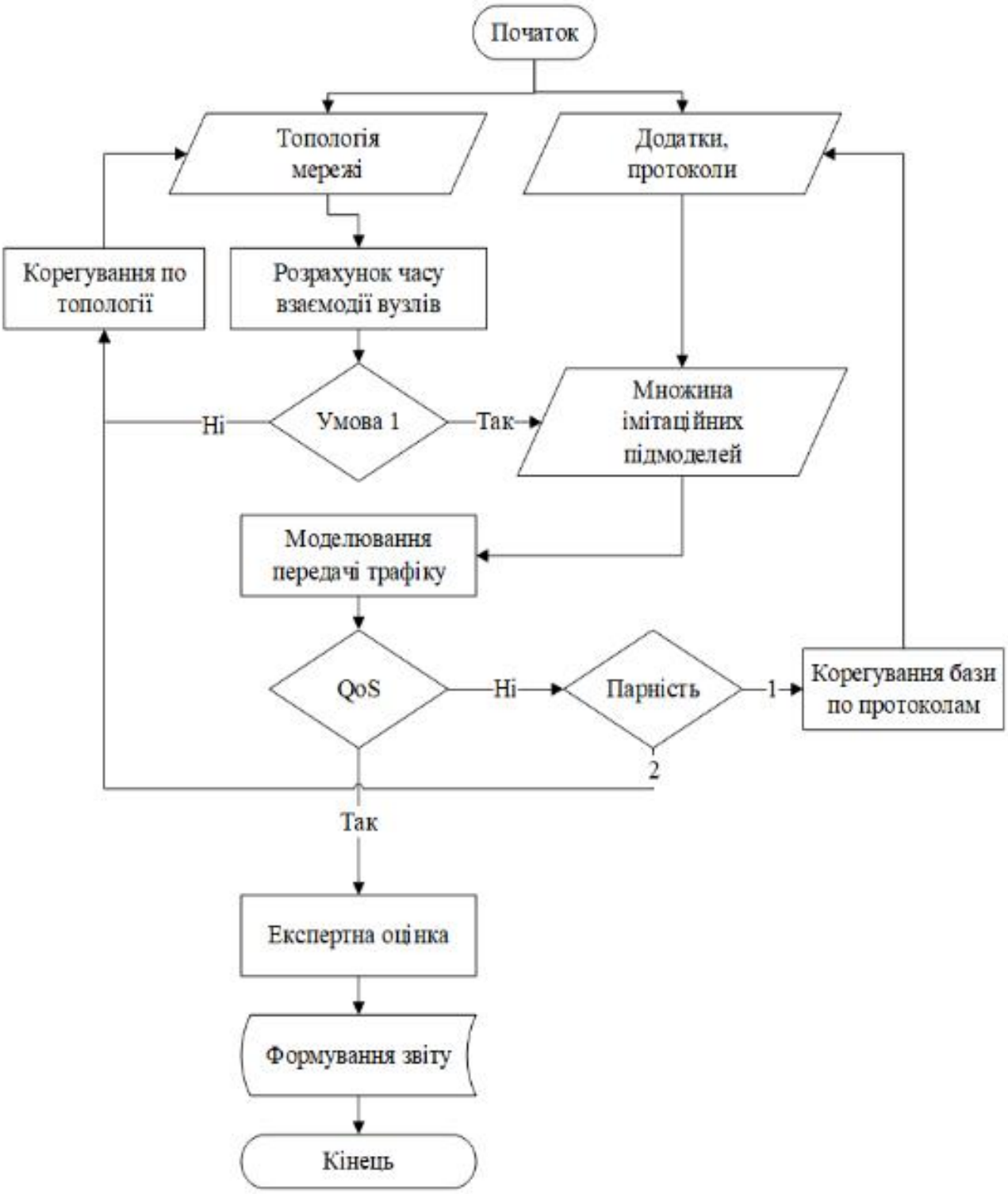

Рисунок 3.2 – Алгоритм роботи програмного комплексу за модифікованим квест-методом аналізу оверлейних мереж з невизначеною кількістю оверлеїв

На другому можливість передачі мультисервісного трафіку. В результаті формується рішення про придатність або непридатність заданої оверлейних мереж з невизначеною кількістю оверлеїв для роботи в заданих умовах.

Представлений в алгоритмічній вигляді процес роботи програмного комплексу, заснований на розробленій моделі, показує, яким чином проводиться вибір оверлейних мереж з невизначеною кількістю оверлеїв, реалізація якої найбільш доцільна в заданих умовах.

Таким чином, модифікований квест-метод аналізу оверлейних мереж з невизначеною кількістю оверлеїв дає можливість для розробки комплексів програм з моделювання оверлейних мереж з невизначеною кількістю оверлеїв з динамічною топологією мережі.

4 ДОСЛІДЖЕННЯ ХАРАКТЕРИСТИК СИСТЕМИ АНАЛІЗУ ОВЕРЛЕЙНИХ МЕРЕЖ З НЕВИЗНАЧЕНОЮ КІЛЬКІСТЮ ОВЕРЛЕЇВ

Розглядаючи рівень взаємодії вузлів оверлейної мережі з невизначеною кількістю оверлеїв можна зіставити модель мережі масового обслуговування, кожна система якої буде моделювати окремий оверлей передачі інформації (вхідний/вихідний інтерфейс мережі). Розглянутий інтерфейс можна представити у вигляді одноканальної системи масового обслуговування з умовними втратами, але для більшої відповідності реальному об’єкту можна обмежити число місць очікування.

На наступному (канальному) рівні можна вже не тільки розглядати віртуальні з’єднання, що існують між вузлами, а й враховувати напрямки передачі до оверлеїв мережі. Кожен напрямок буде задано окремою системою масового обслуговування, тип якої визначається дисципліною обслуговування реальної системи обробки інформаційних потоків. Крім того, слід враховувати, що основні функціональні залежності досліджуваних показників є функціями від завантаження пристроїв (систем масового обслуговування).

Отримані при розрахунку значення ймовірносно-часових показників необхідно використовувати для отримання значень оцінок якості обслуговування QoS на наступному мережному рівні, так як в конкретній реальній оверлейній мережі розподіл потоків по вузлах не випадковий, а підпорядковується таблиці маршрутизації. На даному рівні також необхідно враховувати розподіл інформаційних і сигнальних потоків по мережі.

Для кожного маршруту можна записати такі формули, за якими буде здійснюватися передача даних:

$$p_{втрат} = 1 - \sum_{i=1}^{m} (1 - p_{втрат,i}), \qquad (4.1)$$

якщо це сукупність гілок на маршруті у вигляді послідовного графа;

$$T_{затримки} = \sum_{i=1}^{m} T_{затримки,i}\ , \tag{4.2}$$

де m визначається загальним числом підмереж, складових маршруту передачі даних (при необхідності можна врахувати і наявність обхідних маршрутів).

Для всієї мережі в цілому є сенс визначити загальний середній час затримки:

$$T_{затримки} = \sum_{i=1}^{n} T_{затримки,i} \tag{4.3}$$

де n – загальна кількість підмереж в мережі.

У атестаційній роботі, у попередньому розділі було розглянуто тензорний підхід до задачі визначення параметрів якості обслуговування в даній мережі, тому скористуємося відомим апаратом. Для цього необхідно визначити примітивну мережу. Примітивна мережу при вузловому методі складається з незамкнутих відокремлених гілок, її найпростіший елемент – гілка. Рівняння стану гілки визначається співвідношенням:

$$\lambda = \mu\rho \tag{4.4}$$

де ρ – завантаження системи масового обслуговування (СМО);

λ – інтенсивність надходження повідомлень;

μ – інтенсивність обслуговування в системі [25].

Геометричні об’єкти примітивної мережі:

- $\bar{\lambda}'$ – вектор інтенсивностей потоків повідомлень в гілках;
- $\bar{\rho}'$ – вектор завантаження систем масового обслуговування;

- $\overline{\mu}'$ – квадратна матриця, діагональні елементи виражають інтенсивності обслуговування пакетів, інші елементи характеризують взаємний вплив оверлеїв один на одного:

$$\overline{\mu}' = \begin{bmatrix} \mu'_{11} & \mu'_{12} & \dots & \mu'_{1n} \\ \mu'_{21} & \mu'_{22} & \dots & \mu'_{2n} \\ \dots & \dots & \dots & \dots \\ \mu'_{n1} & \mu'_{n2} & \dots & \mu'_{nn} \end{bmatrix} \tag{4.5}$$

Еквівалентна система рівнянь, що описують примітивну мережу, відповідно до (4.4) буде мати наступний вигляд:

$$\begin{cases} \lambda'_1 = \mu'_{11}\rho'_1 + \mu'_{12}\rho'_2 + ... + \mu'_{1n}\rho'_n \\ \dots \\ \lambda'_n = \mu'_{n1}\rho'_1 + \mu'_{n2}\rho'_2 + ... + \mu'_{nn}\rho'_n \end{cases} \tag{4.6}$$

Рівняння перетворення для оверлейної мережі:

$$\begin{cases} \rho'_1 = A_{11}\rho_1 + A_{12}\rho_2 + ... + A_{1n-k}\rho_{n-k} \\ \dots \\ \rho'_1 = A_{n1}\rho_1 + A_{n2}\rho_2 + ... + A_{n,n-k}\rho_{n-k} \end{cases} \tag{4.7}$$

У матричної формі:

$$\overline{\rho}' = A\overline{\rho}\,, \tag{4.8}$$

де $\overline{\rho}$ – вектор завантаження в вихідній мережі;

A – матриця перетворення;

$\overline{\rho}'$ – вектор завантаження примітивної мережі.

При складанні матриці перетворення A для завантажень гілок примітивної мережі встановлюється співвідношення з завантаженнями в вихідної мережі, отримані коефіцієнти при завантаженнях вихідної мережі утворюють матрицю A.

Для встановлення формул перетворення геометричних об’єктів використовується припущення про те, що потік викликів з однієї і тієї ж інтенсивністю (λ) надходження при незмінній інтенсивності обслуговування викличе одне і те ж завантаження (ρ) пристроїв при зміні структури, і можна вважати, що буде виконуватися співвідношення (інваріант): $\rho\lambda = \rho'\lambda'$, де змінні зі штрихом відносяться до однієї структури мережі, без штриха – до іншої. Вектор інтенсивностей потоків повідомлень:

$$\overline{\lambda} = A^T\overline{\lambda}'. \qquad (4.9)$$

Матриця значень інтенсивностей обслуговування повідомлень в системах:

$$\overline{\mu} = A^T\overline{\mu}'A. \qquad (4.10)$$

Примітивна вузлова мережа з n гілок складається з n незамкнутих відокремлених гілок. Визначення компонент геометричних об’єктів примітивної мережі полягає в знаходженні векторів $\overline{\lambda}'$, $\overline{\rho}'$ і $\overline{\mu}'$. Знаходячи співвідношення між завантаженнями гілок примітивної мережі і завантаженнями в вихідній мережі, знаходимо матрицю переходу з однієї системи координат в іншу A. Виконавши заміну рівняння вихідної мережі на рівняння:

$$A^T\overline{\lambda}' = A^T\overline{\mu}'A\rho', \qquad (4.11)$$

Отримуємо систему лінійних рівнянь, що дозволяє визначити завантаження вихідної мережі, а далі можуть бути знайдені значення завантажень в гілках при заданих значеннях ймовірностей надходження викликів в окремі системи масового обслуговування. При відомих параметрах пристроїв з використанням розрахованих тензорним методом значеннях завантаження можна знайти параметри QoS для кожного маршруту.

При проектуванні або експлуатації телекомунікаційної мережі, що працює на базі протоколу IP за наведеними теоретичними викладеннями, можна враховувати, як розподіляти потоки в мережі, щоб кожен потік отримав необхідну якість обслуговування. На сьогоднішній день в керівних документах і рекомендаціях Міжнародного союзу електрозв’язку для мереж передачі даних за протоколом IP визначені норми затримок для різних класів обслуговування в службах передачі даних [2].

Для застосування тензорного аналізу до задачі оцінки системи аналізу оверлейної мережі як показників QoS була створена програмна система, яка реалізує тензорний підхід до вирішення поставленого завдання. У розробленій програмній системі використовується наступна послідовність етапів аналізу мережі.

Крок 1. Визначається структура примітивної мережі і задаються матричні компоненти, що описують примітивну мережу. Вектори інтенсивностей потоків надходження $\overline{\lambda}'$ і завантажень $\overline{\rho}'$ мають стільки елементів, скільки є гілок, і містять відповідні величини для кожної з них.

Крок 2. Знаходиться матриця перетворення *A*. Відносно першого вузла в новій мережі вибирається *k* нових незалежних вузлових завантажень. Для кожної окремої гілки завантаження примітивної (допоміжної) мережі $\overline{\rho}'$ виражаються через вузлові завантаження вихідної мережі $\overline{\rho}$. Коефіцієнти при нових завантаженнях утворюють матрицю перетворення *A*.

Крок 3. Обчислюються складові матриць $\overline{\mu}$ і $\overline{\lambda}$ за допомогою виразів (4.3) і (4.4), тобто визначаються компоненти матричного рівняння для

вихідної мережі через складові інваріантного рівняння для примітивної мережі.

Крок 4. Відповідно до (4.5) будується матричне рівняння, рішення якого дозволяє знайти вектор вузлових завантажень $\overline{\rho}$ і визначити коефіцієнти використання вузлів в мережі ($\overline{\rho}_{вузлів} = \overline{A}\overline{\rho}$), завдяки чому можна знайти необхідні параметри невідомих вузлів-оверлеїв (як по окремим системам, так і всієї мережі в цілому) і розподіл інтенсивностей потоків по вузлах мережі як $\overline{\lambda}_{оверлеїв} = \overline{\mu}_{оверлеїв}\overline{\rho}_{оверлеїв}$.

Розглянута послідовність етапів аналізу і розроблений теоретичний апарат добре формалізуються, що дозволяє, використовуючи доступні обчислювальні і програмні засоби, забезпечити реалізацію тензорного підходу в програмній системі аналізу оверлейної мережі. Укрупнений алгоритм роботи програмної системи представлений на рисунку 4.1.

Розроблений програмний комплекс створений в середовищі програмування Visio Studio і має дві основні робочі області: «Model» і «Analysis». В області «Model» виконується побудова досліджуваної схеми, в «Analysis» вводяться вихідні дані і визначаються основні характеристики мережі. Область «Analysis» містить кілька вкладок для введення і виведення даних. На вкладці «Матриця переходу» виводиться матриця *A*. На окремих вкладках виводяться матриці інтенсивностей надходження і обслуговування $\overline{\lambda}$ і $\overline{\mu}$, виражені в символах примітивної мережі відповідно до (4.3), (4.4). На вкладці «Введення установок» задаються необхідні інтенсивності надходження і обслуговування, тип систем масового обслуговування, розмірність буфера, порядок розподілу, маршрут і формат представлення чисел. На вкладці результатів виводяться результати розрахунку як для кожної системи окремо, так і для всієї оверлейної мережі в цілому: стаціонарні ймовірності (для систем), середня чергу, середній час затримки, вузлові завантаження, обчислені значення завантажень і інтенсивностей надходження пакетів в систему. Отримані значення для уявних гілок

відзначені зірочкою і в розрахунках для загального середньої часу затримки (або для маршруту) і ймовірності втрат на маршруті не беруть участь.

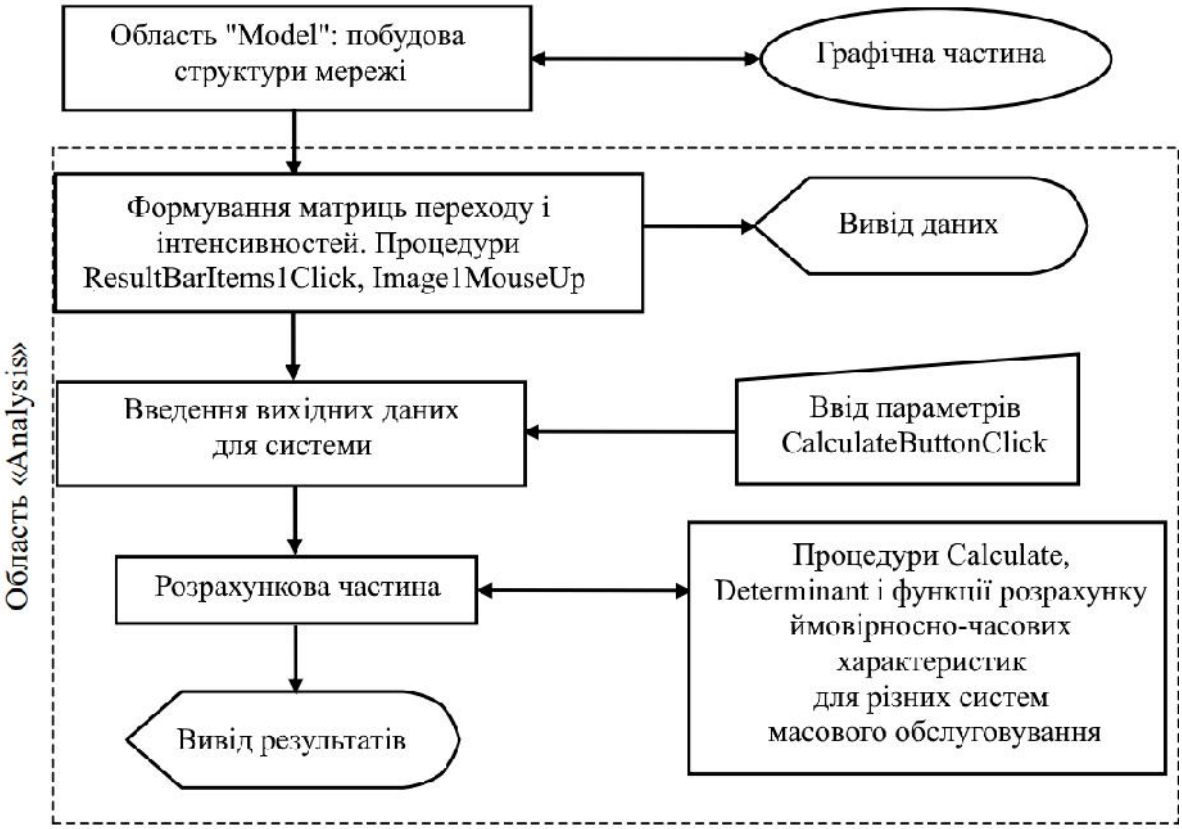

Рисунок 4.1 – Укрупнений алгоритм роботи програми

Основними вихідними компонентами, з якими працює програмна система, є масиви даних NodeMASS і VetvMASS. У цих масивах знаходиться вся необхідна інформація для подальшого обчислення необхідних характеристик як окремих систем, так і для всієї мережі в цілому.

NodeMASS – це масив даних, що містить інформацію про координати вузлів. В даному випадку під вузлом розуміється точка сполуки кількох систем масового обслуговування або входу / виходу гілки. Координата вузла має строго задані значення на поле побудови схеми в робочій області «Model». Кожен вузол однозначно описується парою чисел, що визначають місце вузла на даному полі. Однак для уявних гілок заданий діапазон зсуву від відображуваних на поле вузлів з метою реалізації вузлового методу тензорного аналізу мереж.

VetvMASS є масивом даних, що містить інформацію про номер, координатах початку і кінця (і їх повторному використанні) і удаваності гілки. Під гілкою в даному випадку розуміється система масового обслуговування. Ознака повторного використання координат дозволяє виявити з'єднання гілок між собою і наявність замкнутих контурів і, відповідно, забезпечити їх перетворення в вузлові пари.

Для наочного уявлення схеми досліджуваної мережі і формування масивів NodeMASS і VetvMASS використовується процедура Image1MouseUp. Окремий цикл даної процедури забезпечує формування уявних гілок з відповідними координатами. Уявні гілки формуються в зв'язку із застосуванням в ядрі програми вузлового методу, тобто все контури перетворюються в вузлові пари, в зв'язку з чим з'являються уявні гілки з тими ж інтенсивностями надходження і обслуговування, що і для істинної.

На рисунку 4.2 показано перетворення схеми з появою уявної гілки оверлею 5, для якої $\lambda_5 = \lambda_4$ і $\mu_5 = \mu_4$. При цьому на самій схемі (область «Model») уявні гілки не відображаються. Введення уявної гілки 5 дозволяє зберегти співвідношення $\lambda_1 = \lambda_2 + \lambda_4$ і $\lambda_3 = \lambda_2 + \lambda_4$, які можуть бути порушені при заміні контуру на вузлову пару.

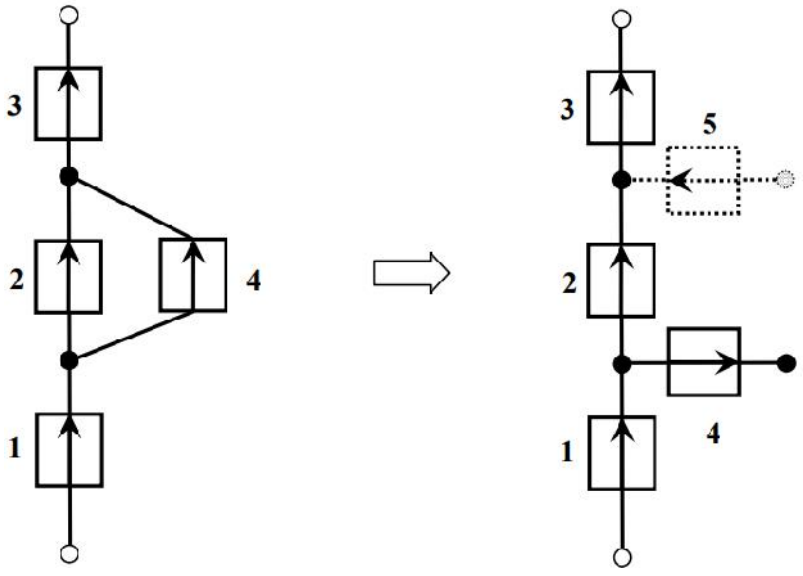

Рисунок 4.2 – Заміна закритого контуру на відкритий в оверлейній мережі

Процедура ResultBarItems1Click формує з отриманих масивів NodeMASS і VetvMASS матрицю переходу і на її основі матриці інтенсивностей надходження Matrix_N і обслуговування Matrix_T в нових координатах. Так як виконується умова суми інтенсивностей надходження в вузлі, визначаються інтенсивності надходження, які є лінійними комбінаціями інших інтенсивностей, і, відповідно, визначаються номери систем, для яких немає необхідності в завданні даних інтенсивностей. У цій же процедурі знаходиться блок підготовки даних для використання у обчисленнях: заміна точок на коми, обмеження довжини і типу введення.

Після введення даних на вкладці «Введення установок» (заповнення матриці (4.2)) процедура CalculateButtonClick забезпечує перевірку вихідних даних на коректність заповнення для всіх неуявних галузей: заповнення матриць інтенсивностей надходження і обслуговування, вибір значень буфера для систем з обмеженим буфером, вибір порядку для систем з ерланговський розподілом тривалості обслуговування, завдання формату виведення результату (число знаків після коми, науковий або інженерний формати).

Основна розрахункова процедура Calculate забезпечує знаходження вузлових завантажень з використанням отриманого матричного рівняння і введених вихідних даних. Спочатку відбувається обнуління масивів імовірнісних параметрів, а також перевірка гілок на уявність і при необхідності внесення змін до таблиці вихідних даних. Далі для кожної гілки визначаються інтенсивності надходження і обслуговування, з яких формуються масиви *N* (інтенсивностей надходження: рівняння (4.3)) і *T1* (інтенсивностей обслуговування: рівняння (4.4)) на основі вихідних даних і матриці переходу. Після цього проводиться перевірка отриманих матриць на вирожденність, що дозволяє виявити некоректність заповнення таблиці вихідних даних. Для знаходження вузлових завантажень використовується функція Determinant, яка дозволяє знайти визначник при вирішенні системи лінійних алгебраїчних рівнянь із застосуванням формул Крамера. В

алгоритмі функції Determinant реалізований захист від наявності нульових елементів на головній діагоналі. На основі отриманих вузлових завантажень розраховуються завантаження систем масового обслуговування, а також визначаються нові значення маршрутів прямування даних в оверлейних мережах з невизначеною кількістю оверлеїв (рисунок 4.3.)

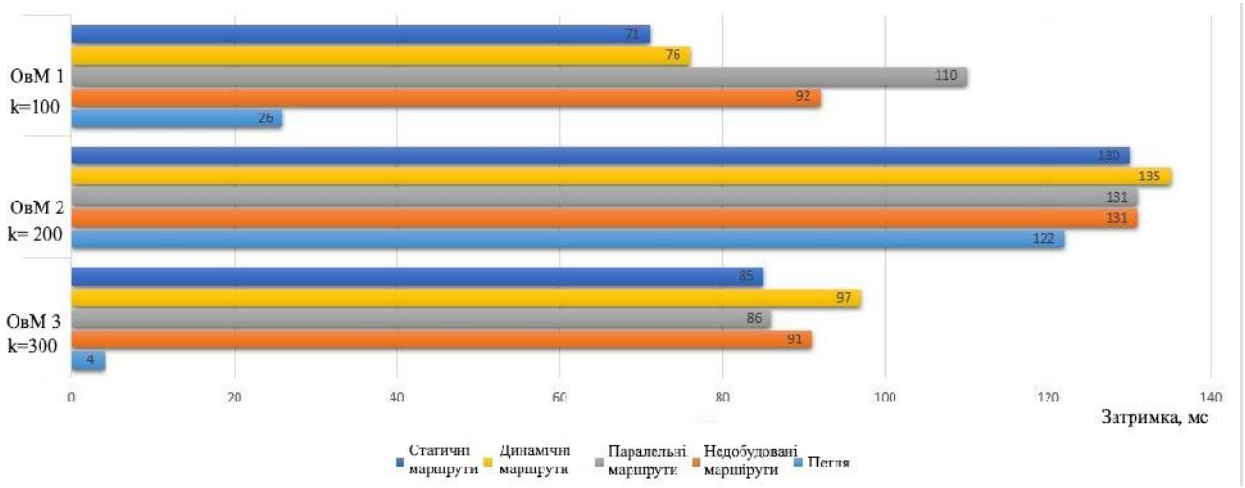

Рисунок 4.3 – Вирахувані значення маршрутів прямування даних в оверлейних мережах з невизначеною кількістю оверлеїв

При цьому проводиться перевірка на перевантаження систем, тобто забезпечується значення завантажень систем, що не перевищує одиниці, і перевірка на нульове або від’ємне значення інтенсивностей надходження. Попередньо обчислені значення залишаться завантажень далі передаються в блок розрахунку систем, де розраховуються середня чергу, середній час затримки, ймовірності станів і втрат.

Виходячи з отриманих значень для кожної системи проводиться розрахунок середнього часу затримки і ймовірності втрат на маршруті, загального часу затримки в мережі. Всі результати передаються в блок підготовки даних для виведення в таблицю результатів на відповідній вкладці.

ВИСНОВКИ

В результаті реалізації модифікованого квест-методу для задачі аналізу оверлейних мереж з невизначеною кількістю оверлеїв була розроблена програмна система, яка дозволяє забезпечити можливості побудови будь-яких топологій схем і вибору моделі для кожної гілки оверлейної мережі з метою забезпечення анонімності її користувачів. При цьому результатом теоретичної роботи є розрахунок довжини черг і час затримки в гілках, а також ймовірності переповнення буферів і стаціонарні ймовірності станів для кожної гілки, обраного маршруту і всієї мережі в цілому, що впливає на ступінь анонімності мережі.

Виходячи з усього сказаного можна зробити наступні висновки: модифікований квест-метод аналізу оверлейних мереж з невизначеною кількістю оверлеїв є новим і дозволяє об’єднати практично всі існуючі послуги мереж з забезпеченням заданого рівня якості передачі даних; тензорний метод дає можливість досить просто формалізувати проектні процедури для мереж такого типу; програмна реалізація квест-методу дозволяє оцінювати необхідні показників якості при прийнятних обчислювальних витратах.

Дослідження в рамках виконання атестаційної роботи проводилися на базі системи віртуалізації навчально-наукової лабораторії мікроконтролерних систем кафедри електронно-обчислювальних машин Харківського національного університету радіоелектроніки.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. V. Tkachov, M. Hunko, M. Bondarenko, S. Artyomov Technology of Load Balancing in Anonymous Network Based on Proxy Nodes Cascade Platform / Міжнародна науково-технічна конференція "Комп'ютерні та інформаційні системи і технології". – Харків. – С. 82.

2. Талалаев А. А. Организация конвейерно-параллельных вычислений для обработки потоков данных //Информационные технологии и вычислительные системы. – 2011. – №. 1. – С. 8-13.

3. Кротов К. В. Жадный алгоритм построения расписаний обработки данных в конвейерных системах // Вестник Воронежского государственного университета. Серия: Системный анализ и информационные технологии. – 2015. – №. 1. – С. 44-60.

4. Можаєв О.О. Аналіз поведінки черг маршрутизаторів у мережах передачі даних в залежності від типів трафіку / О.О. Можаєв, Ю.Ю. Завизіступ, А.А. Коваленко. Системи озброєння і військова техніка. – 2009. – № 2(18). – С. 103-105.

5. S. Zhang, C. Zhang and Q.Yang. Data preparation for data mining, Applied artificial intelligence, Vol. 17, No. 5-6, pp. 375-381, November 2003.

6. V. Tkachov, V. Tokariev, Ya. Dukh and V. Volotka. Method of Data Collection in Wireless Sensor Networks Using Flying Ad Hoc Network, in Proc. 5th International Scientific-Practical Conference Problems of Infocommunications. Science and Technology, Kharkiv, Ukraine, October 9-12, 2018, pp. 197-201.

7. Коваленко А.А. Аналіз та підходи до використання методів зміни простору рішень для оптимізації управління трафіком мультисервісних мереж / А.А. Коваленко // Сб. тез. доповідей по матер. 12-ї наукової конференції Харківського університету повітряних сил ім. Івана Кожедуба «Новітні технології – для захисту повітряного простору». – Х.: ХУ ПС. – 2016. – 13-14 квітня 2016. – С. 232.

8. O. Ulyanov, M. Plakhov, A. Shevtsova, O. Ulyanova, A. A. Skoryk and V. Tkachev. Testing the speed of the FFT using the NVIDIA graphic cards, in Proc. International Young Scientists Forum on Applied Physics (YSF), Dnipropetrovsk, Ukraine, 2015, p. 1.

9. C. Semeria. Supporting differentiated service classes: queue scheduling disciplines, Juniper Networks, 2001, pp. 11-14.

10. T.A. Tarray and M.R. Bhat. A nonlinear programming problem using branch and bound method, Investigación Operacional, vol. 38, No. 3, pp. 291-298, January 2018.

11. Коваленко А.А. Моделі транзитної комутації гетерогенних мереж / А.А. Коваленко, А.В. Аніщенко // Проблеми інформатизації. Тези доповідей шостої міжнародної НТК. – Черкаси: ЧДТУ; Баку: ВА ЗС АР; Бельсько-Бяла: УТіГН; Харків: НТУ «ХПІ», 2018. – 14-16 листопада 2018. – С. 78.

12. J. Edmonds and R.M. Karp. Theoretical improvements in algorithmic efficiency for network flow problems, Journal of the ACM, vol. 19, No. 2, pp. 248-264, April 1972.

13. O. Lemeshko and O. Yeremenko. Enhanced method of fast re-routing with load balancing in software-defined networks. Journal of Electrical Engineering, Vol. 68.6, pp. 444-454, 2017.

14. Коваленко А.А. Метод балансування навантаження маршрутизаторів IP-мереж / А.А. Коваленко, О.В. Купріков // Проблеми інформатизації. Тези доповідей шостої міжнародної НТК. – Черкаси: ЧДТУ; Баку: ВА ЗС АР; Бельсько-Бяла: УТіГН; Харків: НТУ «ХПІ», 2018. – 14-16 листопада 2018. – С. 78.

15. A. Dodonov, O. Gorbachyk, M. Kuznietsova. Increasing the survivability of automated systems of organizational management as a way to security of critical infrastructures, in Proc. XVIII International Scientific and Practical Conference "Information Technologies and Security" (ITS 2018), Kyiv, Ukraine, November 27, 2018, pp. 261-270.

16. G. Churyumov, V. Tkachov, V. Tokariev and V. Diachenko. Method for

Ensuring Survivability of Flying Ad-hoc Network Based on Structural and Functional Reconfiguration, in Proc. XVIII International Scientific and Practical Conference "Information Technologies and Security" (ITS 2018), Kyiv, Ukraine, November 27, 2018, pp. 64-76.

17. Ruban I.V. Provision of Survivability of Reconfigurable Mobile System on Exposure to High-Power Electromagnetic Radiation / Igor V. Ruban, Genadiy I. Churyumov, Volodymyr V. Tokarev, Vitaliy M. Tkachov // Selected Papers of the XVII International Scientific and Practical Conference on Information Technologies and Security (ITS 2017). – CEUR Workshop Processing. – Kyiv, Ukraine, November 30, 2017. – Pp. 105-111.

18. Tkachov V.M. Method for transfer of data with intermediate storage / V.M. Tkachov, V.Ye. Savanevych // IEEE First International Scientific-Practical Conference «Problems of Infocommunications. Science and Technology» (PICS&T-2014). – October 14-17, 2014. – P. 105-106.

19. Коваленко А.А. Аналіз методів навчання радіально-базисних мереж для обробки зображень / А.А. Коваленко, Буй Ван Тунг // Проблеми інформатизації. Тези доповідей шостої міжнародної НТК. – Черкаси: ЧДТУ; Баку: ВА ЗС АР; Бельсько-Бяла: УТіГН; Харків: НТУ «ХПІ», 2018. – 14-16 листопада 2018. – С. 75-76.

20. Ткачев В.Н. Моделирование инфраструктуры сбора телеметрической информации распределенных датацентров на базе аппаратных решений стандарта IEEE 802.11ac / В.Н. Ткачев, Д.Е. Митин, В.О. Лебедев // Четверта міжнародна науково-технічна конференція «Проблеми інформатизації». – Черкаси – Баку – Бельсько-Бяла – Полтава. – 2016. – С. 19.

21. Ткачов В.М. Метод передачі даних в комп’ютерній мережі проміжного зберігання даних складної інформаційної системи / В.М. Ткачов // Системи управління, навігації та зв’язку. – Полтава: Полтавський національний технічний університет ім. Ю. Кондратюка, 2017. – № 3 (43). – С. 117-119.

22. Свиридов А.С. Анализ методов моделирования сетевых ресурсов с целью формирования оптимальной полосы пропускания / А.С. Свиридов, С.О. Сосунов, А.А. Коваленко // Проблеми інформатизації. Матеріали першої міжнародної НТК. – Черкаси: ЧДТУ; Київ: ДУТ; Тольятті: ТДУ; Полтава: ПНТУ, 2013. – 19-20 грудня 2013. – С. 30.

23. Ткачов В.М. Тенденції розвитку анонімних оверлейних мереж / В.М. Ткачов, В.О. Середа, С.Є. Філіппов // Шоста міжнародна науково-технічна конференція «Проблеми інформатизації». – Черкаси-Баку-Бельсько-Бяла-Харків. – 14-16 листопада 2018 р. – С. 27-28.

24. Kuchuk G. An Approach To Development Of Complex Analysis of Cumulative Metric For Multiservice Network Security Assessment / G. Kuchuk, A. Mozhaev, A. Kovalenko // Statistical Methods Of Signal and Data Processing (SMSDP – 2010): International Conference Program. Kiev, Ukraine, October 13-14, 2010. – P. 14.

25. Kuchuk G. Method for creation of timescales in multiservice network queues analysis / G. Kuchuk, A. Mozhaev, A. Kovalenko // Гарантоздатні системи, сервіси та технології. Матеріали 5 міжнародної НТК. – Кіровоград: НВП «Радій», 2010. – 12-15 травня 2010. – С. 26.

26. Kuchuk G. Selection of algorithms for optimization problem solving by multiservice network nodes / G. Kuchuk, I. Ruban, A. Kovalenko // Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління. Матеріали шостої міжнародної НТК. – Полтава: ПНТУ; Баку: ВА ЗС АР; Кіровоград: КЛА НАУ; Харків: ДП «ХНДІ ТМ», 2016. – 21-22 квітня 2016. – С. 47.