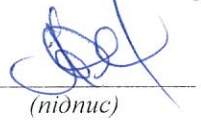
ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ТЕЛЕКОМУНІКАЦІЙ
Кафедра Мобільних та відеоінформаційних технологій

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Розробка методу геоінформаційного позиціонування телекомунікаційного обладнання»

на здобуття освітнього ступеня магістра
зі спеціальності 172 Телекомунікації та радіотехніка
освітньо-професійної програми Апаратура радіозв'язку, радіомовлення і телебачення

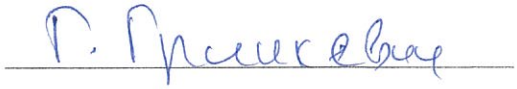
Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело



(підпис) Максим ЛЬВОВ (Ім'я, ПРІЗВИЩЕ)

Виконав: здобувач вищої освіти
групи АРДМ-61
Максим ЛЬВОВ

Керівник: Андрій ЛЕМЕШКО

Рецензент: 

Київ – 2023

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Навчально-науковий інститут телекомунікацій

Кафедра Мобільних та відеоінформаційних технологій

Ступінь вищої освіти Магістр

Спеціальність 172 Телекомунікації та радіотехніка

Освітньо-професійна програма Апаратура радіозв'язку, радіомовлення і телебачення

ЗАТВЕРДЖУЮ
Завідувачка кафедрою МВТ
_____ Наталія РУДЕНКО
« » _____ 2023 року

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Львову Максиму Олексійовичу

1. Тема кваліфікаційної роботи: «Розробка методу геоінформаційного позиціонування телекомунікаційного обладнання»,
керівник кваліфікаційної роботи Андрій ЛЕМЕШКО, PhD,
затверджені наказом Державного університету інформаційно-комунікаційних технологій від 19.10.2023 р. №145.

2. Строк подання кваліфікаційної роботи 25.12.2023 р.

3. Вихідні дані до кваліфікаційної роботи:
   1. Методи геоінформаційного позиціонування.
   2. Телекомунікаційне обладнання.
   3. Інформаційні портали геоінформаційного позиціонування.
   4. Науково-технічна література.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):
   1. Методи геоінформаційного позиціонування.
   2. Метод геоінформаційного позиціонування телекомунікаційного обладнання.
   3. Практична реалізація методу МГТО та наборів контрольних точок.
   4. Порівняння точності методу МГТО з іншими методами геопозиціонування.

5. Перелік ілюстративного матеріалу: презентація.

6. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Підбір науково-технічної літератури	19.10.23	Викон.
2.	Методи геоінформаційного позиціонування	31.10.23	Викон.
3.	Метод геоінформаційного позиціонування телекомунікаційного обладнання	15.11.23	Викон.
4.	Практична реалізація методу МГТО та наборів контрольних точок	29.11.23	Викон.
5.	Порівняння точності методу МГТО з іншими методами геопозиціонування	06.12.23	Викон.
6.	Висновки, вступ, реферат	11.12.23	Викон.
7.	Розробка презентації	18.12.23	Викон.

Здобувач вищої освіти _____ Максим ЛЬВОВ
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

Керівник кваліфікаційної роботи _____ Андрій ЛЕМЕШКО
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: 81 стор., 27 рис., 10 табл., 35 дж.

Мета роботи – розробка методу геоінформаційного позиціонування для визначення місцезнаходження телекомунікаційного обладнання.

Об'єкт дослідження - процес геоінформаційного позиціонування об'єктів.

Предмет дослідження – процес геоінформаційного позиціонування об'єктів.

Короткий зміст роботи: У магістерській роботі розглянуто сучасні методи геолокації та описано основні принципи їх роботи. Робота розділена на 2 частини - теоретичну та практичну.

Перша частина присвячена теоретичному опису цих методів та роз'ясненню основних понять і механізмів, що використовуються при визначенні фізичного розташування станції на основі її IP-адреси.

Більш вичерпна друга частина роботи стосується опису реалізації методу МГТО на мові програмування Java та його розгортання в експериментальній мережі PlanetLab. Важливою передумовою є створення так званого набору опорних точок і цілей, які необхідні для перевірки всього алгоритму методу МГТО.

Досягнуті результати в точності оцінки положення цілей методом МГТО та їх порівняння з результатами активних методів CBG, SOI та пасивних GeoIP представлені в кінці роботи.

КЛЮЧОВІ СЛОВА: ГЕОІНФОРМАЦІЙНЕ ПОЗИЦІОНУВАННЯ, ТЕЛЕКОМУНІКАЦІЙНЕ ОБЛАДНАННЯ, МГТО, PLANETLAB, КОНТРОЛЬНІ ТОЧКИ, GEOIP

ABSTRACT

Text part of the master's qualification work: 81 pages, 27 pictures, 10 table, 35 sources.

The purpose of the work - development of a method of geo-informational positioning for determining the location of telecommunications equipment.

Object of research – the process of geoinformation positioning of objects.

Subject of research – the process of geoinformation positioning of objects.

Summary of the work: The master's thesis examines modern methods of geolocation and describes the main principles of their work. The work is divided into 2 parts - theoretical and practical.

The first part is devoted to a theoretical description of these methods and an explanation of the main concepts and mechanisms used in determining the physical location of a station based on its IP address.

The more comprehensive second part of the work concerns the description of the implementation of the MGTO method in the Java programming language and its deployment in the PlanetLab experimental network. An important prerequisite is the creation of the so-called set of reference points and goals, which are necessary for checking the entire algorithm of the MGTO method.

The achieved results in the accuracy of target position estimation by the MGTO method and their comparison with the results of active CBG, SOI and passive GeoIP methods are presented at the end of the work.

KEYWORDS: GEOINFORMATION POSITIONING, TELECOMMUNICATION EQUIPMENT, MGTO, PLANETLAB, CONTROL POINTS, GEOIP

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут телекомунікацій

ПОДАННЯ
ГОЛОВІ ЕКЗАМЕНАЦІЙНОЇ КОМІСІЇ
ЩОДО ЗАХИСТУ КВАЛІФІКАЦІЙНОЇ РОБОТИ
на здобуття освітнього ступеня магістра

Направляється здобувач Львов М.О. до захисту кваліфікаційної роботи
(прізвище та ініціали)

за спеціальністю 172 Телекомунікації та радіотехніка
освітньо-професійної програми Апаратура радіозв'язку, радіомовлення і телебачення

на тему: Розробка методу геоінформаційного позиціонування телекомунікаційного обладнання

Кваліфікаційна робота і рецензія додаються.

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

Висновок керівника кваліфікаційної роботи

Здобувач Львов Максим Олексійович показав гарну теоретичну та інженерну підготовку, уміння володіти технологіями нового покоління, вміло працювати з програмно-апаратними засобами мобільного зв'язку, довідковою і науково-технічною літературою. Працюючи над завданнями, які доручались керівником, проявив ініціативність, сумлінність та хист до інженерної роботи.

Все це дозволяє оцінити виконану кваліфікаційну роботу Львова Максима Олексійовича на оцінку «відмінно» та присвоїти йому кваліфікацію «Магістр з телекомунікацій та радіотехніки».

Керівник кваліфікаційної роботи _____ Андрій ЛЕМЕШКО
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

" " _____ 2023 року

Висновок кафедри про кваліфікаційну роботу

Кваліфікаційна робота розглянута. Здобувач Львов М.О.
(прізвище та ініціали)

допускається до захисту даної роботи в Екзаменаційній комісії.

Завідувачка кафедри
Мобільних та відеоінформаційних технологій _____ Наталія РУДЕНКО
(підпис) *(Ім'я, ПРІЗВИЩЕ)*

" " _____ 2023 року

ВІДГУК РЕЦЕНЗЕНТА

на кваліфікаційну магістерську роботу

здобувача вищої освіти Львова Максима Олексійовича

на тему: «Розробка методу геоінформаційного позиціонування телекомунікаційного обладнання»

Актуальність:

Геопозиціонування перетворилося на досконалий інструмент галузі телекомунікацій. Це пов'язано з його потенціалом для масових випадків комерційного використання, наприклад, автоматизації промисловості, віддаленої експлуатації та викликів надзвичайних ситуацій. Робота базується на даних про позиціювання та розробленому методі геоінформаційного позиціонування.

Для моделювання зв'язку між вимірюваннями та положеннями обладнання використано порівняння точності методу МГТО з іншими методами геопозиціонування.

Тому тема обраних досліджень є актуальною та своєчасною.

Позитивні сторони:

Проведені дослідження свідчать про високий науково-технічний рівень використання інформаційних технологій в даному дослідженні.

Робота викладена науковою мовою, логічно й послідовно відбиває мету та поставлені в роботі завдання. Пояснювальна записка відповідає стандартам до її оформлення.

Недоліки:

1. В роботі не пояснено чи можливо визначити позицію шуканого обладнання за допомогою методу МГТО, навіть якщо не отримано відповідь ICMP.

2. Доцільно було б приділити більше уваги працям вітчизняних дослідників проблеми.

Відзначені зауваження не впливають на загальну позитивну оцінку кваліфікаційної роботи.

Висновок: кваліфікаційна робота заслуговує оцінку «відмінно», а здобувач Львов Максим Олексійович заслуговує присвоєння кваліфікації: «Магістр з телекомунікацій та радіотехніки».

Рецензент: ДТН. проф Т. Гриненко

(науковий ступінь, вчене звання) *(підпис)* *(Ім'я, ПРІЗВИЩЕ)*

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ 9
ВСТУП 10
1 МЕТОДИ ГЕОІНФОРМАЦІЙНОГО ПОЗИЦІОНУВАННЯ 12
1.1 Визначення геопозиціонування 12
1.2 Пасивні методи геопозиціонування 14
1.3 Активні методи геопозиціонування 21
1.4 Обмеження активних методів геопозиціонування 32
2 МЕТОД ГЕОІНФОРМАЦІЙНОГО ПОЗИЦІОНУВАННЯ ТЕЛЕКОМУНІКАЦІЙНОГО ОБЛАДНАННЯ 35
2.1 Основи методу МГТО 35
2.2 Перетворення затримки на відстань 37
2.3 Місце призначення 42
2.4 Непряма маршрутизація 44
2.5 Затримка "остання миля" 46
2.6 Експериментальна мережа - PlanetLab 47
3 ПРАКТИЧНА РЕАЛІЗАЦІЯ МЕТОДУ МГТО ТА НАБОРІВ КОНТРОЛЬНИХ ТОЧОК 49
3.1 Орієнтири та цілі інфраструктури 49
3.2 Орієнтири 51
3.3 Програма МГТО - практична реалізація 52
3.4 Вимірювання в реальній мережі 63
4 ПОРІВНЯННЯ ТОЧНОСТІ МЕТОДУ МГТО З ІНШИМИ МЕТОДАМИ ГЕОПОЗИЦІОНУВАННЯ 69
4.1 Результати методу МГТО 69
4.2 Результати методу МГТО в порівнянні з іншими методами 69
4.3 Графічне представлення результатів із використанням CDF 71
4.4 Графічне представлення результатів KDE 73
ВИСНОВКИ 76
ПЕРЕЛІК ПОСИЛАНЬ 78
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ 82

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

CBG	Constraint-based Geolocalization
CDF	Cumulative distribution function
CentOS	Community ENTerprise Operating System
DMOZ	Open Directory Project
DNS	Domain Name System
GPS	Global Positioning System
IANA	Internet Assigned Numbers Authority
ICMP	Internet Control Message Protocol
ISP	Internet service provider
JSON	JavaScript Object Notation
KDE	Kernel Density Estimation
LBS	Location Based Services
LSA	Link-state advertisement
NAT	Network Address Translation
OSPF	Open Shortest Path First
RFC	Request for Comments
RIPE NCC	Réseaux IP Européens Network Coordination Centre
RIR	Regional Internet Registry
RTT	Round-Trip Delay Time, ms
SOI	Speed of Internet
SSH	Secure Shell
SSID	Service Set Identifier
TBG	Topology-based Geolocation
TCP	Transmission Control Protocol
VoIP	Voice over IP
VPN	Virtual private network

ВСТУП

Сьогодні геопозиціонування перетворилося в інструмент, про який складно було уявити раніше, в період перших думок про створення Інтернет-мережі.

Життя, з географічної точки зору, з кожним днем стає все складніше - мегаполіси ростуть, потреби людей все різноманітніше, тому в бізнесі, компанії, які будуть користуватися простором максимально ефективно, доб'ються успіху. У цьому їм допоможе геопозиціонування. Існують кілька об'єктивних факторів підвищення популярності геолокаційні систем, серед яких: особиста вигода; розширення меж соціальності; залученість в процес рекомендацій, оцінок, рад.

Картографія перетворилася в чудо - з шалено великих аркушів паперу в те, що вміщується на долоні. На сьогодні, карти використовуються не тільки для пошуку культурного центру в Парижі, але щоб мати можливість знайти все: найближчий готель, аптеку або лікарню, продуктовий магазин, університет або школу, місце проведення концерту Pink в Сан-Франциско.

Трансформація технології на цьому не зупиниться. Це стане краще і точніше, як ніколи.

У 1989 році в США був запущений самий перший супутник GPS для комерційного використання. Розвиток цієї технології став швидким з цього моменту. Всього за 10 років Benefon Inc. стала першою компанією, що представила перший комерційно доступний мобільний телефон з вбудованим GPS. Сьогодні 2023 рік і пройдено довгий шлях з часів Benefon. Сьогодні можете знайти загублених домашніх тварин з нашийниками, дітей, батьків, які мають вбудований GPS в розумні пристрої. Зараз можна знайти місце розташування вашого улюбленого гамбургера і його доставку, не встаючи зі свого дивана. Отже використання цієї технології присутнє індивідуально в повсякденному житті.

Геопозиціонування використовується в набагато більших масштабах - підприємствами для відстеження і націлювання на потрібних користувачів, щоб збільшити продажі і зробити охоплення потрібної аудиторії клієнтів.

Мета роботи – розробка методу геоінформаційного позиціонування для визначення місцезнаходження телекомунікаційного обладнання (МГТО).

Об'єкт дослідження – процес геоінформаційного позиціонування об'єктів.

Предмет дослідження – методи геоінформаційного позиціонування для визначення місцезнаходження телекомунікаційного обладнання.

Методи дослідження. Для досягнення поставлених цілей в магістерській роботі використано: методи теорії інформації та передавання сигналів; методи теорії розповсюдження електромагнітних хвиль; методи геопозиціонування; методи комп'ютерного моделювання; математичної статистики.

1 МЕТОДИ ГЕОІНФОРМАЦІЙНОГО ПОЗИЦІОНУВАННЯ

1.1 Визначення геопозиціонування

Геопозиціонування – це метод, за допомогою якого отримується географічне розташування обраного об'єкта (наприклад, комп'ютера чи мобільного телефону) на основі наявних даних. Місцезнаходження можна визначити більш-менш точно на основі IP-адреси, мережі Wi-Fi, мобільного телефону або приймача GPS та їх подальшого порівняння з конкретною базою даних або даними, отриманими в результаті вимірювань у мережі. Перший метод використовується так званими пасивними методами геопозиціонування, а другий – методами геопозиціонування, які називаються активними.

Потім виявлене місцезнаходження користувача (станції) використовується низкою програм та службами LBS (Location Based Services). Зростає інтерес до цих послуг. Наприклад, послуга Foursquare зросла з початкових 100 000 учасників (квітень 2009 р.) до сьогоднішніх 15 000 000 (вересень 2023 р.). Один з найбільших гравців у "соціальних мережах", компанія Facebook, також бачить потенціал у цих послугах, оскільки не так давно вона придбала компанію Gowalla, яка раніше конкурувала з послугою Foursquare [1].

Геопозиціонування на практиці та послуги LBS

Хоча цих служб існує велика кількість, тут перелічено лише представники найбільших. Наприклад, сервіс розваг Foursquare (www.foursquare.com) або Facebook Place. Розроблений для мобільних телефонів, Foursquare базується на обміні цікавими точками між користувачами.

Наприклад, можна оцінити хороший ресторан, а потім поділитися його позицією з іншими членами служби. Facebook Place використовується на однойменному порталі, щоб повідомити, з якого місця зареєстрований користувач написав свою публікацію [2].

Іншим способом використання відомого місцезнаходження користувача є націлювання вмісту веб-сторінок за місцем розташування. Це використання

спільно називається геотаргетингом. Якщо веб-сайт знаходить місцезнаходження користувача, він може змінити мову відображення, а також запропонувати користувачеві відповідний вміст. Наприклад, для користувачів, що здійснюють доступ з Києві, на веб-сайті компанії, що займається міжнародною продажею маршрутизаторів, будуть відображатися лише центри в Україні або лише в околицях Києва. Але це ідеальний випадок. Це дуже багато що залежить від того, як ви отримаєте позицію. Зазвичай у цій області використовують пасивні методи геопозиціонування, що означає пошук IP-адреси в певній базі даних. У разі загальнодоступних баз даних (наприклад, RIPE) місцезнаходження визначається на рівні країни, а при використанні платної бази даних (пропонує базу даних з точністю до регіонів) місцезнаходження може бути вказане до рівня міста, або частини міст.

Інші можливі варіанти використання геопозиціонування:

• Цільова реклама. Користувачі відображаються в рекламі браузера з пропозиціями речей/послуг місцевих компаній.

• Пошук. Пошукова система, така як Google, повертає географічно відповідні результати, коли вводиться слово "ресторан" у вікно пошуку. Користувача повертатимуть до знайдених ресторанів лише у його країні, місті, а не, наприклад, на іншому кінці світу [3].

• Спам можна фільтрувати повідомлення від країн, де немає спаму політики або знаходиться на дуже низькому рівні.

• VoIP- телефонія

Постачальники пропонують своїм VoIP-серверам доступ з будь-якого місця, тому знайти місце абонента не так просто, як із традиційним телефоном. У разі екстреного виклику рятувальні служби повинні мати можливість визначити позицію викликаної сторони. Тому кожен реєстратор VoIP має центральний реєстр, який зберігає дані про місцезнаходження телефонів VoIP, отримані від клієнтів під час першого замовлення послуг. В основному це біржі або, можливо, встановлені набори.

Якщо користувач використовує перевагу з'єднання VoIP - мобільність, то

визначення позиції є більш складним і повинен покладатися на один із методів геопозиціонування. Той факт, що деякі цифри є географічними, також може допомогти з локалізацією. Це означає, що його префікс ua прив'язаний до певної області (в Україні це регіони) [4].

1.2 Пасивні методи геопозиціонування

Пасивні методи геопозиціонування використовують доступні бази даних, з якими вони порівнюють отримані дані (наприклад, IP-адресу), і таким чином отримують позиції цільових станцій. Точність спадкоємця залежить від якості і цілісності записів в доступних базах даних.

За даними, з якими вони працюють, їх можна розділити на кілька типів.

Локалізація на основі IP-адреси

У глобальній мережі всі станції мають власну адресу (32-бітну), яка їх ідентифікує. Розподіл IP-адрес регулюється Інститутом присвоєних номерів в Інтернеті (IANA). Він представлений в окремих частинах світу так званим RIR (Regional Internet Registry). Для Європи та Близького Сходу це RIPE NCC (Центр координації європейських мереж IP). Подальшим розподілом є змога потрапити до компаній, які безпосередньо спілкуються з клієнтами в окремих країнах. Ці компанії поширюють діапазони IP-адрес місцевим провайдерам, а потім – окремим клієнтам. Звичайно, кожна з компаній веде базу даних із записами того, кому була призначена та чи інша IP-адреса. Найвідомішими вільно доступними базами даних є Whois та GeoIP (вона пропонує лише базу даних державного рівня безкоштовно) [5].

GeoIP

Послуга геопозиціонування GeoIP розроблена та управляється американською компанією MaxMind, яка також дбає про заповнення та перевірку однойменної бази даних. Принцип роботи цієї послуги простий. Користувач або корпоративний клієнт купує базу даних за окрему плату, де, простіше кажучи, знаходяться географічні місця розташування різних IP-адрес. Звичайно, база

даних не містить переліку всіх існуючих IP-адрес та їхніх позицій, але логічно поділена відповідно до географічних районів з відповідними блоками адрес. Цей показник не тільки пришвидшує пошук, але й зменшує розмір отриманої бази даних.

Наприклад, придбана база даних може підключитися до веб-сайту та скористатися пошуковим скриптом або програмою, щоб з'ясувати, де користувач підключився. Наприклад, клієнт з Німеччини, який звертається до електронного магазину і користується послугами GeoIP, автоматично перемкне сайт на німецьку, а також встановить ціни на товари в євро. Основною проблемою цього методу є надійність даних. Більшість даних отримується з форм, які користувач повинен заповнити під час різних реєстрацій. Оскільки певний відсоток отриманих таким чином даних може бути неправдивим, MaxMind все одно повинен зберігати їх у базі даних GeoIP.

MaxMind повідомляє про точність своєї бази даних на рівні 99,8% на рівні країни, 90% на рівні штату та 80% на рівні міста, де шукана мета лежить в радіусі 40 км.

Навпаки, великою перевагою є швидка реакція на запит про розташування, а також регулярне оновлення бази даних. Для старших версій оновлення виконується щотижня, в середньому від 1 до 2% бази даних змінюється або відновлюється [5].

Локалізація на основі системи DNS

По-перше, корисно буде коротко переглянути, що приховано під абревіатурою DNS (Система доменних імен). Простіше кажучи, DNS – це група серверів, яка використовується для перекладу IP-адрес (IPv4 - 77.75.73.138, IPv6 - 2a02: 598: 2: 0: 0: 0: 0: 109) запам'ятовуються доменні імена (IPv4 - mapy.seznam.com, IPv6 - ipv6.mapy.seznam.com) і навпаки. Принцип показаний на рис.1.1, де користувач ввів адресу www.mapy.seznam.com у своєму браузері.

На додаток до адреси IPv4 (або доменного імені), сервери DNS можуть повертати іншу корисну інформацію. Це, наприклад, записи типу (відповідь на www.seznam.com, додаток nslookup у Windows подано в дужках) [7]:

• A (запис адреси)

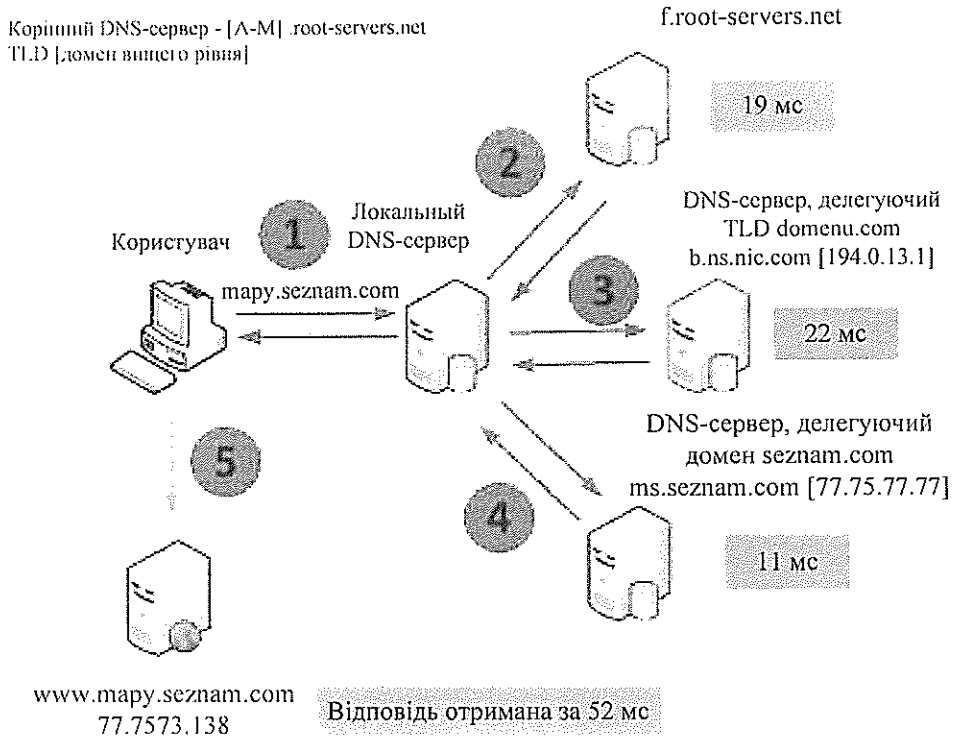


Рис. 1.1. Принцип DNS

Тут вказана вже згадана адреса IPv4. Для даного прикладу - 77.75.76.3.

• AAAA (запис адреси IPv6)

Адреса IPv6 вказана тут. На цей приклад відповіді немає. Компанія Seznam.com, оскільки за цією адресою не підтримує IPv6.

• CNAME (канонічний запис імен)

Тут показано інші можливі імена для того самого домену. У даному випадку значення не вказано.

• MX (запис обміну поштою)

Вказує ім'я домену поштового сервера, до якого пошта призначена для цього домену буде доставлена. У тому випадку, є резервне копіювання поштових серверів. Вони розрізняються за пріоритетом – найбільше число в назві = найнижчий пріоритет.

Для даного випадку 60 для mx60.seznam.com, 50 для mx50.seznam.com.

• NS (Authoritative server name)

Запис визначає доменне ім'я DNS-сервера, де знаходиться інформація про

домен, або у вигляді IP-адреси, що належить цьому доменному імені, або як посилання на інший доменний сервер. Для вказаного випадку: ns.seznam.com., Ms.seznam.com.

• Іншими записами є, наприклад, TXT (необов'язкова примітка), SOA (інформація в ньому використовується для зонних передач між DNS-серверами), DNSKEY (цей ключ використовується для асиметричного шифрування DNS-записів – користувач знає, чи відповідає відповідь DNS-сервера, або ні) та інші [9].

Найважливіший запис для цієї роботи ховається під LOC (LOC Records). Цей запис містить інформацію про місцезнаходження домену. Далі наведено приклад домену myvideostore.com. Потрібно використовувати Linux (nslookup не підтримує записи LOC [8]). У командному рядку вводиться dig LOC myvideostore.com.

Відповідь буде у форматі <ім'я домену> <TTL> <клас> LOC (d1 [m1 [s1]] {"N" | "S"} d2 [m2 [s2]] {"E" | "W"} alt ["m"] [siz ["m"] [hp ["m"] [vp [„m"]]]]). Де окремі символи позначають [10].

Таблиця 1.1.

Змінні у відповіді DNS LOC

Змінна	Значення	Опис
dl	[Від 0 до 90]	широта - градуси
d2	[Від 0 до 180]	довгота - градуси
m1, m2	[Від 0 до 59]	хвилин ширина/довжина
sl, s2	[Від 0 до 59 999]	секунди ширина/довжина
alt	[-100000,00 до 42849672,95]	висота в метрах
siz, hp, vp	[Від 0 до 90000000,00]	розмір еліпсоїда (розмір) та дисперсія (точність по горизонталі, вертикалі) у метрах

Тому ця інформація відображається в командному рядку. Див. рис. 1.2. У

Wireshark цей запит фіксується та показаний на рис. 1.3.

```
[root@localhost ~]# dig LOC myvideostore.com

; <<>> DiG 9.7.3-P3-RedHat-9.7.3-8.P3.el6 <<>> LOC myvideostore.com
;; global options: +cmd
;; Got answer:
;;  ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 48436
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 0

;; QUESTION SECTION:
;myvideostore.com.              IN      LOC

;; ANSWER SECTION:
myvideostore.com.       5       IN      LOC     35 59 58.910 N 86 35 18.650 W 186.00m 1m 10000m 10m
```

Рис. 1.2. Запис DOC LOC

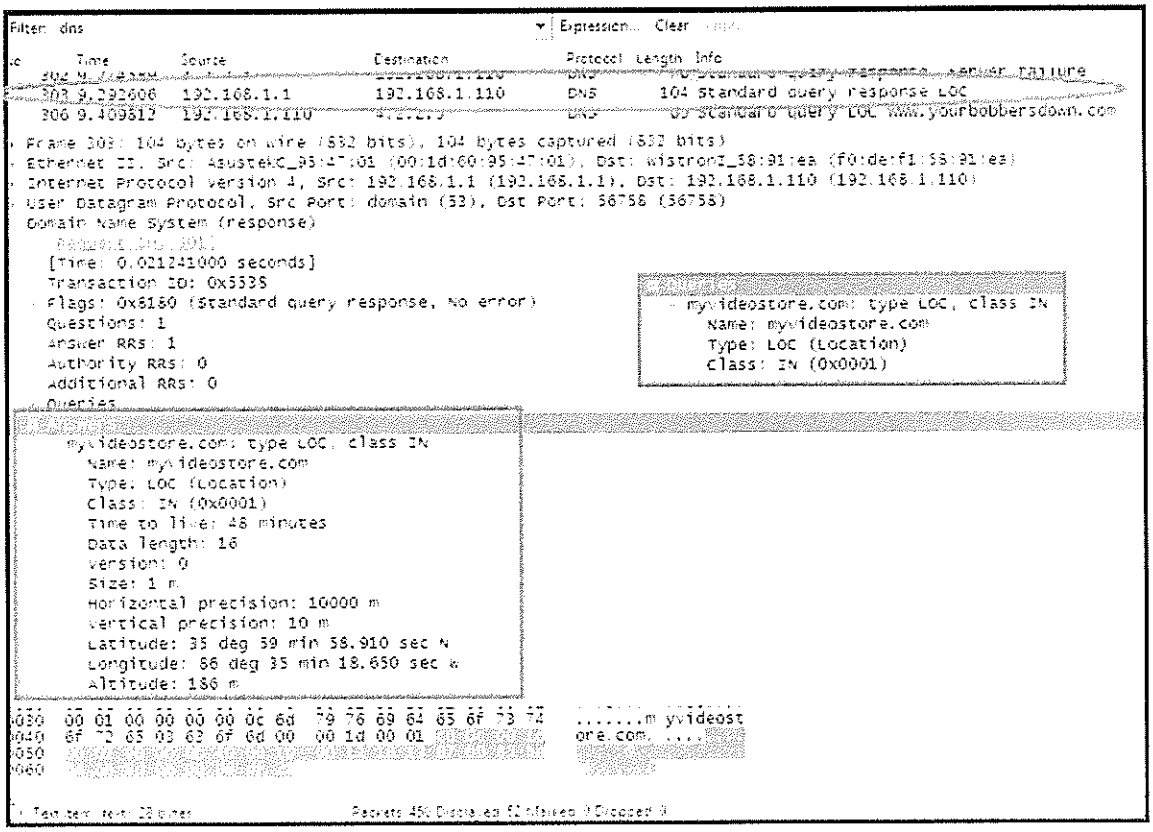


Рис. 1.3. DNS-запит/відповідь (DNS QUERY/DNS QUERY RESPONSE), записаний у Wireshark

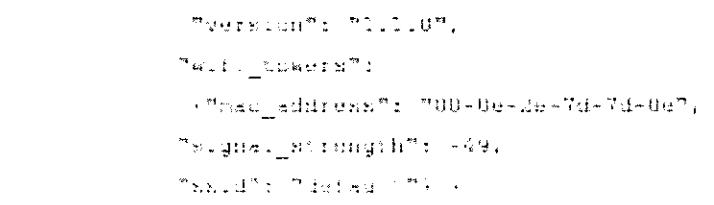
Переваги та недоліки подібні до інших методів, що використовують бази даних. Найважливішим елементом тут знову є достовірність та своєчасність баз даних. На жаль, ці записи, які почали використовуватись приблизно в 1997 р., з міркувань безпеки не поширились до такої міри, щоб можна було точно

визначити більшу кількість станцій.

Розташування на основі мережі Wi-Fi

У 2010 році відбулася велика справа Google, яка збирала інформацію про бездротові мережі під час картографування території для Google Street View. Незважаючи на початкові проблеми, компанії вдалося створити величезну базу даних бездротових точок (MAC і джерсі та SSID) разом із їх позицією. Ця база даних постійно оновлюється за допомогою «розумних» телефонів Android. Ці згадані телефони регулярно оглядають своє оточення, а потім надсилають дані про місцезнаходження та потужність сигналу доступних мереж Wi-Fi разом із унікальним ідентифікатором телефону та їх місцезнаходженням, що визначається, наприклад, модулем GPS у телефоні. База даних також може виправити будь-які неточності, що виникли, наприклад, переміщенням маршрутизатора. Виправлення виконується, коли MAC-адреса маршрутизатора кілька разів записується в іншому місці [11].

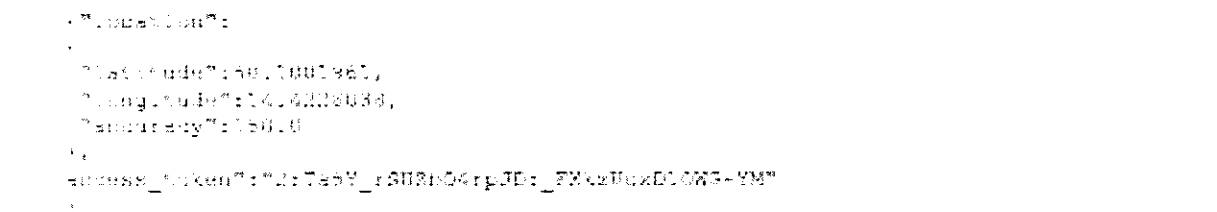
Цю базу даних можна використовувати для пошуку окремих станцій. Наприклад, якщо треба з'ясувати розташування користувачів, які підключаються до веб-сайту, просто потрібно розмістити на ньому простий код (мова програмування - JavaScript) і викликати функцію navigator.geolocation.getCurrentPositio. Йому передається єдиний параметр, а саме функція, яка викликається для визначення положення, і яка передається об'єкту із виявленою інформацією про навколишні мережі Wi-Fi. Кінцевий користувач після відображення з'єднання у браузері запитає інформацію, необхідну для забезпечення локалізації. Після затвердження запит надсилається за допомогою протоколу HTTP - POST на адресу за допомогою захищеного протоколу HTTPS (Hypertext Transfer Protocol Secure):



Щоб побачити назву Wi-Fi мережі (тобто. SSID - Service Set Identifier) з

MAC - адреса бездротового пристрою, і сили сигналу. Якщо навколо користувача транслюється кілька мереж Wi-Fi, інформація про них також буде надіслана, і отримана позиція буде більш точною.

Відповідь з отриманою позицією має вигляд:



Відповідь буде містити інформацію про широту/довготу і точність повідомляється в метрах. Це використовується для подальших запитів щодо пошуку того самого користувача. Відправлений запит також було зафіксовано у Wireshark і показано на рис. 1.4 [12].

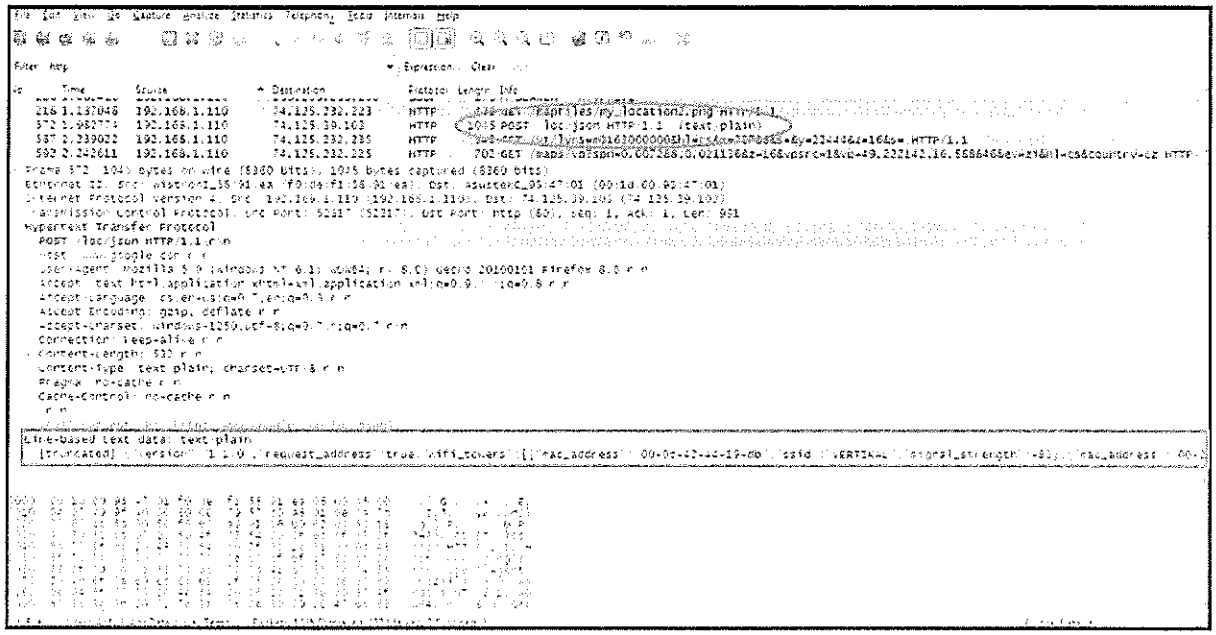


Рис. 1.4. Надсилання запиту про місцезнаходження через Wi-Fi на http://www.google.com/loc/json

Можна відобразити розраховану позицію за допомогою сервісу google maps (див. рис. 1.5). Через те, що радіус дії мереж Wi-Fi досить малий (100-150 метрів у місті), отримане положення є відносно точним.

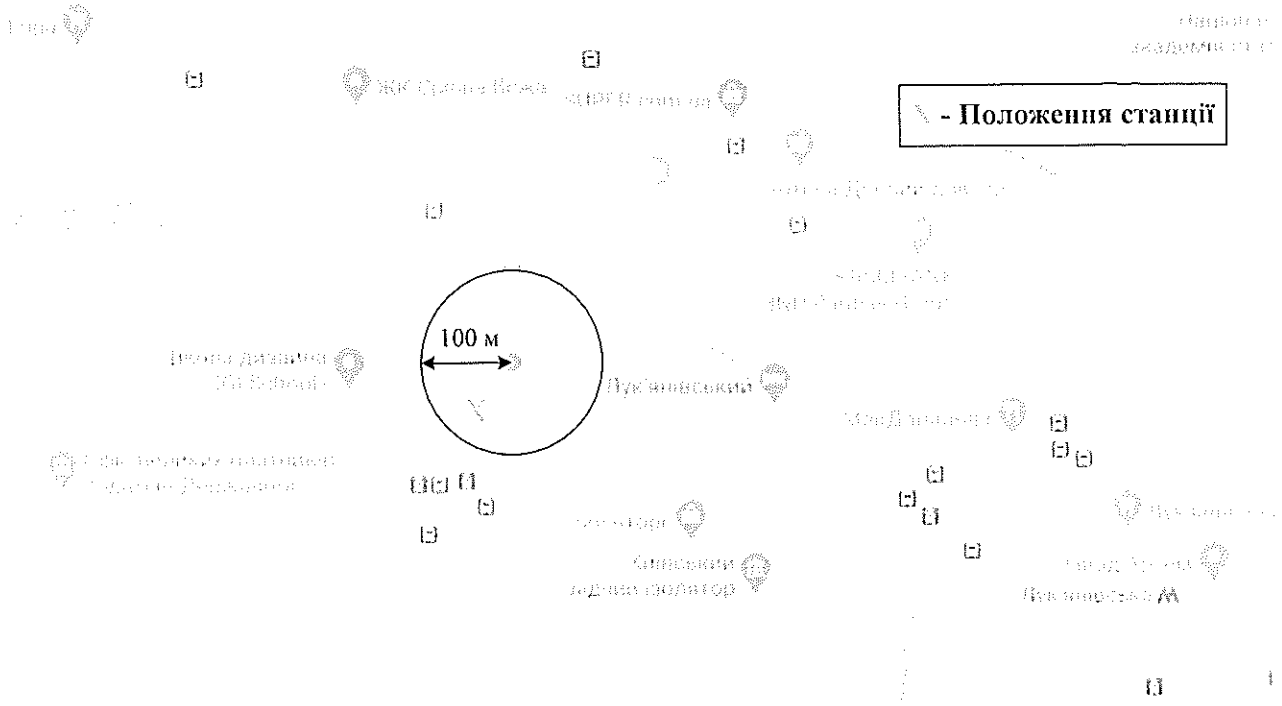


Рис. 1.5. Геопозиціонування за допомогою мереж Wi-Fi (http://maps.google.com/)

1.3 Активні методи геопозиціонування

На відміну від описаних раніше методів, ці методи відрізняються тим, що активно проводять вимірювання в мережі. Таким чином, вони не порівнюють отримані дані з базою даних, а використовують складні математичні формули та набір опорних точок для обчислення положення цілі (розташування однієї точки) або області, в якій знаходиться цільовий вузол (розташування області). У цьому розділі розглядаються методи, які використовують значення затримки на рядках.

Щоб ознайомитися з проблемою, доцільно спочатку пояснити, яка затримка, що її спричиняє та пояснити деякі терміни, які з'являються далі в тексті.

Визначення затримки та її видів

Затримку можна визначити як час, що проходить від відправки повідомлення вихідним вузлом до його отримання на вузлі призначення. Можна розділити його відповідно до місця походження на затримки, спричинені передачею носія, активними елементами або станцією джерела/призначення [15].

Затримка передачі

Ця затримка виникає, коли світловий або електричний сигнал передається

через саме середовище. Його розмір визначається використовуваним середовищем і, звичайно, довжиною. Загалом можна відзначити, що сигнал поширюється в оптичному волокні зі швидкістю 0,65 * с, де с – швидкість світла і 0,75 * с в металевій лінії. На відстанях до 1000 км ця затримка незначна. У випадку більших відстаней ця затримка повинна бути врахована та включена в остаточний розрахунок [15].

Затримка, спричинена активними елементами в мережі

Під час своєї подорожі пакет (точніше, блок даних) повинен пройти через ряд активних елементів, які його аналізують, а потім далі вирішити, в якому напрямку рухатися. Кожна з цих операцій (маршрутизація) додає свою частку до загальної затримки.

Пакет надходить до активного елемента, і оскільки він може одночасно обробляти лише одну одиницю даних, він потрапляє в чергу і чекає обробки. Після того, як пакет потрапляє в маршрутизатор/комутатор, він аналізується і відправляється на вихідний порт за допомогою внутрішніх процесів. Якщо вихідний рядок перевантажений, він знову включається у відповідний буфер. Наприклад, для прискорення процесу використовується цілеспрямований відбір за допомогою алгоритму кругового обробітку або пріоритетна обробка. Незважаючи на складні методи та техніки, пропоновані цими пристроями, вони найбільше впливають на результуючу затримку. Звичайно, це також залежить від кількості пристроїв, які обробляють пакет на своєму шляху [16].

Затримка, спричинена станцією джерела/пункту призначення

Інша затримка, яка впливає на результуючий час, це та, яка виникає під час обробки та надсилання даних до станції призначення/джерела [17].

Затримки також можна розділити на типи:

• Односторонні

Час між відправленням пакету до джерела та отриманням його до місця призначення.

• Двонаправлені (RTT - Round-trip delay time)

Час, який включає в себе шлях пакета назад і вперед, а також їх обробку на

цільовій ділянці.

Активні методи геопозиціонування працюють найчастіше при цьому типі затримки. Точніше, лише зі своєю половиною. Тому він передбачає так звану симетричну затримку, коли час передачі до пункту призначення такий же, як і час від нього.

Великою перевагою є той факт, що цільова станція реагує на вимірювання автоматично, тому немає необхідності запускати на ній програму підтримки.

Протокол повідомлень про керування Інтернетом (ICMP)

Цей протокол використовується для повідомлення про особливі ситуації в мережі. Це є, як правило, повідомлення до вихідного вузла або маршрутизатора, що сталася помилка під час передачі пакета IP. Ці помилки спричинені, наприклад, тим, що маршрутизатор повинен скинути пакет, коли TTL перевищений, або відсутністю запису в таблиці маршрутизації про мережу призначення, до якої направляється пакет [18].

Щоб визначити повідомлення про помилку, використовують два однобайтові поля (Тип, Код) у заголовку пакета ICMP, розмір яких завжди становить 8 байт. У таблиці 1.2 вибрано лише деякі типи пакетів ICMP [19].

Повідомлення Echo response (0) та Echo request (8) використовуються мережевою програмою під назвою ping (Packet InterNet Groper). Ця програма використовується для визначення активності цільового вузла, а також відображає двонаправлену затримку (RTT).

Методи визначення цілі як єдиної точки

Результатом роботи цих методів є отримання положення однієї точки - цілі. Наприклад, метод GeoPing або GeoTrack.

GeoPing Метод використовує три типи станцій для своєї роботи. Це так звані зонди, які вимірюють затримку. Крім того, орієнтири (англійською мовою називаються landmarks), для яких відомо їх географічне положення. І як остання цільова станція/вузол (цільовий вузол), позиція, яку шукають.

Загальний принцип методу заснований на порівнянні виміряних часів між зондами, орієнтирами, цільовою станцією і показаний на рис. 1.3 [20].

Таблиця 1.2.

Типи та коди повідомлень ICMP

Тип (грубий розподіл)	Код (із зазначенням поділу)	Детальний опис
0	0	відповідь на запит на відповідь (ехо-відповідь)
3	вузол призначення недоступний (пункт призначення недоступний)	
	3	недоступний порт - UDP (недоступний порт)
	4	фрагментація пакетів потрібна, але не дозволена (встановлено фрагментацію та не фрагментовано)
	5	невдала маршрутизація (вихідний маршрут не вдався)
8	0	ехо-запит
11	таймер минув (час перевищено)	
	0	Значення TTL перевищено - генерується маршрутизатором
	1	пакет не може бути відновлений, наприклад, один із фрагментів - генерований вузлом - відсутній

Зонди використовують пінг ICMP для отримання значень затримки між ними та окремими орієнтирами. Так званий вектор затримки будується з виміряних часів – названий DVX, де x – орієнтир. Подібний вектор затримки також побудований для затримки між зондами та цільовим вузлом – DVC. Потім отримані вектори DVX порівнюють з вектором DVC.

Отримане позиціонування враховує той факт, що якщо вимірювані затримки подібні, географічне положення подібне. Таким чином, положення станції призначення визначається положенням орієнтира, який має подібні значення затримки у векторі затримки.

З самої природи дії цього методу ясно, що на його точність буде сильно впливати вибір та кількість орієнтирів. На жаль, незважаючи на велику кількість цих пунктів, географічна відстань від пункту призначення може сильно

відрізнятися. Це тому, що ніколи неможливо охопити всі варіанти пошуку пункту призначення. Найточніших результатів можна досягти, рівномірно розподіливши більшу кількість точок орієнтації. Їх кількість повинна бути якомога більшою, а знання позиції якомога точнішим. Кількість зондів має бути від 7 до 9, оскільки при цій кількості метод працює найбільш ефективно [21].

Вектори затримки для прикладу на рисунку 1.6 можуть мати значення, наведені в таблиці 1.3 DV1 отримують від першого орієнтира - рис. 1.6. частина 1, DV2 від другого орієнтира - частина 2, DVC від зондів до цілі - частина 3.

Таблиця 1.3.

Вектори затримки

	Зонд 1 [мс]	Зонд 2 [мс]	Зонд 3 [мс]	Зонд 4 [мс]
DV1	89	101	130	121
DV2	135	136	35	38
DVC	90	99	135	122

Значення DV1 та DVC подібні, тому остаточне положення визначається першим орієнтиром (рис. 1.6. частина 1).

GeoTrack метод використовує програму під назвою traceroute. Ця програма працює, надсилаючи пакети - запити ICMP (код 0, тип 0) із поступовим збільшенням терміну служби - TTL. Його принцип можна побачити на рис. 1.7.

Спочатку три пакети зі значенням TTL = 1 надсилаються на IP-адресу призначення. Потім значення TTL збільшується до TTL = 2 і знову відправляються три пакети. Маршрутизатор, на який надходять пакети, зменшить значення TTL на одиницю. Потім він аналізує заголовок IP, якщо TTL = 0, надсилає відповідь ICMP (код 10, тип 0) ресурсів. На рисунку 1.7 рядки ресурсів позначені чорним кольором, а також видно, як додаток запитує доменні імена маршрутизаторів, використовуючи зворотні запити DNS (червона стрілка). За допомогою цієї програми можна "виявити та зв'язатися" з усіма маршрутизаторами, які перебувають на шляху до адреси призначення, та

дізнатися їх доменні імена.

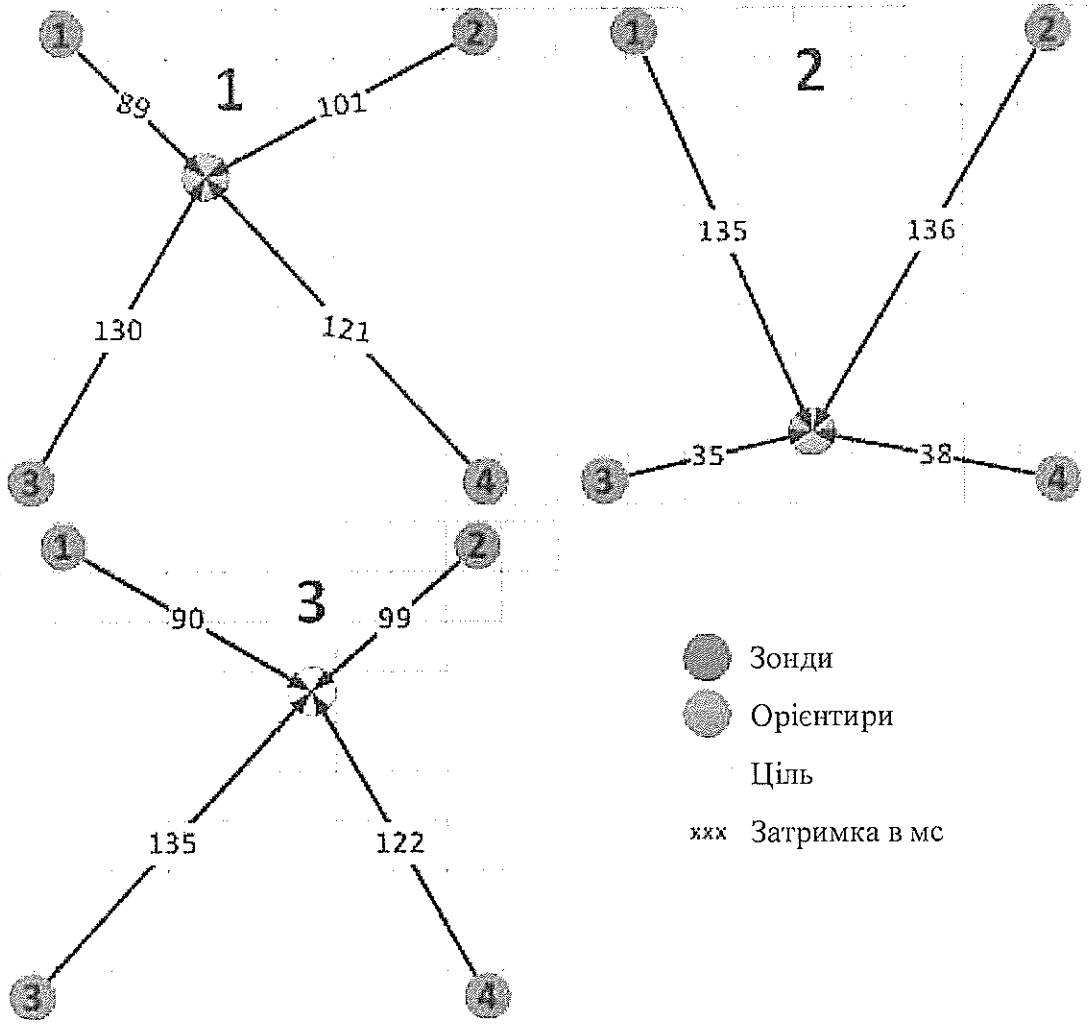


Рис. 1.6. Загальний принцип методу GeoPing

Саме ці доменні імена використовує метод GeoTrack для отримання позиції шуканого вузла. Доменні імена часто зберігають інформацію, яка допомагає визначити місце призначення. Прикладами є міжнародний код країни, поштовий індекс або код міста. Цю інформацію надає власник маршрутизатора. Більшість з них є місцевими провайдерами Інтернету, і інформація в основному використовується для її адміністрування [22].

Якщо доменне ім'я маршрутизатора надає певну географічну інформацію, маршрутизатор позначено як впізнаваний GeoTrack. Положення останнього використовуваного маршрутизатора на шляху до пункту призначення також визначає положення пункту призначення.

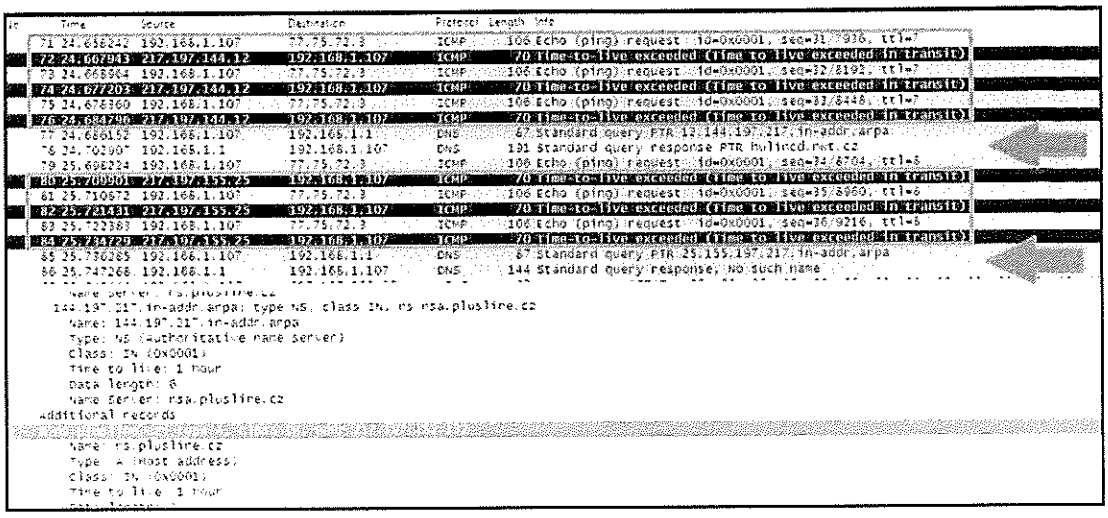


Рис. 1.7. Витяг із програми Traceroute

Існує кілька недоліків цього методу:

• Інформація, що міститься в доменному імені, не має єдиного стандарту. Наприклад UPC Ukraine, як називає свої маршрутизатори цифровим кодом, який представляє більш точну позицію та абревіатуру міста, в якому знаходиться маршрутизатор. Повна назва міста відображається в маршрутизаторах, що експлуатуються NWT.

• Останній придатний для використання маршрутизатор може бути дуже далеким від фактичного місця призначення.

• Інформація про положення маршрутизатора може бути неправдивою.

Як спрощений приклад методу GeoTrack, можна ввести команду tracert на www.zyvideostore.com. Останній маршрутизатор перед вузлом призначення має назву – INGRAM- ENTE.car2.Nashville1.Level3.net. З назви зрозуміло, що вона розташована в місті Нашвілл, а також позиція сервера, який шукається. Цей факт було підтверджено електронним запитом. Отже, у цьому випадку метод спрацював би.

Методи визначення місцеположення попадання цілі.

Ці методи відрізняються від раніше описаних тим, що результатом їх роботи є визначення області, в якій цільова станція, швидше за все, буде знаходитися. Це, наприклад, CBG (геопозиціонування на основі обмежень), TBG (геопозиціонування на основі топології) або методи МГТО (описані окремо в

розділі 4).

Наріжним каменем цих методів є вимірювання затримки між орієнтиром та ціллю. Отриманий час перетворюється на географічну відстань. Тут слід зазначити, що кожен із методів підходить до цього питання дещо по-різному. Деякі «спрощують» перетворення, тому отримана позиція оцінюється не дуже точно (наприклад, метод CBG, який використовує лише лінійні функції для обчислення так званої «найкращої лінії»).

Якщо позначити отриману географічну відстань як d, то положення станції призначення знаходиться в безлічі точок, які лежать на одній і тій же відстані d від фіксованої точки. Ця фіксована точка являє собою контрольну точку, з якої проводилось вимірювання. В геометричному вигляді це коло.

Очевидно, що неможливо отримати положення цільової точки, якщо лише значення з вимірювань відомі лише з однієї опорної точки. На практиці використовується понад 90 орієнтирів, з яких виходить затримка до мети. Згодом збирається набір кіл і з їх перетинів збирається суцільна область, в якій знаходиться цільовий вузол [22].

Метод CBG використовує лише орієнтири, для яких позиція відома. Кожен із орієнтирів вимірює затримку до цілі і множить цю затримку на коефіцієнт r, створюючи таким чином набір точок, що представляють положення цільового вузла. Геометрично ця межа може бути виражена колом, на якому лежать позиції цільової точки, з центром у поточному орієнтирі. Згодом виконується так звана мультіратерація (подібно до системи GPS) і визначається область, в якій цільовий вузол, швидше за все, знаходиться. [23]

Хоча процедура попереднього тексту може бути настільки простою, на практиці це не так.

Знаходження коефіцієнта r

В ідеальній мережі, де існують лише прямі зв'язки, значення r було б рівним 2/3 швидкості світла. Це відповідає швидкості поширення сигналу в оптичному волокні. У реальних мережах слід передбачити низку проблем, починаючи з непрямої маршрутизації, затримок на маршрутизаторах і закінчуючи

перевантаженими лініями. Через ці події, які "подорожуючий пакет сповільнює ", значення r має бути певним чином відрегульоване, щоб визначити відповідну залежність географічної відстані від затримки. Це базується на механізмі, який можна було б назвати калібруванням орієнтаційних вузлів [20].

• Калібрування, що вказує на їх точки

Кожен орієнтир вимірює затримку з іншими орієнтирами. Потім він призначає відстані до цих точок цим значенням і вставляє їх у графік, де вісь x виражає відстань, а вісь y – затримку. (Слід звернути увагу, що місце розташування всіх орієнтирів відомо).

Щоб отримати відповідний коефіцієнт, необхідно знайти рішення, яке найкраще фіксує співвідношення між затримкою та відстанню, прийнятним для всіх орієнтирів. Рішенням є пряма лінія, над якою розташовані всі значення затримки до інших орієнтирів. Цей рядок називається найкращим рядком і представляє значення коефіцієнта r [20].

Примітка: Лінійне рішення методу CBG – лінія замінюється методом МГТО на складне рішення. Лінія замінюється функцією, яка складається з часткових простих функцій, що слідують одна за одною в точках контакту.

Отримавши коефіцієнт r, тоді можна перетворити затримку, виміряну окремими орієнтирами, в цільову точку, в граничні межі. Межі обмежень представлені колами, які вказують на можливе положення цілі. Потім більш чітке визначення області ураження цілі визначається за допомогою багатобічного механізму [20].

Багатосторонність

Багатофункціональний – це механізм, який, як уже зазначалося, дозволяє знайти невідому точку на основі часу та відстані, яку проходить сигнал від цілі до точок з відомим положенням. У даному випадку це орієнтири, кожна з яких має межовий кордон. Результатом є пошук області, в якій знаходиться цільовий вузол. Принцип показаний на рис. 1.8.

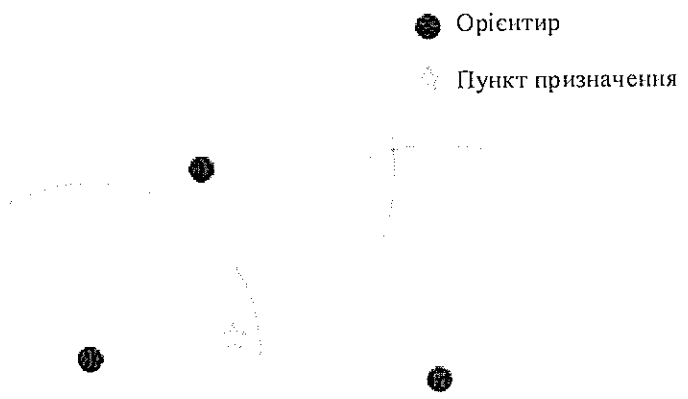


Рис. 1.8. Метод CBG

На рисунку 1.8 показано визначення площі, в якій ціль найімовірніше лежить. Можливим рішенням отримати цільову точку з цієї області є вираз її у вигляді багатокутника з вершинами в точках, де перетинаються окремі кола (межі обмеження), та визначення її центру ваги.

Метод SOI дуже схожий на попередній метод CBG. У ньому працюють подібні принципи, наприклад, багатофункціональність. Він відрізняється створенням так званої найкращої лінії, яка використовується для перерахунку відстані від затримки.

Авторам цього методу вдалося виміряти, а потім переконатися, що з великою ймовірністю (80%) відстань між станціями така, ніби дані, що проходять по лінії 4/9 швидкості світла. Для решти 20% вони використовували коефіцієнт швидкості світла 2/9. Таким чином, замість комплексного розрахунку коефіцієнта затримки- відстані, метод SOI визначає фіксований коефіцієнт [20].

Таким чином, на відміну від методу CBG, SOI не повинен вимірювати затримку між орієнтирами. Обмеження створює обмеження лише на основі перетворення затримки на відстань з коефіцієнтом 4/9 швидкості світла.

Навіть при цьому фіксованому діаметрі він працює з такою ж точністю, як і більш складний метод CBG [24].

Метод GeoWeight заснований на методі CBG. Таким чином, як і CBG, він створює межі обмеження і використовує багатосторонній механізм для визначення області, де ціль найімовірніше буде розташована. Найбільша зміна – це створення меж обмежень. Метод GeoWeight ділить отриману затримку між орієнтиром і ціллю на однаково великі ділянки і призначає їм вагу. Розмір цих вагових коефіцієнтів визначає ймовірність того, що цільова точка відбудеться в даному розділі. Чим він більше значення ваги, тим більше ймовірність того, що ціль знаходиться тут. Вимірюючи з декількох орієнтаційних вузлів, отримують серію кіл, що перекриваються. Метод GeoWeight додає вагу окремих ділянок, і площа з найбільшою вагою визначається як найбільш вірогідна зона появи цілі [25].

Принцип методу GeoWeight з трьома орієнтирами (L1, L2, L3) та ціллю показаний на рис. 1.9.

Таблиця 1.4.

Приклад вагової таблиці методу GeoWeight

Оріснтир	Розмір затримки, мс	0-250 км	250–500 км	500-750 км	750-1000 км
L1	100	0	0	0,2	0,8
L2	35	0,2	0,2	0,2	0
L3	15	0,7	0,6	0,1	0

У таблиці 1.4 наведені величини вимірюваних затримок від окремих вузлів до цілі, а також сегменти відстані. Для кожного розділу розраховується вага і присвоюється поточному вузлу. Його розмір, як правило, виражає, наскільки надійним є розділ для визначення цільової точки. Наприклад, під час вимірювання з вузла L1 можна побачити, що ціль, швидше за все, знаходиться в останньому відрізку (750-1000 км). Остаточне визначення області цільового положення остаточно визначається сумою ваг окремих секцій. На рис. 1.9 це розділи L1 -0.8; L2- 0,7; L3 - 0,6. Сума ваг найвища - 2,1, а площа (багатокутник) позначена

червоними крапками.



Рис. 1.9. Загальний принцип методу GeoWeight

1.4 Обмеження активних методів геопозиціонування

Хоча поле геопозиціонування постійно розвивається, воно все ще стикається з низкою перешкод, які суттєво обмежують його точність. Наступний текст описує ці проблеми.

• Змінна затримка в мережах

Глобальна мережа складається з різних технологій передачі та резервних з'єднань, і тому однакова швидкість передачі і, отже, затримка не може бути гарантована. Розділ, де величина затримки найбільше постраждала, називається "остання миля".

Остання миля – це термін, що використовується для позначення підключення користувача (станції) до мережі передачі. Ці лінії з'єднують "домашніх користувачів" зі своїм Інтернет-провайдером (ISP). Зазвичай це підключення через Wi-Fi, ADSL (асиметрична цифрова абонентська лінія) або

кабельне підключення. Зазвичай це лінії, ємність яких спільно використовують кілька користувачів одночасно. Деякі методи геопозиціонування вирішують цю проблему, вимірюючи затримку лише після останнього маршрутизатора в мережі передачі [17].

• Непряма маршрутизація

Під час своєї "подорожі" пакет проходить через серію маршрутизаторів, які визначають відповідно до заданих параметрів, на який інтерфейс маршрутизатора вони будуть відправляти пакет. Це насправді вплине на майбутній шлях пакету. Маршрутизатори приймають рішення на основі так званих метрик лінії, навантажень ліній або інших параметрів, які можна встановити вручну або автоматично. Таким чином, при вимірюванні затримки (пінгу), пакет не повинен їхати таким же шляхом назад. Це також вплине на отриману затримку.

Чому пакет насправді змінює свій шлях? Відповідей може бути декілька. Одне з них полягає в тому, що лінія між маршрутизаторами, названими, наприклад, A і B, руйнується. Маршрутизатор A інформує або повідомляє про розрив з'єднання на лінії (у випадку використання протоколу OSPF [Open Shortest Path First] з повідомленням LSA [Link-state advertisement]) і починає надсилати пакети, призначені маршрутизатору B, на інтерфейсі резервного копіювання. Іншою можливістю, чому пакет маршрутизується по-іншому, є, наприклад, перевантаженість лінії, де лінія перевантажена великим трафіком даних з вищим пріоритетом, ніж Інтернет-протокол керуючих повідомлень (ICMP), що використовується методами геопозиціонування. Тому в більшості випадків пріоритет раунду ICMP встановлюється на нижчий рівень, щоб він не впливав на інші потоки даних [16].

Непряма маршрутизація у вдосконалених методах геопозиціонування обробляється так званим калібруванням орієнтирів - (метод CBG, МГТО тощо).

• Сховати станцію - NAT

З розвитком мережі Інтернет в якості глобальних мереж, перелік IP - адрес скорочується. Наближаючись до часу їх вичерпання, слід припинити послугу "Переклад мережевої адреси", більш відому під абревіатурою NAT (Network

Address Translation). Зменшення кількості IP - адрес, хоча послуга допомагає, але в той же час вона була зменшена і точність геопозиціонування. Наслідком NAT (точніше PAT [переклад адреси порту]) є "приховування" декількох станцій під однією загальнодоступною IP-адресою. Визначення місцезнаходження «прихований» станції є тому неможливо в це час. Будь-яка спроба його локалізації закінчується для визначення положення прикордонного маршрутизатора (який в основному належить ISP). Єдиний факт, що можна використовувати в цьому випадку, і в той же час вважати, є те, що в більшості випадків ці «приховані» станції розташовані поблизу напрямку кордону. Надалі ця проблема буде усунена за допомогою IPv6-адрес, де кожна станція матиме власну загальнодоступну IP-адресу (або більше). Однак виникає питання щодо розподілу адрес IPv6 та подальшого їх внесення до бази даних. Адреси будуть тому, що величезна кількість і підтримка цієї бази даних може бути неможливою. Практично неможливо використовувати зворотний DNS для IPv6, що в основному виникає внаслідок великої кількості адрес IPv6 на одній станції, де ці адреси є додатковими до змінної часу [18].

• IP - VPN (віртуальна приватна мережа), проксі-сервер тощо.

На даний момент, користувач буде підключений до мережі через VPN або з допомогою будь-якого приватного чи публічного проксі - серверу буде його точне місцезнаходження неможливо. Дійсно, користувач буде працювати під його IP - адресою, але буде мати IP адресу мережу, в якій підключення встановлено через згадані VPN. Однак це може бути далеко від його позиції у світі.

• Заборона станції призначення "відповідати" на запити ICMP

Це ситуація, коли адміністратор станції (сервера) забороняє з міркувань безпеки відповіді даних на запити ICMP. Виявити затримку станції буде неможливо, і спроба її виявити не вдасться.

2 МЕТОД ГЕОІНФОРМАЦІЙНОГО ПОЗИЦІОНУВАННЯ ТЕЛЕКОМУНІКАЦІЙНОГО ОБЛАДНАННЯ

2.1 Основи методу МГТО

Метод МГТО – це складний набір принципів і процедур, що дозволяють знайти обмежену область найбільш ймовірного положення цільової станції. У наступних рядках усі ці процедури та вдосконалення, які сприяють якнайбільшій точності цього методу, будуть поступово пояснені та представлені.

Раніше описані методи використовували для своєї роботи лише так звану позитивну інформацію, тобто інформацію, що обмежує місце, де найімовірніше лежить ціль. Зазвичай це коло з центром на орієнтирі та радіусом, яке отримують шляхом перетворення виміряної затримки від орієнтира до цілі на реальну географічну відстань. Радіус кола визначає межу, за якою ціль уже не існує. Крім позитивної інформації, МГТО також використовує так звану негативну інформацію. Негативна інформація вказує на область, як правило, коло, в якій ціль не лежить [22].

Основна перевага методу МГТО є його модифікації. На додаток до затримок, для генерації позитивної чи негативної інформації можна використовувати ряд доповнюючих механізмів. Наприклад, база даних Whois вказує область, де могла б знаходитися цільова точка, або використовує географічні та демографічні дані (океан, незаселені райони), щоб виключити місце.

МГТО використовує комбінацію цієї інформації для створення обмежень. Як видно з рис. 2.1 межі обмеження (кола з радіусами RP, RN) обмежують зони попадання цілі – як правило, ціль не може бути далі радіуса RP і ближче радіуса RN [22].

Зазвичай межі обмеження отримують шляхом перетворення затримки на реальну відстань від набору орієнтирів, положення яких принаймні частково відомі, до цілі. За точністю розташування орієнтирів їх можна розділити:

• Первинні орієнтири (первинні орієнтири), що позначає ці точки, відоме з високою точністю, і тому його вплив на остаточний розрахунок обмеження кордонів буде мінімальним. Положення визначається, наприклад, системою глобального позиціонування (GPS) або шляхом перетворення адреси в географічні координати. Обидві граничні межі для первинного орієнтиру показані на рис. 2.1.

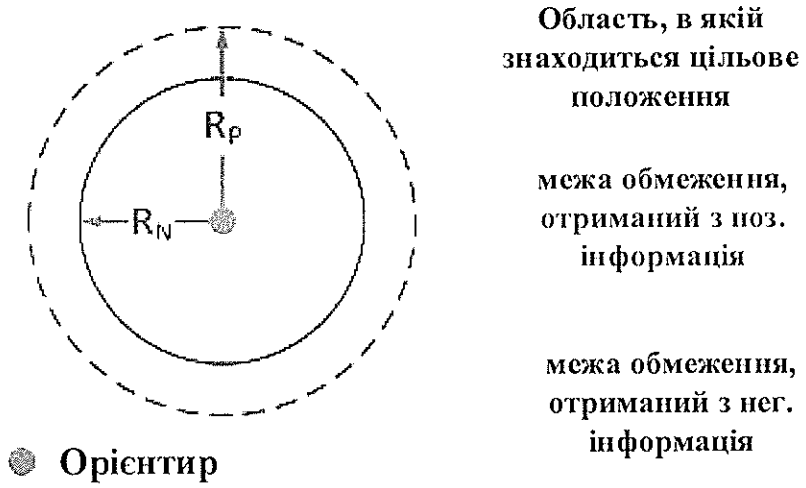


Рис. 2.1. Обмеження відстані для основного орієнтиру

• Вторинні орієнтири (вторинні орієнтири), положення цих точок буде відомо тільки приблизно і виражається в обмеженій області. Їх точне положення повинно бути визначено будь-яким методом геопозиціонування, де становище вторинного орієнтира є фактично об'єктом, що шукається [22].

МГТО, на відміну від інших методів, може працювати як з основними точками орієнтації, так і з вторинними. У разі використання першого, це затримка.

Ситуація складніша для вторинних орієнтирів. Приклади вторинних орієнтирів можуть бути маршрутизатором, який буде розташований на шляху до цільової пам'яті каналу. Розташування кожного з них визначається МГТО, і тому кожен з цих маршрутизаторів може створювати свої обмеження на кордоні. Збільшується із збільшенням кількості орієнтирів, а отже, і загальної точності оцінки площі цільових явищ [22].

2.2 Перетворення затримки на відстань

Попередній текст лише загалом пояснює, для чого використовується межа обмеження та як вона створюється. Однак, оскільки це одна з ключових особливостей методу МГТО, вона буде докладно описана в цьому розділі.

Для створення граничних меж необхідно перетворити значення виміряної затримки на реальну відстань. Найпростішим рішенням було б припустити, що швидкість поширення сигналу дорівнює швидкості світла і не враховувати будь-яку додаткову затримку (наприклад, затримку обробки або збільшену затримку, спричинену навантаженням лінії). Цей метод, хоча і найшвидший з точки зору обчислень, виявився неточним на практиці (згідно з результатами, представленими в роботах авторів методу МГТО) [22].

Хоча метод МГТО міг би працювати таким чином, частіше вибирають процедуру, при якій додана затримка включається до розрахунку. Тоді отримане цільове положення буде набагато точнішим – точніше, отримана площа цільового розташування буде меншою. Для наочності описані окремі етапи отримання відстані від затримки [22].

Процедура створення меж обмежень за допомогою методу МГТО полягає в наступному:

• Кожен орієнтир надсилає повідомлення ping іншим орієнтирам після його запуску і разом із відповіддю кожного з них отримує суму затримки на нього. Отримані значення вони зберігають у кореляційній таблиці.

Потім регулярно оновлює збережені значення, повторно надсилаючи повідомлення ICMP. За допомогою цих оновлень метод МГТО здатний гнучко реагувати на затримки змін у надзвичайних ситуаціях мережі (наприклад, активація шляху резервного копіювання тощо).

• Для кожного значення затримки в таблиці кожен орієнтир призначає відстань до орієнтира, з якого він отримав цю затримку. Це можливо, оскільки місцезнаходження визначних пам'яток відомо. Створюється графік, аналогічний графіку на рис. 2.2 (поки без складної конверти).

Іншою метою є отримання відношення затримки-відстані, за допомогою якого орієнтир зможе перерахувати отримані значення затримки до обшукуваної цілі на реальну географічну відстань і, таким чином, фактично отримати межі обмеження.

Цей коефіцієнт виходить шляхом створення так званого складного конверта, який оточує всі точки (показано чорним на рис. 2.2). Наприклад, коли орієнтир вимірює затримку до цілі 50 мс, він може визначити, що ціль, яку шукають, знаходиться не ближче 2500 км (внизу конверта) та 4000 км (вгорі конверта) від нього. Перше значення позначає внутрішній радіус (RN), а друге зовнішнє коло (RP). Якщо з'єднати два кола між собою, створюється проміжне коло, яке створює так звану граничну межу (рис. 2.1).

Усі точки (рис. 2.2) обмежені ліворуч пунктирною лінією, що представляє межу, утворену лише затримкою, спричиненою поширенням сигналу в волокні (приблизно | швидкість світла). З попереднього тексту вже відомо, що цей рядок також може бути використаний для отримання позитивних граничних обмежень (радіус RP). Однак він не використовується через низьку точність [22].

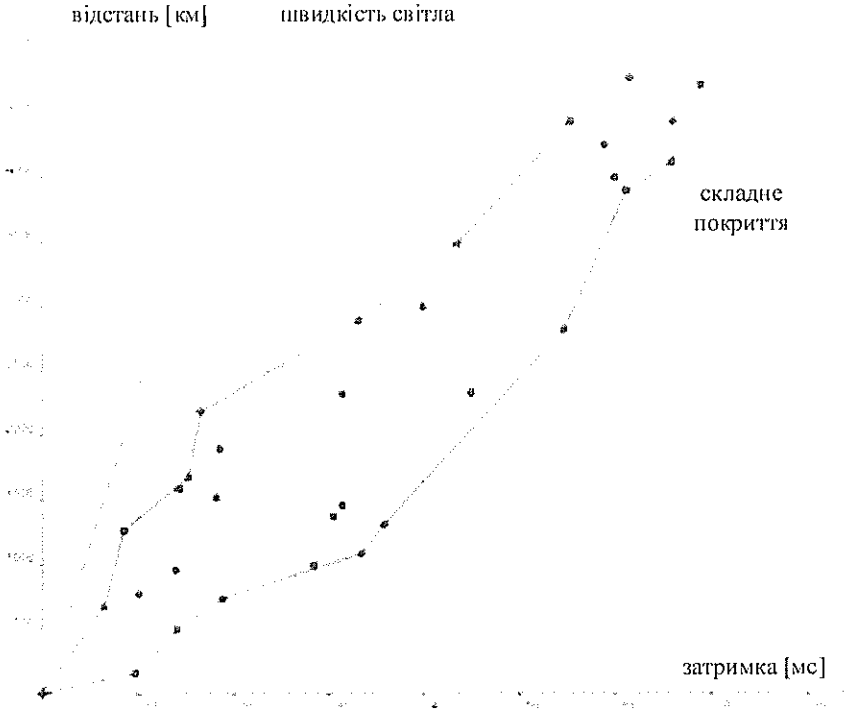


Рис. 2.2. Графік затримки від відстані

Алгоритм розрахунку складної оболонки не згадується більш точно в «Комплексній структурі геолокалізації Інтернет-хостів» [22]. Тому будуть використовуватися так звані прямолінійні настанови.

Перш за все, варто нагадати, які є вже згадані директиви, або директивне рівняння рядка, яке буде використано при розрахунку складної огинаючої.

Рівняння прямої

Сукупність усіх точок X [x, y], координати яких задовольняють рівняння

$$y = kx + q \qquad k, q \in R \tag{2.1}$$

- це лінія, що розходиться з віссю y. Це рівняння називається прямолінійним рівнянням. Коефіцієнт k називається напрямком прямої і визначається

$$k = tg\varphi = \frac{\Delta y}{\Delta x}. \tag{2.2}$$

Де кут, який пряма робить з позитивним напрямком осі x. Коефіцієнт q вказує переріз, який пряма перетинає на осі y [26].

Директива виражає величину відхилення від осі x.

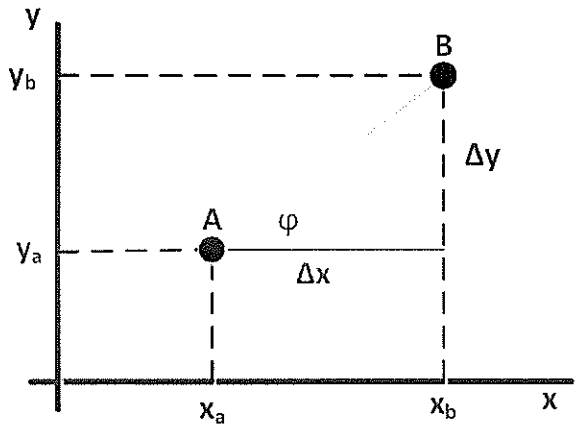


Рис. 2.3. Графічне зображення напрямку лінії

Отже, процедура розрахунку складної оболонки полягає в наступному (спочатку для нижньої межі):

• Є ряд точок з координатами: A (2,4), B (6,4), C (8,18) і D (12,10).

• Спочатку всі точки сортуються відповідно до їх координат на осі x. Точка

A, отже, має найменше значення координати x. Інші точки позначаються у порядку зростання за алфавітом.

• У наступній точці нульову точку зв'язують з точкою A прямою лінією. Тобто, обчислюється рівняння прямої. (Рис. 2.4 зліва).

Таблиця 2.1.

Приклад розрахунку для прямої, визначеної точками 0, A

Точки, що визначають пряму	Рівняння рівнянь	Приклад розрахунку
Точка A (2.4) Точка O (0,0)	—2% + y = 0	Точка C: —2 * 8 + 18> 0 Точка D: —2 * 12 + 10 <0 Точка B: -2 * 6 + 4 <0

• Алгоритм визначає, чи всі інші точки знаходяться вище цієї лінії – він порівнює, чи будуть значення лівої частини рівняння лінії після підстановки координат точки більшими, ніж права частина рівняння лінії. Для даного прикладу ця умова не поширюється на пункти B і D.

• Умова не виконується, тому "кінцева точка" рядка відкидається (точка A) і замінюється іншою точкою в послідовності (у нашому випадку точка B). (Рис. 2.4 праворуч).

Таблиця 2.2.

Приклад розрахунку для лінії, визначеної точками 0, B

Точки, що визначають пряму	Рівняння рівнянь	Приклад розрахунку
Точка B (6.4) Точка O (0,0)	—2% + 3y = 0	Елемент C: -2 * 8 + 3 * 18> 0 Пункт D: -2 * 12 + 3 * 10> 0

• На наступному кроці виконується умова всіх точок над лінією (Рис. 2.4 праворуч) і таким чином рівняння лінії зберігається.

• Наступний крок потім "переміщує" лінію до наступних точок послідовності. Однак кінцева точка (точка B) першого рядка повинна бути початковою точкою другого рядка. У даному випадку програма також розраховуватиме на лінію, визначену двома точками - B і C.

• Під час роботи програма зберігає в пам'яті кінцеву точку рядка і завжди порівнює її з останньою точкою серії (у даному випадку точкою D) після збереження рядка. Якщо вони рівні, то більше немає сенсу і програма закінчується.

• Для розрахунку верхньої точки ситуація схожа на ситуацію, за винятком зміни функції управління. Він перевірить розташування всіх точок нижче поточної лінії.

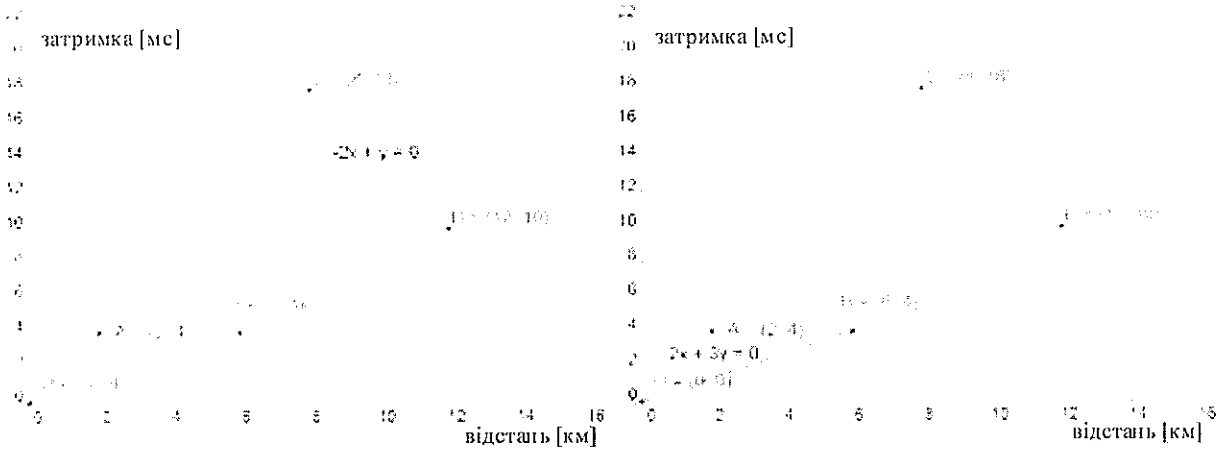


Рис. 2.4. Процедура розрахунку нижнього променя (пакету)

За допомогою цієї процедури, яку виконує кожен вузол орієнтації, отримується співвідношення між затримкою та реальною відстанню – тобто, набір рядків, що вказують на це співвідношення. Таким чином, кожен орієнтир може створити свої межі обмежень [20].

2.3 Місце призначення

Як тільки кожен з орієнтирів знає свої межі обмеження, можна переходити до оцінки положення самої цілі. Точніше – поки що лише райони найбільш вірогідної появи цілі. Розмір цієї площі залежить від розмірів граничних меж (тобто ширини кільцевої кільця). Вузли, які вимірювали найбільшу затримку до цілі, мають найбільші обмеження. Це означає, що вони віддалені від цільового вузла. Перетинаючи уявні проміжні кільця з найдальшого до найближчого, область, яку шукають, починає звужуватися і зменшуватися в розмірах (див. рис. 2.5).

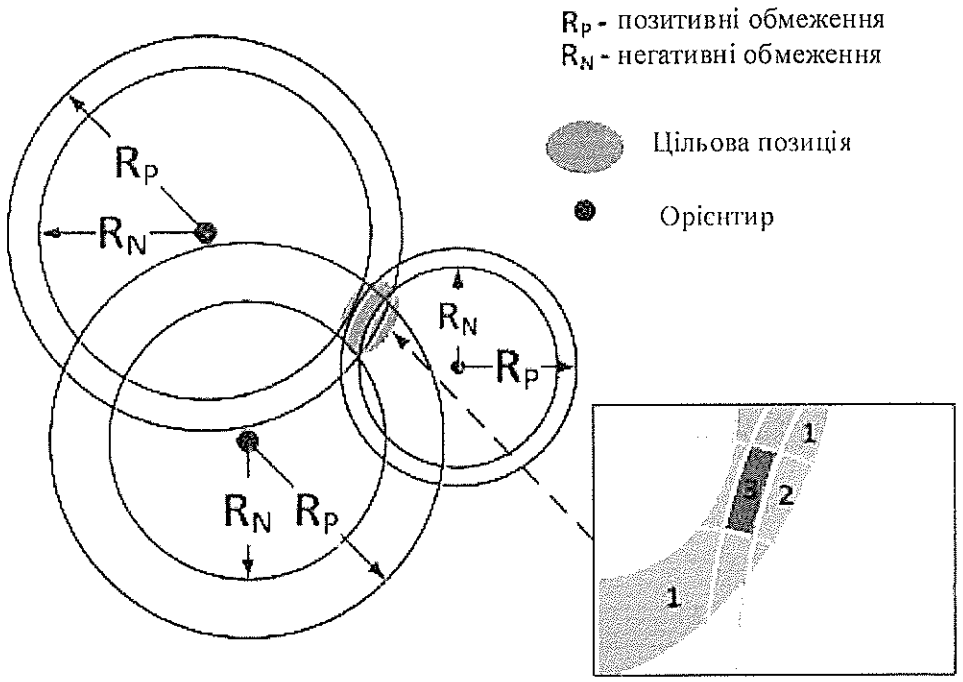


Рис. 2.5. Місце призначення

Однак для найвіддаленіших орієнтирів існує велика ймовірність того, що на їх затримку в досягненні пункту будуть впливати події та затримки окремих активних елементів у мережі. Іншими словами, пакет, швидше за все, буде прокладений за іншими шляхами, довшими чи коротшими, через перевантаженість лінії чи інші показники і т. д. Оцінка місця розташування за цими орієнтирами, отже, менш надійна, ніж від ближчих. Для того, щоб зменшити вплив найдальших орієнтирів та підвищити постійну точність, метод МГТО

вводить систему так званих мір зважування. Це означає, що кожна нововизначена область отримує свій коефіцієнт значення – вагу. Можливі призначені вагові коефіцієнти показані на рисунку 2.5 праворуч унизу. Ваговий коефіцієнт з найменшим значенням матимуть площі, отримані при перетині меж границь найдальших орієнтирів. Зі зменшенням відстані значення вагових коефіцієнтів буде зростати.

Використання вагових коефіцієнтів також забезпечить, що пошук цільової області не закінчиться жодним рішенням – тобто рішення не буде порожнім набором. Якщо області не змінюють ваговий коефіцієнт після додавання іншого орієнтира, цілком ймовірно, що орієнтир містив помилкові або неточні значення затримки і не буде включений до наступного рішення [22].

Локалізація однієї точки

Цільова оцінка обшукуваної області методом МГТО може бути не остаточною. Деякі програми або послуги вимагають прямого доступу до знайденої позиції. Це, наприклад, генерація веб-сайтів на основі поштового індексу користувача, який отримує доступ до обраної сторінки. У випадку більшої площі програма, розміщена на веб-сайті, не зможе дізнатися цей номер у користувача.

Для обчислення положення однієї точки метод МГТО може використовувати математичний метод обчислення, який базується на використанні випадкових величин та теорії ймовірностей, що називається Монте-Карло, або виразити обмежену площу як багатокутник та обчислити центр ваги. Принцип методу Монте-Карло такий: наприклад, в оцінювану область вставляється тисяча точок, і кожному з них присвоюється вага на основі суми відстаней від цього до інших. Після декількох спроб (із збільшенням кількості спроб "точність" також збільшується) точка з найменшим значенням ваги позначається як шукана мішень [27].

Іншими факторами, що впливають на вимірювання затримки в мережі та погіршення точності, є так звана затримка останньої милі та непряма маршрутизація.

2.4 Непряма маршрутизація

З попереднього тексту випливає, що розмір радіуса кола (межі обмеження) пропорційний довжині маршруту між орієнтиром і кінцевою точкою. На практиці це було доведено у багатьох випадках. Наприклад, маршрутизатор A, зареєстрований у Києві, може опосередковано відправити пакет, призначений для маршрутизатора в Обухів, через маршрутизатор в Борисполі. Замість відстані 185 км (Київ-Обухів) пакет перейде 413 км (Київ-Бориспіль), і розмір затримки різко збільшиться. Цей, можливо, дещо перебільшений приклад показує, що затримка від цього орієнтиру вплине на розмір отриманої області пошуку. Метод МГТО здатний вирішити проблему так званої непрямої маршрутизації та компенсувати її вплив на розмір області, яку шукають [22].

Це робиться за допомогою проміжних маршрутизаторів як вторинних вузлів у кілька етапів.

• Спочатку метод МГТО визначає, які вузли знаходяться на шляху між орієнтиром та пунктом призначення. До прикладу, можна використовувати додаток схоже на трасування.

• Крім того, метод МГТО використовує зворотні записи DNS для виявлення їхніх доменних імен і використовує внутрішні механізми, щоб дізнатись їх місцезнаходження. Цей момент є найскладнішим і буде детально описаний нижче.

• Нарешті, метод МГТО включає ці позиції у свій набір орієнтирів та визначає їх межі. Наступна процедура така ж, як і для первинних орієнтирів. Принцип показаний на рис. 2.6. Межа обмеження (проміжне коло) для ясності опущена для контрольної точки.

Найскладнішим завданням є з'ясування точного положення маршрутизатора. Грубе визначення положення здійснюється методом МГТО шляхом локалізації по частинах, коли положення маршрутизаторів визначається поступово від точки орієнтації до пункту призначення. Маршрутизатори, у яких їхнє положення хоча б частково відомо, використовуються як вторинні орієнтири на наступному кроці.

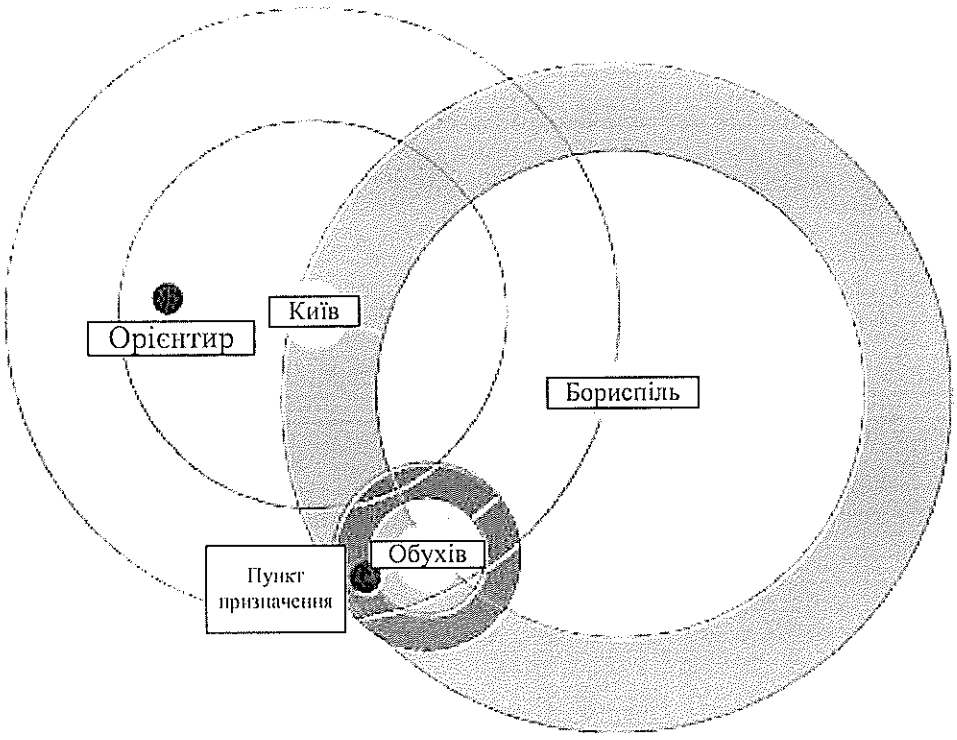


Рис. 2.6. Непряма маршрутизація

Більш точне розташування маршрутизатора (на рівні міста), якщо це можливо, базується на його доменному імені а. Багато постачальників послуг Інтернету дотримуються стандартних зразків при іменуванні маршрутизаторів. Наприклад, маршрутизатор sl-bb11-nyc-3-0.sprintlink.net є магістраллю (bb11) і знаходиться в Нью-Йорку (nyc), маршрутизатор te-0-6-0-0.tr1.pop1.kyv.sloane.com знаходиться в Києві (kyiv), і його доменне ім'я також містить більш точне місце розташування, назву пристрою та інтерфейс, на якому використовується IP-адреса. Отримати попередню інформацію можливо завдяки інструменту undns. Цей сценарій був створений і використовується у проекті під назвою Rocketfuel. Проект зосереджений на новій техніці для відображення топології мережі на рівні маршрутизатора з використанням якомога менше зайвих вимірювань.

2.5 Затримка "остання миля"

Остання миля або останній стрибок – це зв'язок між абонентом та прикордонним маршрутизатором, яким керує його провайдер. Більшість з них – це з'єднання, реалізовані за допомогою більш повільної технології (порівняно з технологіями, що з'єднують активні елементи мереж передачі). Прикладами є технологія 802.11 g, n (Wi-Fi) або xDSL (Digital Subscriber Line). Ці зв'язки зазвичай є джерелом великих затримок, величина яких може впливати на загальну величину результуючої затримки між орієнтиром та ціллю [22]. Деякі методи геопозиціонування вирішують цю проблему так, що затримка вимірюється лише на останньому маршрутизаторі на шляху до пункту призначення. Це насправді означає пропуск останнього стрибка та усунення джерела надмірної затримки [23].

Метод МГТО підходить до цієї проблеми дещо витонченіше. По-перше, він вимірює затримку між парами первинних орієнтирів. Коли первинні орієнтири позначені a, b, c, затримка між ними буде [a, b], [a, c], [b, a]. Оскільки їх положення відоме, їх граничні межі можуть бути розраховані (у геометричному рішенні кола), а теоретична затримка також може бути визначена, використовуючи відношення відстані до затримки. Ця оцінка теоретичної затримки буде позначатися як (a, b), (a, c), (b, a). Тому затримку на останній милі можна визначити, використовуючи просте відношення – між вузлами a, b це відношення [a, b] - (a, b) і воно називається висотою (висотою відповідно до роботи [22]).

Метод МГТО також визначає, наскільки велике значення затримки - a ‘, b’, c ‘буде приписано кожному орієнтиру відповідно до співвідношення:

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} * \begin{bmatrix} a' \\ b' \\ c' \end{bmatrix} = \begin{bmatrix} [a,b] - (a,b) \\ [a,c] - (a,c) \\ [b,c] - (b,c) \end{bmatrix} \quad (2.3)$$

Подібна процедура застосовується до цільового вузла. Якщо значення затримки (тобто висота) відоме як для цільової точки, так і для орієнтирів, окремі

орієнтири можуть змінити свої обмеження (проміжні кільця). Якщо висота цілі менше висоти інших орієнтирів, орієнтир може збільшити радіус зовнішнього кола. Якщо все навпаки і висота цілі менша за висоту інших орієнтирів, то радіус внутрішнього кола зменшується. Таким чином, запобігається більшим спотворенням отриманих обмежень через збільшення значення затримки через так звану останню милю [22].

2.6 Експериментальна мережа - PlanetLab

PlanetLab – це глобальна мережа, що працює з 2002 року низкою світових університетів та великих компаній (таких як Google або Hewlett-Packard), що займаються інформацією та телекомунікаціями. Його головне завдання – забезпечити середовище для глобального тестування мережевих додатків та протоколів. Одним із прикладів є проект, який спрямований на створення ефективної системи трансляції IP-телебачення (послуга Video on Demand) у всьому світі. Без експериментальної мережі PlanetLab її тестування було б обмежене локальною мережею. У кожен проект входить кілька вузлів – цей набір вузлів називається зрізом.

Структура PlanetLab поділяється наступним чином. На найнижчому рівні знаходяться окремі комп'ютери (вузли), на яких запущений дистрибутив CentOS Linux, що забезпечує незалежну віртуальну машину для кожного користувача. Це рішення дозволяє кільком користувачам одночасно використовувати ресурси одного вузла.

Користувачі можуть підключатися до своїх облікових записів або через командний рядок в системі Linux, або використовувати PuTTY якщо вони підключаються з операційної системи Windows. З'єднання обробляється за допомогою протоколу SSH. Він забезпечує зашифроване спілкування з віддаленим вузлом, а також аутентифікацію за допомогою відкритого ключа (асиметричне шифрування) [29].

Вузли із загальними властивостями (наприклад, належать одній організації)

належать до так званих сайтів. На даний момент PlanetLab має (вересень 2023 р.) 1116 вузлів на 544 сайтах.



Рис. 2.7. Розгортання вузлів у мережі PlanetLab

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ МЕТОДУ МГТО ТА НАБОРІВ КОНТРОЛЬНИХ ТОЧОК

3.1 Орієнтири та цілі інфраструктури

Практичну частину цієї роботи слід розділити на кілька частин. Перший стосуватиметься орієнтирів та цілей, другий – описом створеної програми МГТО мовою програмування Java. Останній підрозділ буде присвячений загальній оцінці точності методу МГТО та його порівняння з іншими методами.

База балів

Спочатку потрібно було знайти орієнтири, які відповідатимуть зазначеним умовам. Це означає, що він повинен відповідати на запити ICMP (пінг) і забезпечувати безперебійну роботу записаної програми методу геопозиціонування, яка матиме завдання вимірювати і контролювати коливання затримок мережі між цими вузлами (так зване автоматичне калібрування). Для створення бази даних було використано мережу PlanetLab. Спочатку перевірено, що вузол відповідає на запити ICMP, а потім підтвердив своє місцезнаходження. Вузли з перевіреним положенням і відповідають усім умовам показані на рисунку 3.1.

Методи CBG, МГТО та GeoWeight будуть використовувати ці пункти, щоб програма окремих методів працювала на них, а межі обмежень розраховувались з них. Іншими словами, вони будуть уявними центрами кіл, радіус яких – відстань до цілі, визначена від затримки.

Цільові точки

Інша група точок, яка буде виконувати функції цільових точок (тобто буде здійснено їх пошук), буде отримана з використанням записів DNS типу LOC. Спочатку використання цих записів здавалося ідеальним. Записи показують точне розташування сервера, до якого належить запитувана IP-адреса. На жаль, як виявилося пізніше, кількість отриманих записів не відповідала величині прикладених зусиль, виникло кілька ідей щодо того, як знайти ці моменти. Перша

ідея знайти вільний доступ до бази даних, яка б збирала сервери, що підтримують записи LOC DNS, не була реалізована. Друга ідея, яка вже була успішною, полягала в тому, щоб отримати список доменів та запитувати їх окремо для записів DNS. Тепер потрібно було лише отримати вільно доступну базу даних великої кількості серверів разом з їхніми IP-адресами. 4,5-мільйонний список адрес було завантажено з веб-сайту DMOZ. Ці веб-сайти, які є частиною проекту Open Directory (ODP), мають на меті збирати адреси серверів для створення некомерційного каталогу веб - сайтів [32]. Хоча сценарій LOClookup працював дуже добре, необхідно було створити для нього модифікацію, яка забезпечила б циклічну ротацію запитів до DNS-сервера. Якби цього не сталося, результатом було б непропорційне навантаження на один DNS-сервер, на який було б надіслано 4,5 мільйона запитів для отримання записів DNS LOC.

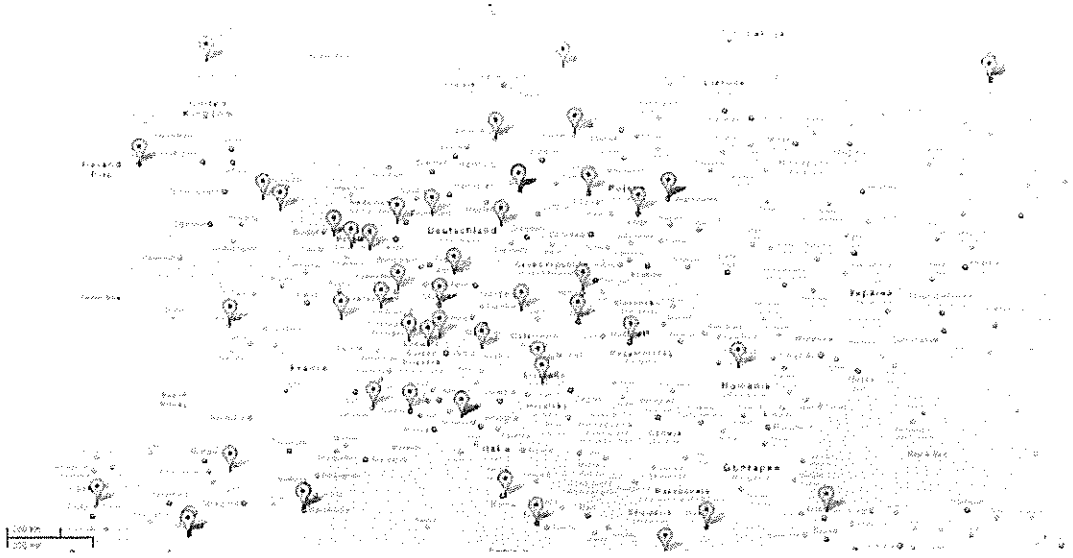


Рис. 3.1. Набір орієнтирів із мережі PlanetLab

DNSrotateLOClookup – це скрипт, створений в об'єктно-орієнтованій мові програмування сценаріїв Python, і його завдання полягає в поступовому чергуванні надсилання запитів на 16 серверів DNS. Обидва сценарії наведені в додатку з докладними коментарями.

Результатом даної роботи стало придбання 181 (приблизно 0,004%) доменів зі списку 4,5 мільйонів адрес, які мають DNS- локальний запис. З цих доменів

лише 143 відповіли на запити ICMP, і лише 90 з них були в Європі.

Результати показують, що серверів із підтримкою DNS LOC дуже мало. Цю умову можна віднести, наприклад, до безпеки або до того, що заповнення запису DNS LOC не є обов'язковим.

3.2 Орієнтири

Для остаточного порівняння всіх вказаних методів наступні пункти потрібно було вибрати та визначити, щоб відповідати кільком критеріям. Основна умова опорних орієнтирів (від мережі PlanetLab) полягала у необхідності постійної доступності пінгу ICMP, а також у високій точності їх відомого розташування. Було створено список із 40 вузлів із вихідних 140. Їх положення показано на рис. 3.2.

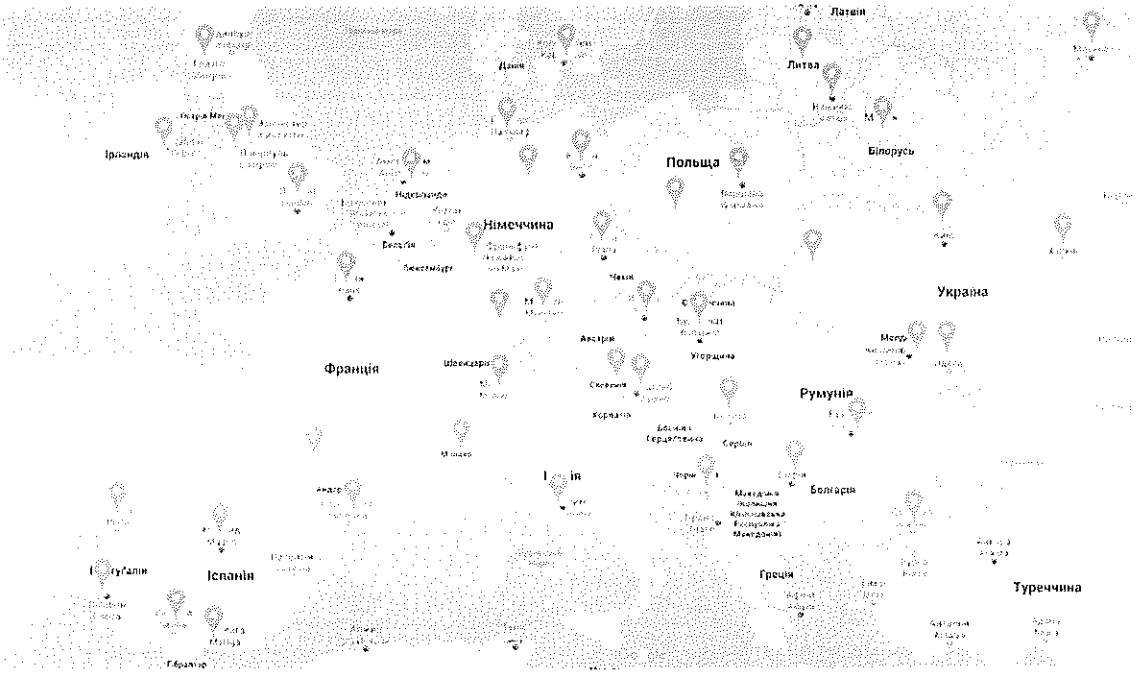


Рис. 3.2. Вузли опорної орієнтації

Орієнтовні цільові точки

Метою було отримати список найбільш доступних вузлів разом з найбільш надійним місцем розташування. Таким чином, оригінальний список із 90 вузлів

призначення (знайдених за допомогою DNS LOC) був зменшений до остаточного числа 76. Автентифікація виконувалася вручну, і список включає вузли, які тривалий час відповідають на запити ICMP, а також їх координати не вказують на незаселені райони чи море.

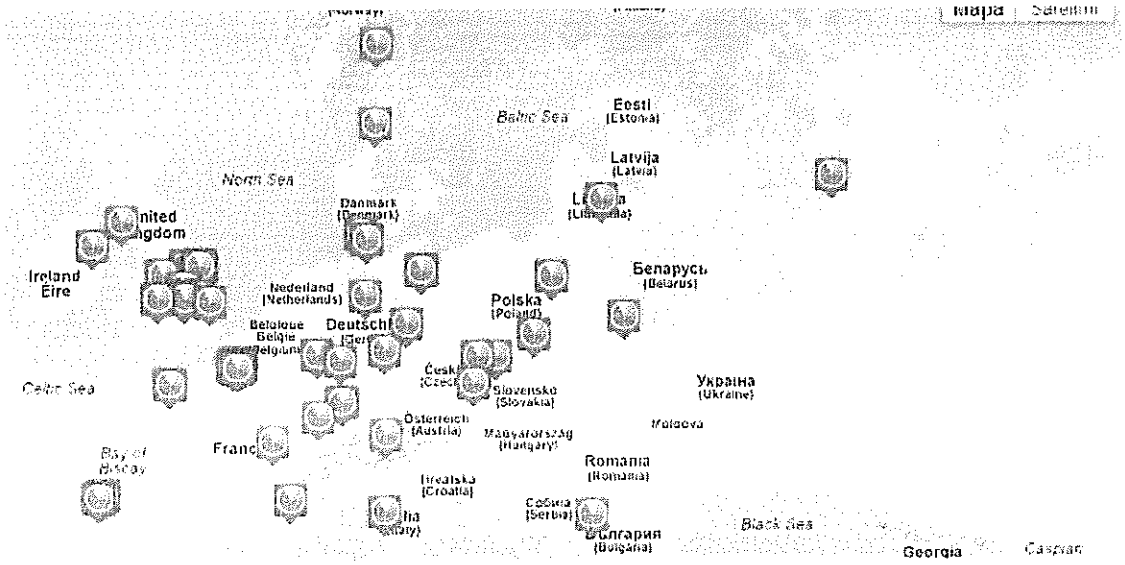


Рис. 3.3. Довідкові вузли призначення

3.3 Програма МГТО - практична реалізація

Загальна реалізація методу МГТО була розділена на кілька підпрограм, які будуть детальніше описані в наступних розділах. Але спочатку буде корисно окреслити, як насправді працює вся програма. Спрощений приклад показано на рис. 3.4.

Після надсилання запиту про розташування пункту призначення запити (IP-адреса призначення) надсилаються на кожен вузол орієнтації в мережі PlanetLab.

На кожному вузлі запущено три частини методу МГТО. Їх завдання – визначити відстань до інших орієнтирів та розрахувати коефіцієнт затримки-відстані (CalculationCovers.jar), який згодом використовується для обчислення межових меж або радіусів кільця (MGTOKlientDoslidPC.jar). Потім межі обмеження повертаються у відповідь MGTOKlientDoslidPC.jar. За їх допомогою та математичних процедур, які будуть пояснені далі, розраховується ймовірне

положення цільової точки. Результат – графічне представлення місця на карті на вже згаданих сторінках.

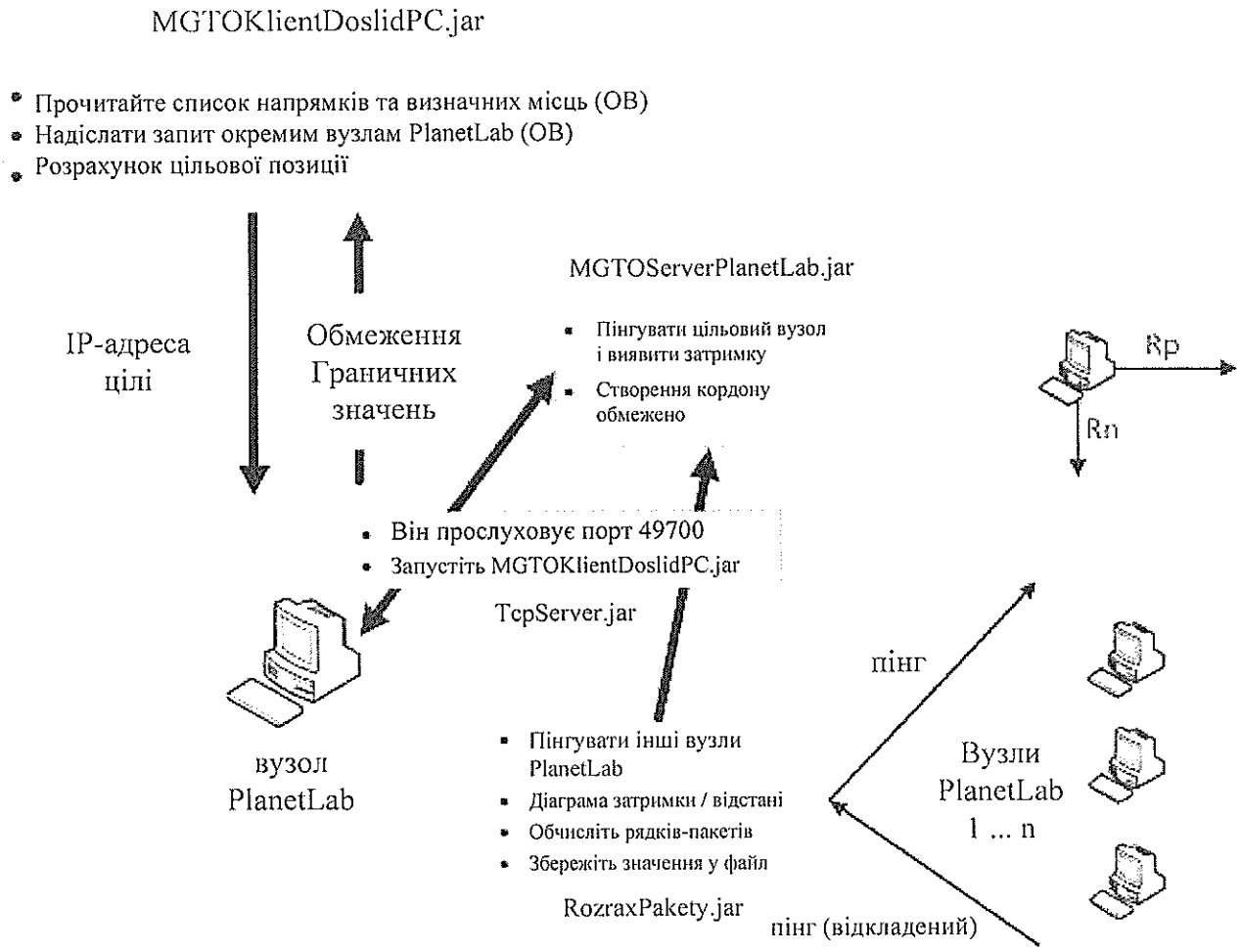


Рис. 3.4. Принцип програми МГТО

Програма, що працює на вузлах мережі PlanetLab

З попереднього рисунка видно, що три частини програми МГТО працюють на орієнтаційних вузлах. Поділ всієї програми на кілька частин мав на меті полегшити налагодження всієї програми, а також її можливі модифікації, такі як зміна типу з'єднання з SSH на TCP або алгоритм для обчислення складного конверта.

• Програма RozraxPakety.jar

Ця частина програми запускається на орієнтир періодично (кожні 50 хвилин) за допомогою програми cron, доступної в усіх операційних системах Unix. Тобто, регулярний перерахунок графіка затримки-відстані, результатом

якого є змінна ширина кільця. За допомогою цього показника метод МГТО може реагувати на зміни в мережі. Якщо затримка в мережі збільшується, ширина меж обмежень від цього вузла також збільшується. Однак на практиці, за винятком кількох випадків, було доведено, що цей захід не є настільки необхідним. На лініях, що з'єднують ОВ, значних змін у затримці не виявлено навіть під час тривалого спостереження.

На рис. 3.5 показано, як програма виявляє та обчислює верхній/нижній пакет за графіком затримки - відстань, яка згодом використовується для обчислення обмежень.

Програма запускається на орієнтаційному вузлі мережі PlanetLab.

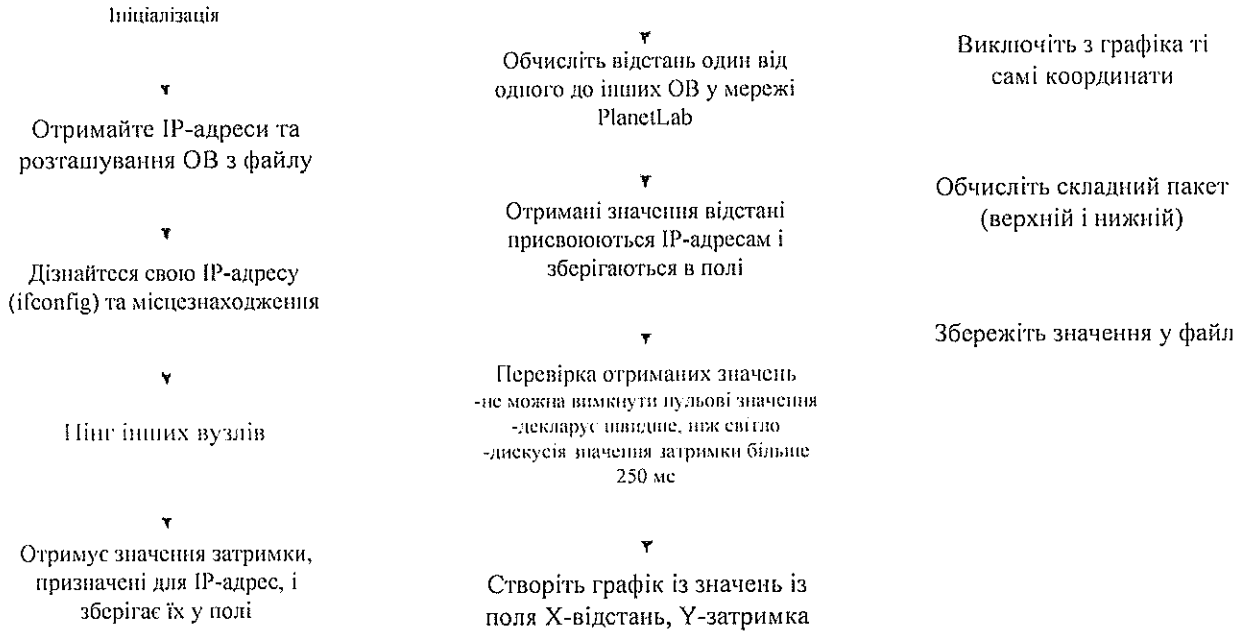


Рис. 3.5. Блок-схема RozraxPakety.jar

Файл, з якого зчитується список IP-адрес інших вузлів орієнтації, знаходиться безпосередньо на вузлах у домашній директорії, а дані в ньому зберігаються у форматі JSON (JavaScript Object Notation). Формат JSON був обраний типовим форматом для вхідних та вихідних даних, оскільки він має фіксовану структуру, яку легко читати як машину, так і людину.

Файлова система, тобто завантаження та збереження з/у файл, була обрана

завдяки легкій зміні прочитаних даних та кращому моніторингу програми, включаючи її аналіз. Змінюючи орієнтири за замовчуванням, просто замініть файл іншим.

Відстань між двома орієнтирами обчислюється за наступною формулою:

$$\text{Відстань} = \arccos(\sin(x_1)*\sin(x_2)+\cos(x_1)*\cos(x_2)*\cos(y_1 y_2))*\text{радіус Землі}, \quad (4.8)$$

де x - широта, y - довгота двох точок, а радіус Землі має значення 6372,8. Цей алгоритм, званий відстанню Великого кола, базується на припущенні, що Земля є сферою заданого радіуса. Таким чином, обчислена відстань є з'єднанням двох точок на сферичній поверхні, яка лежить на колі з центром у центрі ваги (центром сфери) [33].

Результатом роботи програми є файли, в яких координати двох точок, що визначають лінію, а також її напрямок і зміщення зберігаються на кожному рядку. Значення, що вказують верхній конверт, зберігаються у файлі з назвою paketVerxn.txt, а ті, що утворюють нижній конверт, зберігаються у файлі paketNuzn.txt. Для точок з координатами A (600, 200), B (900, 350), C (1200, 400), D (1500, 500), E (1800,800), F (2100,920), G (2400, 1250), H (2700, 1300), I (3000, 1450), файл paketVerxn.txt міститиме значення, наведені в таблиці 3.1, а графічний результат програми показаний на рис. 3.6.

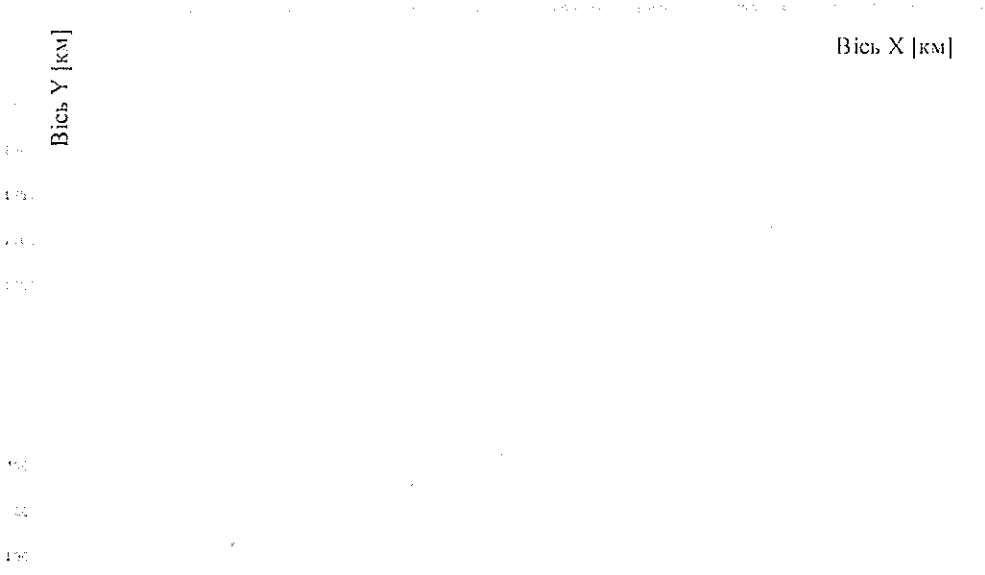


Рис. 3.6. Графік відстані, включаючи пакети

Таблиця 3.1.

Значення у файлі paketVerxn.txt

paketVerxn.txt	Точки, що визначають пряму	Параметр k (напрямок) і q (зміщення)
Перша лінія	A (600, 200), G (2400, 1250)	0,58333, -150
Друга лінія	G (2400,1250), 1 (3000,1450)	0,3333, 450

• Програма TCPServer.jar

Ця програма запускається на всіх фонових вузлах орієнтації за допомогою команди nohup16, яка гарантує запуск програми, а також запобігає її виходу при виході з консолі/терміналу. Його завдання – прослуховувати порт 4970017 та очікувати команди у форматі "відстань + IP-адреса призначення". Після отримання він запускає файл MgtoServerPlanetLab.jar із параметром уже згаданої IP-адреси призначення. У відповідь він потім отримує два значення відстані до цільового вузла - Rp і Rn, що вказує на радіус внутрішнього і зовнішнього кіл кільцевого кільця, які визначають межі обмеження.

Після отримання команди "quit" встановлене TCP-з'єднання розривається.

• Програма MGTOServerPlanetLab.jar

Програма запускається за "командою" з підпрограми TCPServer.jar з параметром IP-адреси цілі (далі – лише ціль).

Спочатку програма використовує запити ICMP (кількість яких встановлено як 10), щоб знайти найменшу затримку до цілі. На наступному кроці значення paketVerxn, paketRozPak значення з файлів paketVerxn.txt, paketNuzn.txt, які були створені програмою RozraxPakety.jar, зберігаються в динамічних полях.

Останнє завдання програми MGTOServerPlanetLab.jar – знайти відповідний рядок, який буде використаний для перетворення затримки на відстань.

Пошук цієї лінії буде заснований на порівнянні координат по осі у частині затримки знайденої до мети і координатами Ypsilon точок, що визначає лінію.

Наприклад, виявлено затримки до цілі 52,45мс. Програма nejpr в визначає, чи лежить значення між координатами точок A і B. У цьому прикладі лежить не

так супроводжується порівнянням з точками B і C. Затримка 52,45 мс, розташованих в цьому інтервалі (від 40 до 80 мс), і тоді відповідним рядком буде, який буде визначатися цими точками (на зображенні нижче буде показаний в зеленому кольорі).

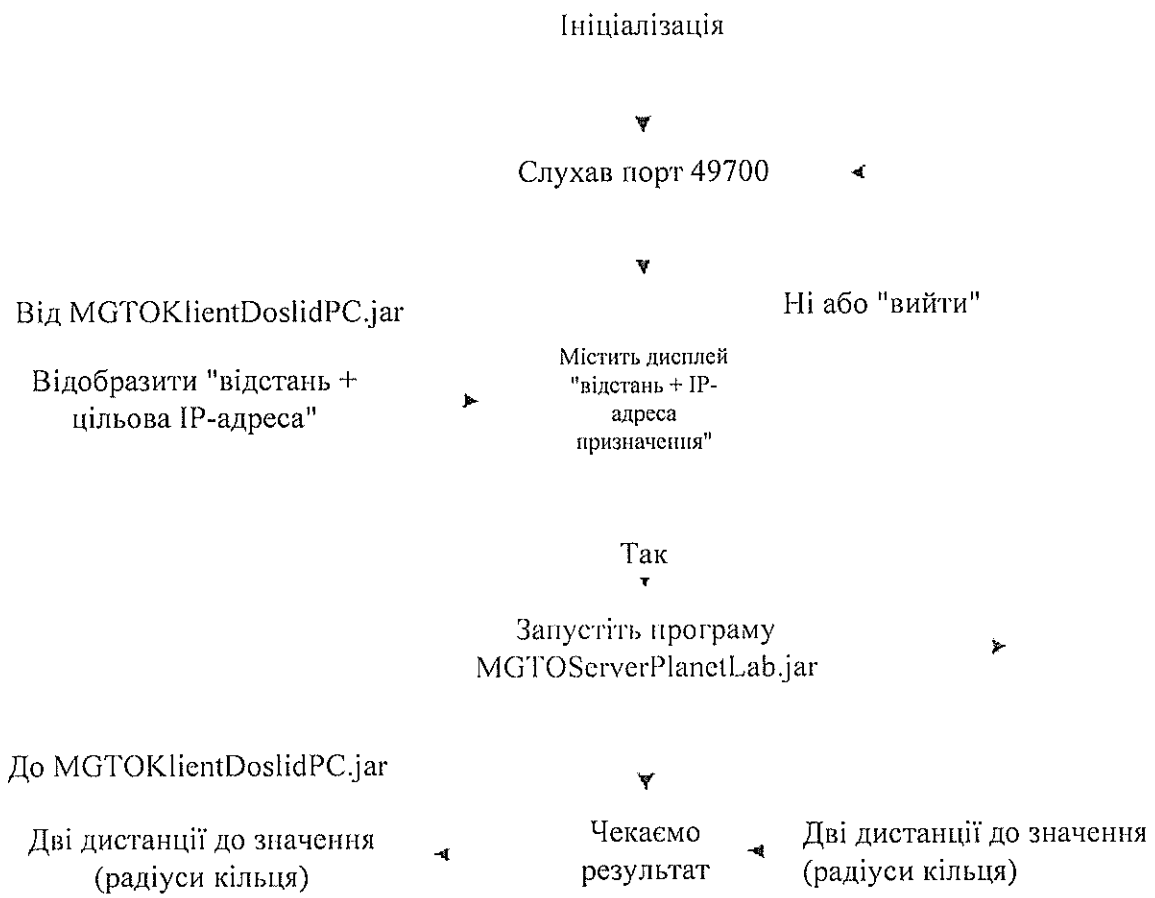


Рис. 3.7. Блок-схема TCPServer.jar

Тепер це готове для вирахування відстані до цілі. Затримка, яка становить 52,45 мс, підставляється в рівняння вибраної лінії y = 10% - 40 на y і обчислюється x, що є необхідною відстанню до цілі. Потім це значення, у даному випадку 9,245 км, передається програмі MGTOKlientDoslidPC.jar. Потім вся процедура повторюється аналогічно для верхньої кришки. Примітка: значення на осі x є лише орієнтовними.

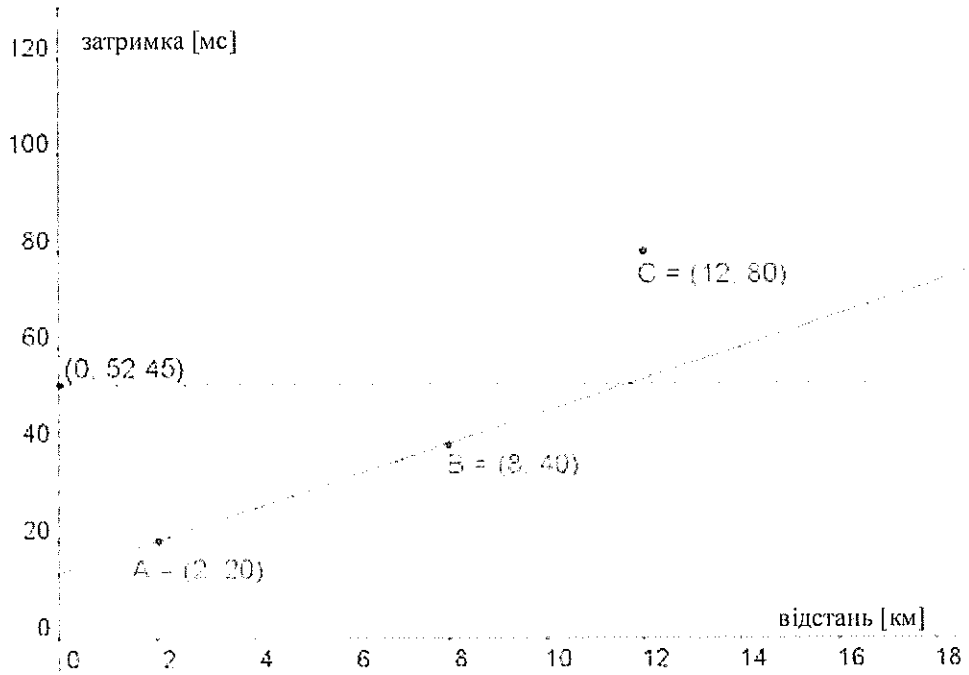


Рис. 3.8. Пошук відповідної лінії

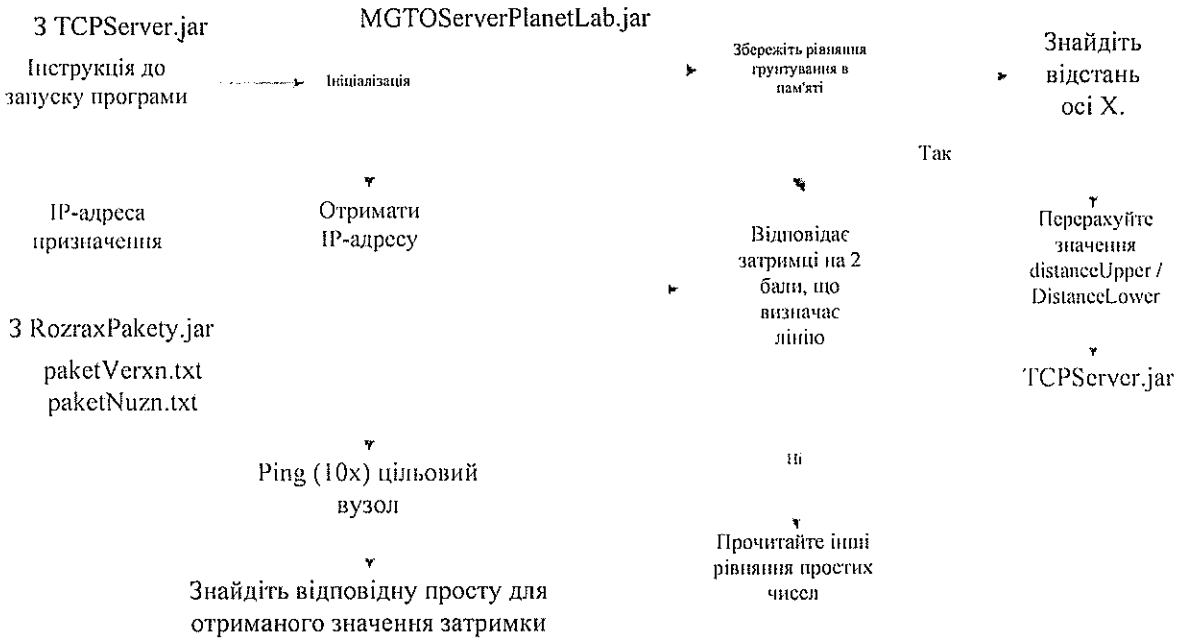


Рис. 3.9. Процедура програми MGTOKlientDoslidPC.jar

Програма, що працює на шкільному сервері

Основна частина програми (MGTOKlientDoslidPC.jar) працює на шкільному сервері.

Процедура програми така:

• Спочатку на вищевказаних сторінках вводиться IP-адреса пункту призначення та вказується перелік орієнтирів, які слід використовувати для вимірювання. Вхідні дані надсилаються програмі у форматі JSON.

• Після надсилання запиту на локалізацію програма запускається і спочатку перевіряє, чи відповідають пошукові пункти запиту ICMP. Якщо так, вони зберігаються в полі активних пунктів призначення та запускається цикл, який виконується один раз для кожного пункту призначення. У цьому циклі програма поступово підключається до всіх вузлів орієнтації в завантаженому списку.

На цих вузлах підпрограма TCPServer.jar прослуховує вказаний порт, який після отримання команди "distance + IP address of the destination" запускає інші програми. Результатом є повернене значення (яке називається delayAB), що складається з двох чисел, розділених пробілом. Перший вказує радіус внутрішнього кола, а другий зовнішнього кола. Усі вхідні значення зберігаються в полі HASH. Це структура даних, невпорядкований список пар ключ/значення, в якому можна легко шукати за ключем. У даному випадку ключем є IP-адреса, а значенням, яке їй присвоюється, є вже згадана затримка AB.

• Якщо припустити, що MGTOKlientDoslidPC.jar отримав значення з орієнтирів, перелічених у таблиці нижче, тоді значення не відповідають реальним вимірюванням, але вибрані таким чином, щоб наступний приклад був якомога наочнішим.

Таблиця 3.2.

Тестові дані

Орієнтир	Центр	Радіуси кіл		Радіус нових кіл
		Внутрішній	Зовнішні	
1	A (500500)	400	500	466,6666667
2	B (1350 450)	400	500	466,6666667
3	C (900 900)	200	700	533,3333333

• На наступному кроці програма створює з цих значень кільцевий простір із

центром біля орієнтира, з якого вийшло дане значення затримки AB. Графічне зображення наведено на рис. 3.10. Внутрішні кола (негативні обмеження обмеження), в яких ціль не лежить, позначені червоним кольором, а зовнішні кола (обмеження позитивного обмеження), в яких ціль більше не знаходиться, показані червоним кольором. Виділене синім кольором виділяє область, в якій ціль найімовірніше лежить.

• Після отримання всіх проміжних кіл (межі обмеження) створюються нові кола з радіусом, рівним радіусу внутрішнього кола плюс ширина кільця. Радіуси цих кіл наведені в останньому правому стовпці таблиці 3.2. Цей коефіцієнт був визначений найбільш точним у довгостроковій оцінці. Графічний дисплей наведено на рисунку 3.11. Використовуючи ці кола, можна обмежити область пошуку і отримати з неї позицію цільового вузла.

• Потім усі перетини цих нових кіл підраховуються послідовно (на рис. 3.11) і зберігаються в полі. Якщо з'являється перехрестя з тими ж координатами, що і вже збережене перехрестя, воно видаляється . Результат можна побачити на рис. 3.12.

• Останнім кроком програми є видалення перехресть, що знаходяться за межами визначеної області, та обчислення положення цілі. Це означає, що якщо перетин лежить поза областю, обмеженою кільцем m (рис. 3.10), воно видаляється. Видаляються також перетини, які перебувають не в перетині всіх проміжних кілець, тобто лежить поза синьою зоною на рис. 3.10.

Потім решта перехресть утворюють область, що обмежує появу цілі. Потім програма замінює цю область багатокутником, сторони якого утворені лініями, що з'єднують окремі перехрестя, які насправді є вершинами цього багатокутника [23].

Програма спочатку обчислює вміст створеного багатокутника за такою формулою:

$$A = \frac{1}{2}\sum_{i=0}^{n-1}(x_i * y_{i+1} - x_{i+1} * y_i), \tag{3.1}$$

де змінні x_t і y_t є координатами згаданих перетинів (вершин багатокутників), a_n - їх загальна кількість. Центр ваги багатокутника є шуканим положенням цільової точки і обчислюється за допомогою таких рівнянь:

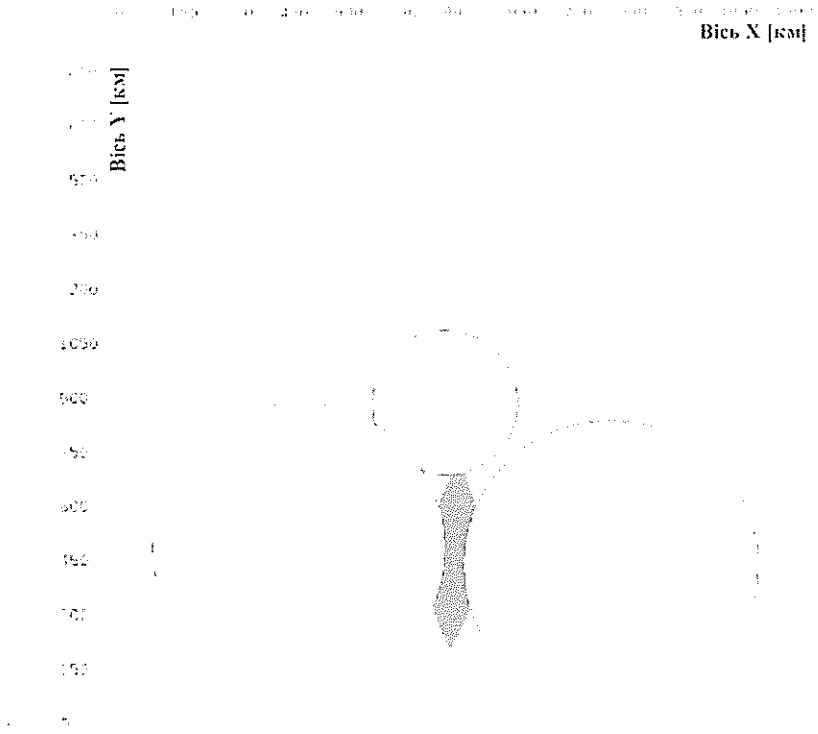


Рис. 3.10. Розраховані граничні значення

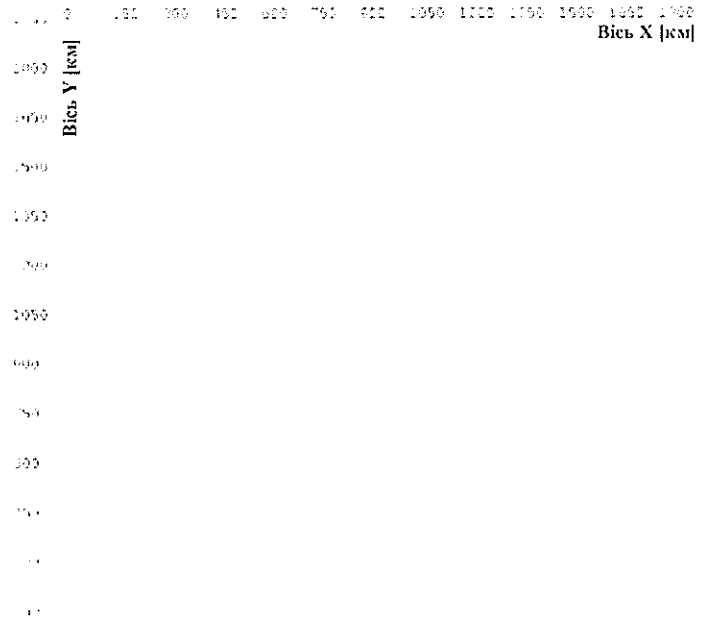


Рис. 3.11. Розрахункові значення нових кіл

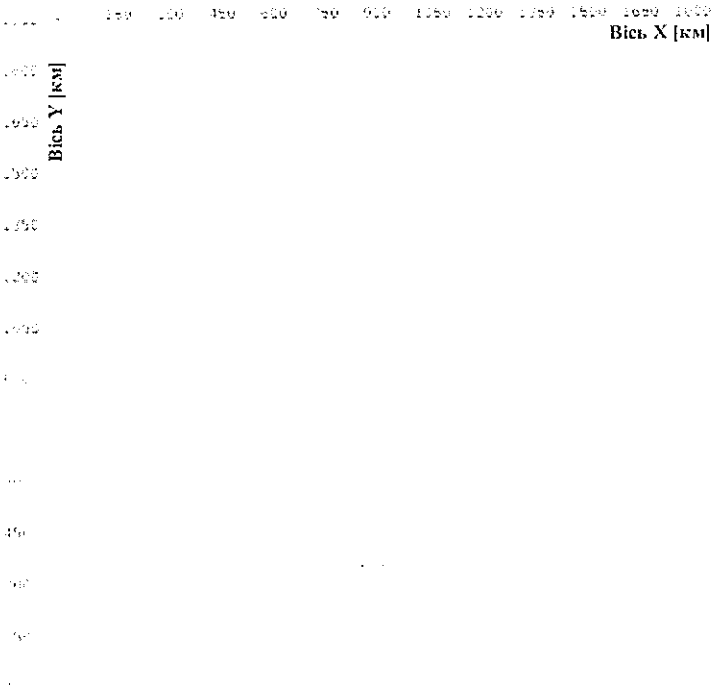


Рис. 3.12. Розрахункові перетини

$$C_x = \frac{1}{6A}\sum_{i=0}^{n-1}(x_i + x_{i+1}) * (x_i * y_{i+1} - x_{i+1} * y_i))$$
$$C_y = \frac{1}{6A}\sum_{i=0}^{n-1}(y_i + y_{i+1}) * (x_i * y_{i+1} - x_{i+1} * y_i)), \quad (3.2)$$

де C_x і C_y вказують координати цілі. Результат роботи програми показаний на рисунку 3.13, ціль - синім кольором [23].

• Знайдена точка призначення має координати (943 932; 514 272). Потім він перетворюється в систему географічних координат у співвідношенні, де 111 330 км по вертикальній осі відповідає 1 ° широті, а 64 152 км по горизонтальній осі відповідає 1° довготи. Після цього регулювання, мета була 14,714 географічні координати по довготі і 4619 по широті.

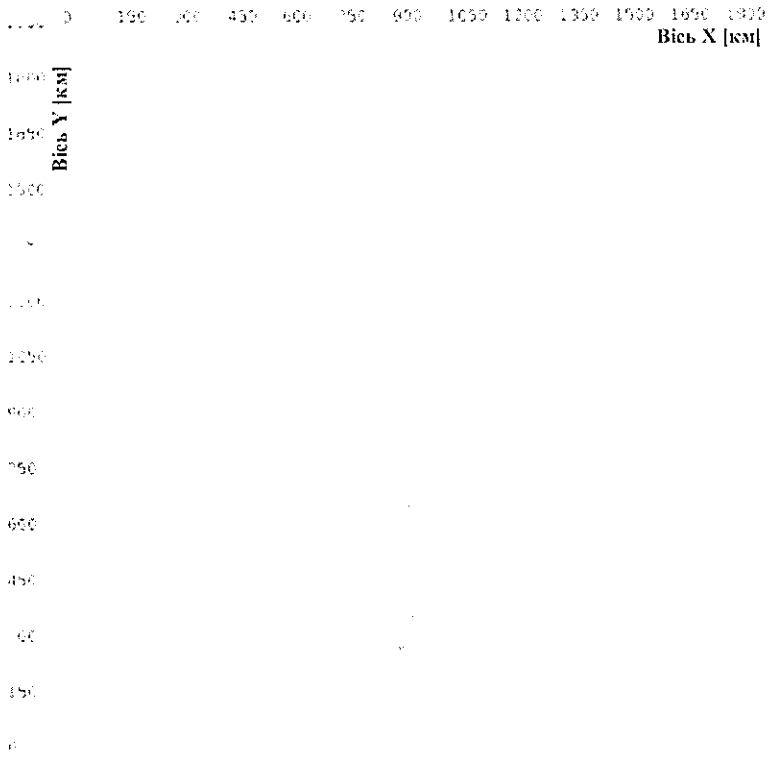


Рис. 3.13. Розрахунковий перетин цілі

3.4 Вимірювання в реальній мережі

Вище була описана робота програми лише над штучними даними. Якби ціль з IP-адресою 131.188.32.37 була розташована і використовувались усі 40 опорних орієнтирів, графічний результат програми мав би виглядати так.

На рисунку 3.14 показані межі обмежень усіх орієнтирів, а на рисунку 3.15 - перехрестя, які лежать у зоні, обмеженій усіма проміжними колами. Знайдена точка призначення виділена синім кольором.

Вихідні дані MGTOServerPlanetLab.jar, наприклад з IP-адресою 131.188.32.37, виглядатимуть так:

[{"id": 236, "duration": 34,0 "image": "Mgto.png", "hostname": "2009.nwerc.eu", "method": "Mgto", "longitude": 8.729427720383182, "latitude": 51.5295843289 2871, "comme nt ": 134971.3273397023, "ip": "131.188.32.37", "idmethod": 13}.

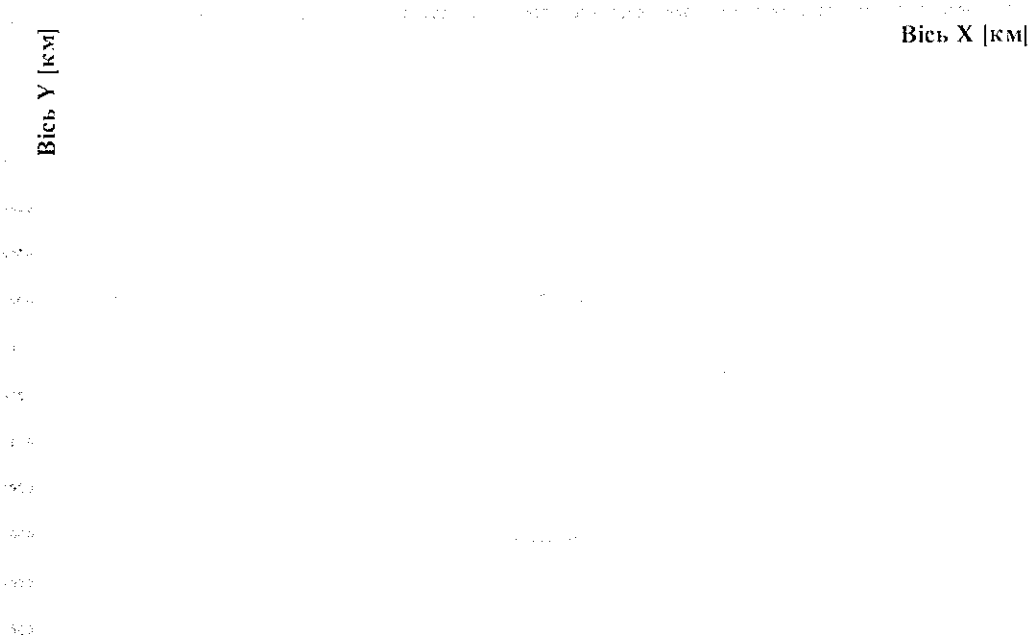


Рис. 3.14. Обмеження для вузла з IP-адресою 131.188.32.37

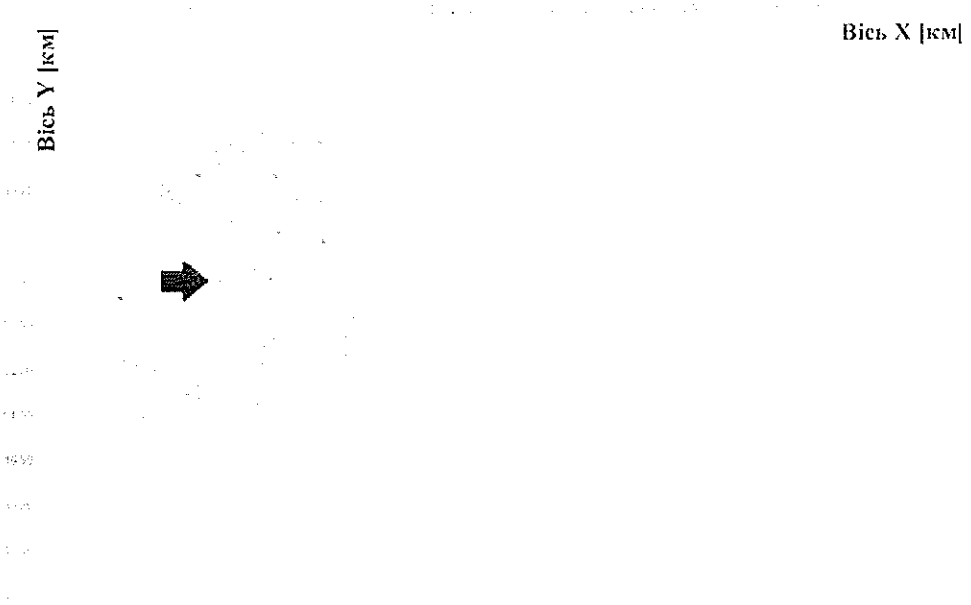


Рис. 3.15. Пункт призначення для шуканого пункту призначення з IP-адресою 131.188.32.37

Області, позначені синім кольором, є значеннями, які були додані до програми на виході. Поле коментаря відображає вміст багатокутника, а поле тривалості вказує, скільки часу працює програма.

Проблеми під час практичного розгортання

Програма вперше була створена та протестована на штучних даних. Після

успішних випробувань її було введено в реальну експлуатацію. Однак цей крок не пройшов без проблем. Деякі з них будуть перелічені в далі. Під час аналізу вдалося записати всі дані в окремі текстові файли. Тому в кожній програмі була створена функція, яка називається Журнал, що забезпечує перелік несподіваних завершень програми та її нетипову поведінку у так званих файлах журналів. Файли зберігаються в каталозі, з якого запущена програма.

Завантажити на вузли PlanetLab

Під час перших тестів програма була завантажена лише в один вузол, де вона була протестована. Були використані програми для багатокопіювання та багатозапитів із пакету Codeploy. Багатокопіювання, як випливає з назви, використовується для копіювання файлів у вузли. Мультизапит використовується для віддаленого виконання команд. Однак ці інструменти не дуже підходять для управління великою кількістю вузлів. Краще скористатися програмою Plman. Ця програма доступна як для дистрибутивів Linux, так і для Windows.

Програма має простий графічний інтерфейс (рис. 3.16) і налаштована за допомогою текстового файлу pl_manager.conf. Структура файлу не буде обговорюватися більш докладно, будуть перераховані лише дані, необхідні для запуску програми:

• Ім'я користувача – тут має бути вказано ім'я для входу в мережу PlanetLab

• Зріз – ім'я призначеного набору вузлів, у нашому випадку це cesnet_vutbr2

• PrivateKeyFile – тут потрібно вказати явний шлях до сформованого ключа

Після зміни та збереження файлу конфігурації всю програму можна запустити за допомогою команди sh run_linux_x86.sh, а потім підключитися до керованих вузлів за допомогою одного з параметрів на вкладці «Підключення». У даному випадку використано опцію Підключитися до хостів у файлі, що дозволило підключитися до всіх вузлів, що зберігаються у зовнішньому текстовому файлі [34].

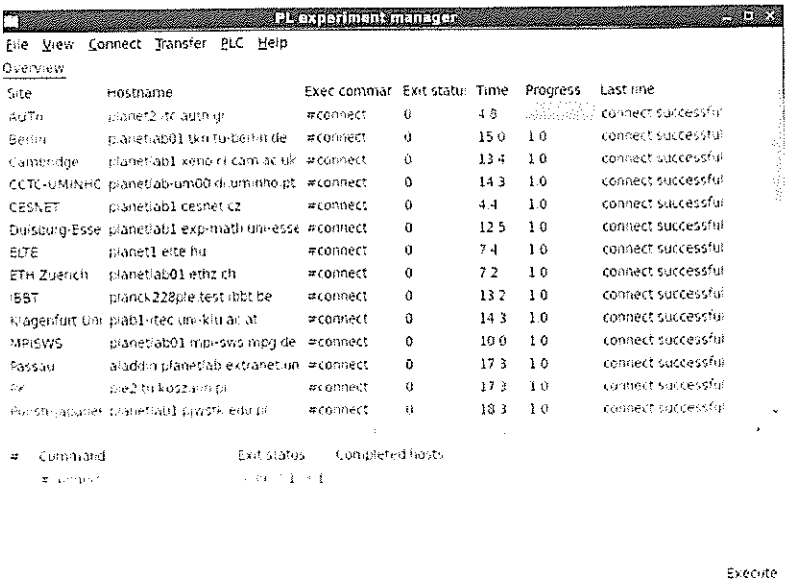


Рис. 3.16. Програма для віддаленого управління мережевими вузлами PlanetLab Plman

Проблеми з пінгом

При створенні графіку затримки/відстані було виявлено, що деякі вузли відповідають на запити ICMP відносно довгим або дуже коротким часом. У таблиці 3.3 перелічені деякі "нісенітницькі" дані.

Таблиця 3.3.

Виміряні значення затримки за допомогою запиту ICMP

IP-адреса вузла	Відстань від вузла 193.166.88.16 [км]	Виміряний пінг [мс]	Швидкість передачі [км/с] RTT - двонаправлена
213.131.1.101	1014,313111	1531,06	близько 1300
130.237.50.124	1856 098	11.18	приблизно 332 038

Ці значення принципово вплинули на формування верхнього/нижнього конверта і, отже, загальну точність отриманого положення. Тому потрібно було додати додаткові функції в програму RozraxPakety.jar r:

• Функція перевірки, чи не поширюється сигнал у з'єднанні швидше, ніж

швидкість світла (лінія c)

• Функція перевизначення всіх значень затримки, що перевищують 250 мс

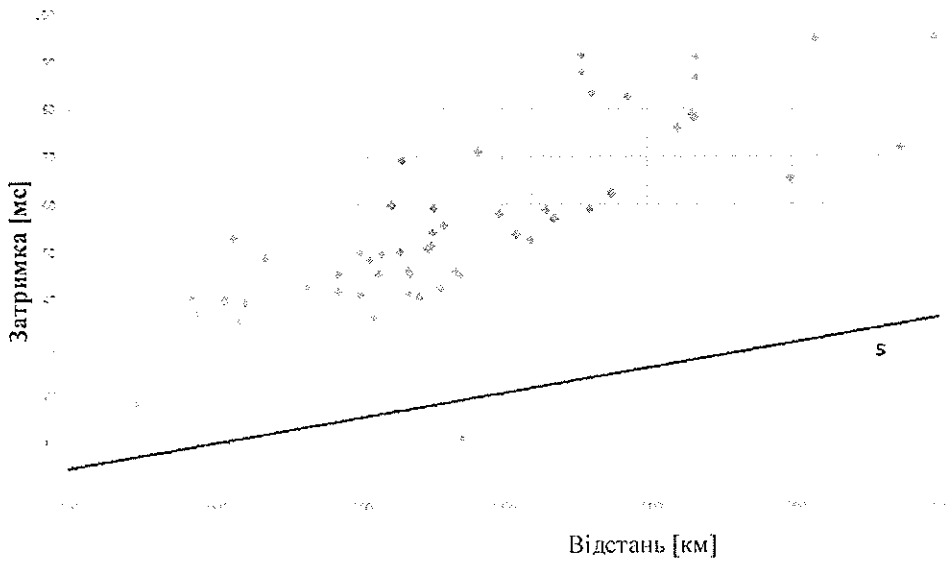


Рис. 3.17. Графік затримки/відстані на вузлі 193.166.154.29

На рисунку 3.17 показано вузол (130.237.50.124), звідки відгук надходив за більш швидкий час, ніж сигнал може рухатися в оптичному волокні (розташованому нижче лінії s). Очевидно, що якби вузол був включений у розрахунок нижньої оболонки, це суттєво вплинуло б на його нахил і, таким чином, без потреби розширило б межі межею. Причиною такої затримки може бути неправильно визначене географічне положення орієнтира, коли вузол фактично розташований ближче/далі, ніж передбачалося.

Змінна величина затримки та проміжного кільця

Оригінальна програма була створена з припущенням, що значення затримки та ширина кільцевого кільця збільшуватимуться із збільшенням відстані орієнтира від цілі. Це припущення добре працювало над штучними даними, але на практиці виявилося помилковим. Було показано, що велика затримка і ширина кільцевого кільця не обов'язково означають велику відстань і навпаки. Наприклад, вузол з IP-адресою 155,207,48,21 на відстані 1077 км від цілі мав ширину затрубного простору 1440 км, а для порівняння, вузол 88,255,65,219 на відстані 1877,6 км від цілі мав ширину затрубного простору 1340 км.

Тому від попередньої концепції програми довелося відмовитись, коли кола спочатку розташовувались відповідно до розміру затрубного кільця (від найменшої до найбільшої відстані від цілі), а потім перевірялись по черзі на перехрестя. З кожним додатковим колом область поступово зменшувалась, поки не була знайдена цільова область, де ціль, швидше за все, лежала.

Використання вторинних вузлів

Через те, що іменування маршрутизаторів компаній, що надають Інтернет, не є уніфікованим або стандартизованим (компанія UPC - te-0-6-0-0.tr1.pop1.pra.sloane.com, компанія NWT - hulincd.nwt.com), було від використання вузлів вторинної орієнтації методом МГТО було відмовлено.

Підключення до вузлів

У першій версії програми для завантаження даних (межі відстані) з вузлів використовувалося з'єднання за допомогою протоколу SSH. Незважаючи на переваги безпеки (автентифіковане та зашифроване з'єднання), цей протокол було відкликано. Виявилося, що час, необхідний для встановлення зв'язку між шкільним сервером (на якому працює програма MGTOKlientDoslidPC.jar), та вузлом мережі PlanetLab становить близько 50 секунд. Загальний час роботи на грам із 90 орієнтирами часто піднімався до 14 хвилин. З яких близько 13 хвилин взяли лише з'єднання [34].

Нещодавно була створена програма TCPServer.jar, яка навіть за рахунок меншої безпеки (незашифрованого з'єднання та відкритого порту) скоротила робочий час програми з попередніх 14 хвилин приблизно до 3 хвилин для одного пункту призначення.

4 ПОРІВНЯННЯ ТОЧНОСТІ МЕТОДУ МГТО З ІНШИМИ МЕТОДАМИ ГЕОПОЗИЦІОНУВАННЯ

4.1 Результати методу МГТО

Вимірювання показують, що середня похибка точності методу МГТО становить 241,73 км, а медіана цієї похибки становить 237,75 км. Похибка здебільшого коливалась від 47,27 км до 597,80 км. Але помилки, що перевищують цей інтервал, також з'явилися на кількох вузлах. Навіть при неодноразових спробах дві опорні цілі показали помилку понад 950 км - вузол з IP-адресою 213.245.116.198 мав похибку оцінки 951,24 км, а вузол з IP-адресою 195.64.165.26 навіть мав помилку 1039,92 км.

Швидкість алгоритму при підключенні через протокол SSH становила в середньому 592 с (приблизно 10 хв) для пошуку однієї цілі. Після відключення протоколу SSH та використання більш простого з'єднання лише за допомогою протоколу TCP середній час для оцінки цільової позиції впав до 28,8 с. Цей факт вже значно проявився в очікуванні результатів оцінки положення.

4.2 Результати методу МГТО в порівнянні з іншими методами

У таблиці 4.1 наведені середні помилки точності, включаючи медіану та цільову швидкість пошуку, за всіма методами.

Тому порівнюються всі методи за допомогою помилок точності (медіана) методом GeoIP. Якщо порівнювати лише активні методи, то найточнішим є SOI, за яким слід GeoWeight. Далі слідують методи CBG та МГТО.

Як видно з таблиці 4.1, пасивний метод GeoIP значно перевершив усі активні методи геопозиціонування не тільки з точки зору точності, але і швидкості. Це означає, що положення вузлів у його базі даних відносно точні, і в багатьох випадках можна покладатися на їх точність. Перевагою є також дуже короткий час від надсилання запиту до цільової позиції до відповіді. Насправді це лише питання пошуку необхідних записів у переважно локальній базі даних.

Однак перелічені переваги застосовуються лише в тому випадку, якщо цільовий запис знайдено в базі даних.

Таблиця 4.1.

Порівняння методів CBG, SOI, МГТО, GeoWeight, GeoIP

Метод	Похибка точності [км]		Швидкість локалізації однієї цілі [мс]	
	Середня похибка точності	Медіана	Середній	Медіана
МГТО	259.70	237,22	58,76	28.00
GeoIP	120,39	13.22	0,00	0,00
CBG	284,69	192,87	78,48	59,00
SOI	183,77	172,99	21.61	21.49
GeoWeight	239,85	184,76	349,48	343,39

Найбільшим "сюрпризом" стала точність методу SOI. Цей метод, хоча і не пропонує жодної додаткової функції, наприклад, автоматичного калібрування затримки між ОВ або заміни прямої лінії складною оболонкою, був найбільш точним після методу GeoIP.

Цікаво також порівняти методи SOI та CBG, які працюють за подібним принципом. Вони відрізняються лише створенням так званої найкращої лінії (лінії, що вказує відношення затримки/відстані). Метод SOI має цю лінію, яка відповідає швидкості світла 4/9, і CBG змінює її залежно від затримки між орієнтирами. Хоча може здатися, що метод CBG буде більш точним із цим "вдосконаленням", практичні тести показали протилежне. Метод SOI із фіксованим кращим коефіцієнтом 1 і не був точнішим у середньому на 100 км (у випадку медіани 20 км), ніж CBG.

Метод GeoWeight, який використовує так звані шкали для визначення положення цілі, перевершив методи CBG та МГТО із середньою точністю. Тому

система зважування довела свою практичну користь.

Метод МГТО, хоча він пропонує обмеження цільового положення за допомогою складного огинаючого з обох сторін та автоматичного калібрування затримки між ОВ, був поміщений між методами GeoWeight та CBG із середньою точністю. Виявилося, що складна оболонка дійсно визначає межі обмежень, за допомогою яких вона потім створює кільцеву простір для грама. Але на практиці ці сфери все ще занадто великі. Дякого вдосконалення можна досягти, включивши вторинні вузли та географічні обмеження. Однак отримати розташування маршрутизатора з його імені DNS і класифікувати його як інший контрольний вузол, додатково отриману область, де знаходиться ціль, щоб проаналізувати, чи знаходиться якась його частина в населеній чи безлюдній області, видається дуже складним рішенням. Особливо, коли практичні тести показали, що у багатьох випадках використання більш простого рішення є вигіднішим.

4.3 Графічне представлення результатів із використанням CDF

Функція кумулятивного розподілу виражає ймовірність появи значення X, яке менше або дорівнює вибраному значенню x. Таким чином, математичним визначенням є F (x) = P (X <x ') [35].

На рис. 4.1 показана ця функція лише для методу МГТО. Середня похибка точності методу знаходиться в інтервалі <0; 1000> км.

З наступного графіка можна прочитати, що 37,8% еталонних цілей були розташовані з похибкою точності менше 150 км. Якби розширили територію до 300 км, це вже покривало б 65,15% цілей. 92,42% цілей буде локалізовано методом МГТО з похибкою менше 500 км.

З загальної кількості 76 вузлів метод МГТО не зміг знайти 8 вузлів. Причиною можуть бути неправильно визначені значення із затримки, через які межі обмеження не визначили правильно область (багатокутник) і, отже,

результуюче положення цілі не вдалося знайти.

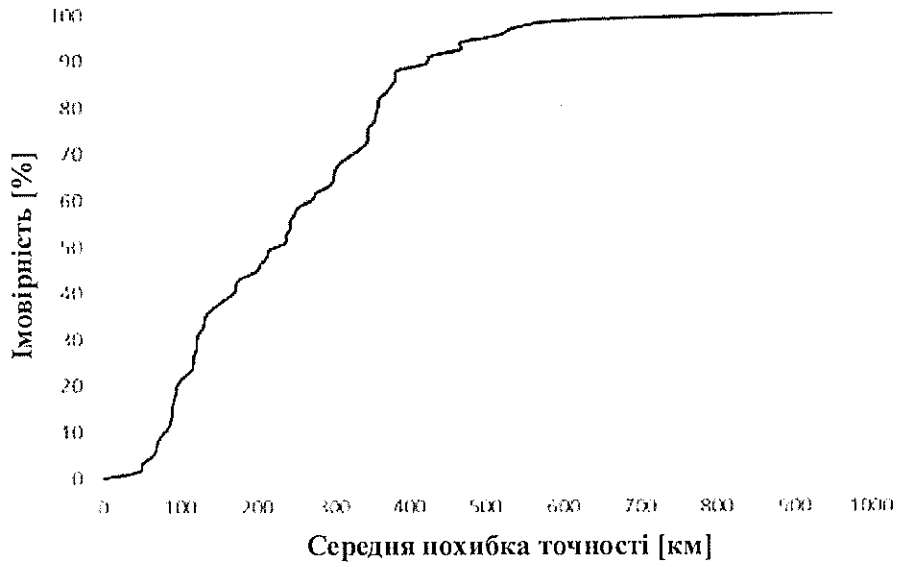


Рис. 4.1. Кумулятивні функції розподілу – метод МГТО

На рис. 4.2 наведено CDF для всіх перевірених методів. Криві кожного методу визначають ймовірність того, що похибка точності буде дорівнює або менше кількості км, показаної на осі x.

Графік, що належить методу GeoIP, очевидний на перший погляд. Цей метод має найбільше значення ймовірності для більшості значень на осі x. Простіше кажучи, це "найточніше". Єдине місце, де це не так, - близько 200 км. Тому, якщо знаходяться цілі з похибкою точності до 200 км, метод SOI дасть кращі результати.

На додаток до методів SOI та GeoIP, інші методи мають подібні значення ймовірності приблизно до 140 км, де метод GeoWeight починає приймати кращі значення та наближається до методу SOI. З похибки точності більше 200 км значення імовірності зростає для всіх активних методів, крім методу МГТО. Значення ймовірності методу МГТО починає значно зростати лише з похибкою точності 380 км, коли воно перевершує інші активні методи.

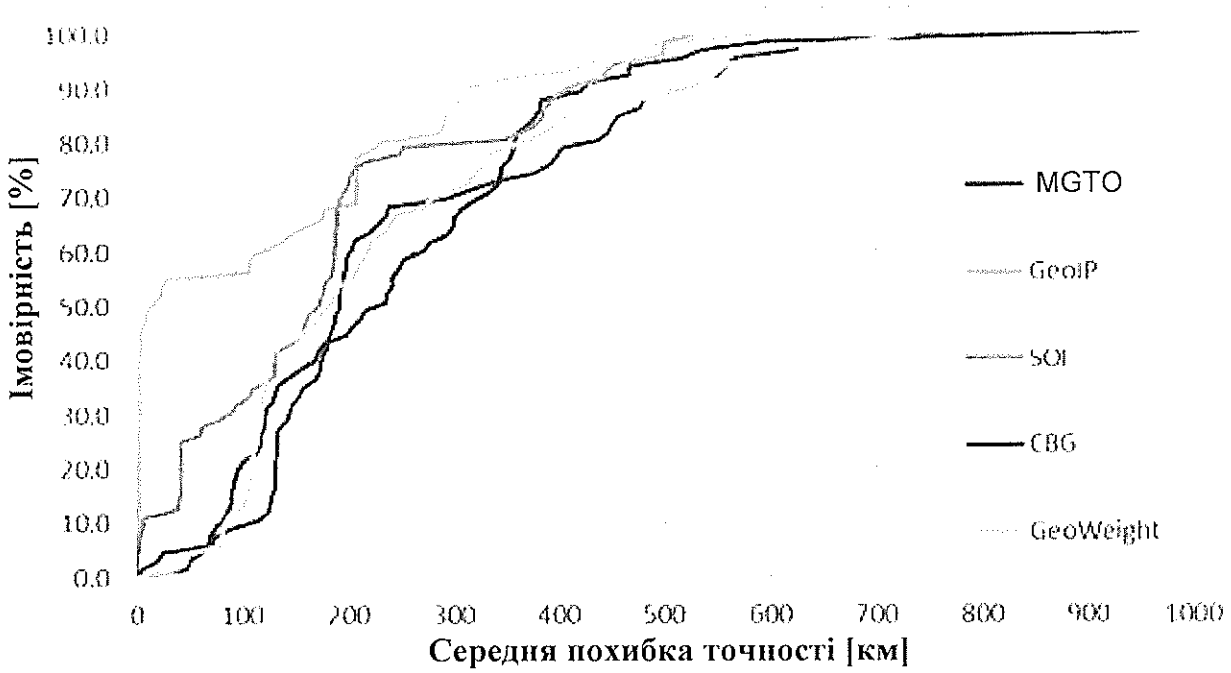


Рис. 4.2. Кумулятивні функції розподілу – методи MGTO, SOI, CBG, GeoWeight та GeoIP

4.4 Графічне представлення результатів KDE

Оцінка щільності ядер – це спосіб отримання згладженої оцінки щільності ймовірності (кривої частоти) випадкової величини. Таким чином, можна отримати графічне представлення найпоширеніших значень помилок точності для кожного методу геопозиціонування. Єдиним недоліком KDE є вибір ширини згладжуючого вікна. Якщо вікно встановлено на невелику ширину, отримана функція занадто «згладжена». В іншому випадку у функції виникають великі піки. Макрос, доступний за цією веб-адресою, можна використовувати для обчислення KDE в Excel.

Для методу MGTO оцінка щільності ядра показана на рисунку 4.3. У випадку з цим методом найпоширенішим значенням є похибка точності близько 120 км. З цією точністю розташовано 27,3% еталонних цілей. Другий менший пік становить близько 308 км, коли 66,69% цілей з цією похибкою точності вже розташовані. Крива методу MGTO не має таких крутих піків, як інші методи, а це означає, що його значення більш розподілені.



Рис. 4.3. Оцінка щільності ядра – метод МГТО

Оцінка щільності ядра для всіх методів показана на рис. 4.4. Тому для кожного методу можна відняти, які значення помилок точності зустрічаються найчастіше.

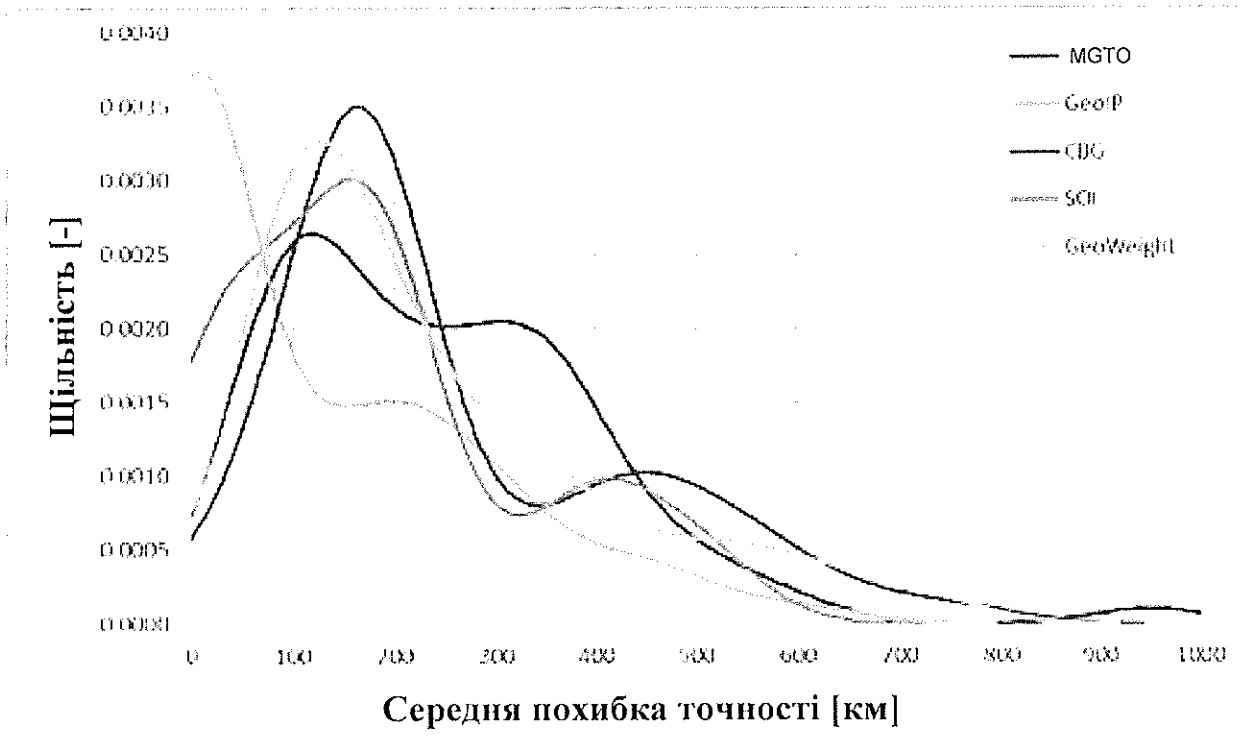


Рис. 4.4. Оцінка щільності ядра – методи МГТО, SOI, CBG, GeoWeight та GeoIP

Для методу GeoIP це значення близько 30 км по осі x. Це означає, що GeoIP визначає більшість цілей (із даного списку опорних точок) з похибкою точності менше 30 км. Якщо повернутися до графіку CDF, метод GeoIP локалізував близько 55% цілей з цією помилкою оцінки. Інший пік для цього методу становить близько 2088 км.

Інші методи геопозиціонування мають найвищу щільність близько 160 км. Метод GeoWeight у 130 км, SOI у 160 км та CBG у 170 км.

ВИСНОВКИ

Приведено аналіз найважливіших фактів IP-геолокації в мережі Інтернет, що дає змогу проникнути в принципи найбільш часто використовуваних методів геопозиціонування.

Найбільший обсяг робіт присвячений методу геопозиціонування, який називається МГТО. Цей відносно новий метод дозволяє використовувати низку вдосконалених методів (використання негативної інформації тощо) для уточнення оцінки цільової позиції. Хоча цей метод детально описаний у дослідженні Бернарда Вонга, використовувані ним механізми пояснюються лише загальними словами. Тому було необхідно розробити кілька математичних рішень та процедур, які призвели б до якомога більшої точності та відповідали умовам, наведеним у дослідженні, що стосується цього методу.

Після перевірки функціональності та методу МГТО на штучних даних було розпочато тестування в експериментальній мережі PlanetLab. З самого початку виникло кілька проблем. Оригінальна програма передбачала, що кола (межі кордонів з центром в орієнтирі) спочатку будуть відсортовані за розміром їх кільцевого кільця. Припускалося, що застосовується умова, що найдальші орієнтири від цілі матимуть найбільші значення радіусів і в той же час не визначатимуть точно шукану площу. Потім кола поступово переміщувались до обчислення, і таким чином поступово зменшували та уточнювали область пошуку.

На практиці було доведено, що цей метод застосовувати не можна. У міру зменшення відстані часто траплялося так, що інші кола не перетиналися, і тому отримана площа, де найімовірніше лежала ціль, була дуже великою. Приклад, коли вузол з IP-адресою 155.207.48.21 на відстані 1077 км від цілі мав ширину кільця 1440 км, а для порівняння вузол 88.255.65.219 на відстані 1877.6 км від цілі мав ширину кільця 1340 км.

В кінці роботи метод піддається порівнянню з іншими активними методами CBG, SOI, GeoWeight і пасивним методом GeoIP на тих самих даних (вузлах).

Ідентифіковано 40 вузлів опорної орієнтації та 76 вузлів призначення. Хоча може здатися, що знайти 76 вузлів в Інтернеті буде просто, проте це не так. Ці вузли повинні відповідати декільком умовам (вони повинні відповідати на запит ICMP, вони знаходяться в Європі тощо), найважливішою з яких є точне знання їхньої позиції. Це слід було визначити, використовуючи DNS-запис типу LOC. Однак виявилося, що із вільно доступного списку з 4,5 мільйонів адрес лише 181 (приблизно 0,004%) станцій підтримували цей тип запису. Після ручної перевірки вказаного положення – щоб їх координати не вказували на океан тощо, залишилось лише 90 станцій. З них лише зазначені 76 станцій відповідали заданим умовам. Результати порівняння методу МГТО та інших методів, їх точність порівнювали на основі середньої похибки точності (тобто оціненої відстані положення цілі від її фактичного), медіана та час розрахунку. Загальне порівняння, що є ще більш очевидним, використовує кумулятивну функцію розподілу та оцінку щільності ядра. Метод МГТО, хоча не всі його аксесуари були впроваджені, був би порівнянним з іншими методами з похибкою точності до 130 км.

Хоча в наш час використовуються досить пасивні методи геопозиціонування, думається, що використання активних методів збільшиться. Хоча вони не досягають точності пасивних, вони мають ту перевагу, що можуть знайти будь-який комп'ютер, який може відповісти на запит ICMP.

ПЕРЕЛІК ПОСИЛАНЬ

1. Team foursquare [Електронний ресурс] // Foursquare blog - Режим доступу: http://blog.foursquare.com (дата звернення 17.10.2023).

2. Соціальна мережа Foursquare [Електронний ресурс] Wikipedia // - Режим доступу: https://uk.wikipedia.org/wiki/Foursquare (дата звернення 17.10.2023).

3. Інструменти для веб-майстрів [Електронний ресурс] Google // - Режим доступу: http://www.google.com/support/webmasters/bin/answer.py?hl=en&answer =62399 (дата звернення 21.10.2023).

4. Використання геолокації в соціальних мережах [Електронний ресурс] Комп'ютерні технології та Інтернет в інформаційному суспільстві // - Режим доступу: http://ir.lib.vntu.edu.ua/bitstream/handle/123456789/22584/107B9.pdf (дата звернення 21.10.2023).

5. Geolocation Technology. What is GeoIP? [Електронний ресурс] MaxMind // - Режим доступу: http://www.maxmind.com/app/ip-location (дата звернення 21.10.2023).

6. RFC 1834. Whois and Network Information Lookup Service [Електронний ресурс] // - Режим доступу: http://www.faqs.org/rfcs/ rfc1834.html (дата звернення 25.10.2023).

7. Creating a DNS Rule Using the DNS Rule Wizard [Електронний ресурс] // - Режим доступу: btp-gss1.dlapiper.com (дата звернення 06.11.2023).

8. NSlookup. Using NSlookup.exe [Електронний ресурс] Microsoft Support // - Режим доступу: http://support.microsoft.com/kb/200525 (дата звернення 06.11.2023).

9. Debian GNU/Linux. DNS Records Explained with Examples. Debian Help. [Електронний ресурс] // - Режим доступу: http://www.debianhelp.co.uk/dnsrecords.htm (дата звернення 06.11.2023).

10. RFC 1876. A Means for Expressing Location Information in the Domain Name System [Електронний ресурс] // - Режим доступу: http://tools.ietf.org/html/rfc1876 (дата звернення 10.11.2023).

11. Google Collect User Data [Електронний ресурс] Wall Street Journal –

technology // - Режим доступу: http://www.online.wsj.com/article/SB10001424052748703983704576277101723453610.html (дата звернення 12.11.2023).

12. Geolocation in the Browser [Електронний ресурс] Mozilla Web Development – technology // - Режим доступу: http://www.blog.mozilla.org›2019/05/01 (дата звернення 12.11.2023).

13. Google code. Geolocation API Network Protocol [Електронний ресурс] Google – technology // - Режим доступу: https://developers.google.com/maps/documentation/geolocation/overview (дата звернення 12.11.2023).

14. Геолокація в браузері [Електронний ресурс] CoderLessons.com // - Режим доступу: https://coderlessons.com/articles/mobil'na-rozrobka-artykliv/geolokatsiya-v-brauzeri (дата звернення 12.11.2023).

15. Інтернет як засіб соціальної комунікації та соціального впливу [Електронний ресурс] Соціологія // - Режим доступу: http://nbuv.gov.ua/j-pdf/PoMe_2018_4_14.pdf (дата звернення 15.11.2023).

16. Маршрутизація в IP мережах [Електронний ресурс] ДУТ // - Режим доступу: http://www.dut.edu.ua/uploads/l_1215_49011858.pdf (дата звернення 15.11.2023).

17. Моделі якості послуг телекомунікаційних мереж [Електронний ресурс] // - Режим доступу: http://ena.lp.edu.ua:8080/bitstream/ntb/41018/2/2017n864_Horbatyi_I-Modeli_yakosti_posluh_192-200.pdf (дата звернення 15.11.2023).

18. Телекомунікаційні системи та мережі наступного покоління [Електронний ресурс] ДУТ // - Режим доступу: http://www.dut.edu.ua/uploads/l_1762_12226724.pdf (дата звернення 15.11.2023).

19. RFC 792 - Internet Control Message Protocol [Електронний ресурс] ДУТ // - Режим доступу: http://tools.ietf.org/html/rfc792 (дата звернення 16.11.2023).

20. Охорона об'єктів інтелектуальної власності у мережі Інтернет [Електронний ресурс] // - Режим доступу: https://www.slideshare.net/Comedianua/ss-78009492 (дата звернення 16.11.2023).

21. An Investigation of Geographic Mapping Techniques for Internet Hosts

[Електронний ресурс] Internet Measurement Conference // - Режим доступу: http://conferences.sigcomm.org/sigcomm/2021/p14-pabmanabhan.pdf (дата звернення 16.11.2023).

22. A Comprehensive Framework for the Geolocalization of Internet Hosts [Електронний ресурс] USENIX Symposium on Networked Systems Design & Implementation // - Режим доступу: https://www.usenix.org/ events/nsdi07/tech/full_papers/wong/wong.pdf (дата звернення 16.11.2023).

23. Constraint-Based Geolocation of Internet Hosts. [Електронний ресурс] ACM/SIGCOMM // - Режим доступу: http://www.cs.bu.edu/fac/crovella/paper-archive/imc04-geolocation-full.pdf (дата звернення 16.11.2023).

24. Towards IP Geolocation Using Delay and Topology Measurements [Електронний ресурс] // - Режим доступу: http://conferences.sigcomm.org/ imc/2016/papers/p7-bassett.pdf (дата звернення 16.11.2023).

25. GeoWeight: Internet Host Geolocation Based on a Probability Model for Latency Measurements [Електронний ресурс] Australasian Computer Science Conference // - Режим доступу: http://dl.acm.org/citation.cfm?id=1862209 (дата звернення 19.11.2023).

26. Стеклов В. К. Підходи до ситуаційного управління телекомунікаційними мережами / В. К. Стеклов, Л. Н. Беркман, Л. В. Рудик, А. С. Стец // Зв'язок. – 2005. – №1. – С. 47–57.

27. Стеклов В. К. Оптимізація та моделювання пристроїв та систем зв'язку : підруч. Для вищ. навч. закл. / В. К. Стеклов, Л. Н. Беркман, Є. В. Кільчицький; за ред. В. К. Стеклова. – К.: Техніка, 2004. – 576 с.

28. Measuring ISP Topologies with Rocketfuel [Електронний ресурс] Computer Science and Engineering University of Washington // - Режим доступу: http://www.cs.washington.edu/research/networking/rocketfuel/ (дата звернення 23.11.2023).

29. PuTTy - a free SSH and telnet client for Windows [Електронний ресурс] // - Режим доступу: http://www.chiark.greenend.org.uk/~sgtatham/putty/ (дата звернення 23.11.2023).

30. PlanetLab An open platform for developing, deploying, and accessing planetary-scale services [Електронний ресурс] PlanetLab // - Режим доступу: http://planet-lab.org/. (дата звернення 23.11.2023).

31. PlanetLab European Tutorial. An open, shared platform for developing, deploying, and accessing planetary scale applications [Електронний ресурс] PlanetLab // - Режим доступу: http://www.ict-fireworks.eu/fileadmin/events/FIREweek/PL_1.pdf. (дата звернення 23.11.2023).

32. DMOZ - найбільший каталог посилань, яким керує людина [Електронний ресурс] Wikipedia // - Режим доступу: https://uk.wikipedia.org/wiki/DMOZ (дата звернення 23.11.2023).

33. Calculate distance, bearing and more between Latitude / Longitude points. Movable Type Scripts [Електронний ресурс] // - Режим доступу: http://www.movable-type.co.uk/scripts/latlong.html (дата звернення 26.11.2023).

34. Jan S. Krogh's Geosite: Other Places of Interest. [Електронний ресурс] The Centre of Europe // - Режим доступу:http://geosite.jankrogh.com/other.htm (дата звернення 26.11.2023).

35. Working with GeoLocations [Електронний ресурс] Xebee // - Режим доступу: http://xebee.xebia.in/2019/10/28/working-with-geolocations/ (дата звернення 26.11.2023).

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Презентація по магістерській роботі на тему:

РОЗРОБКА МЕТОДУ ГЕОІНФОРМАЦІЙНОГО ПОЗИЦІОНУВАННЯ ТЕЛЕКОМУНІКАЦІЙНОГО ОБЛАДНАННЯ

Студент Львов Максим Олексійович

МЕТА РОБОТИ

Об'єкт дослідження – процес геоінформаційного позиціонування об'єктів.

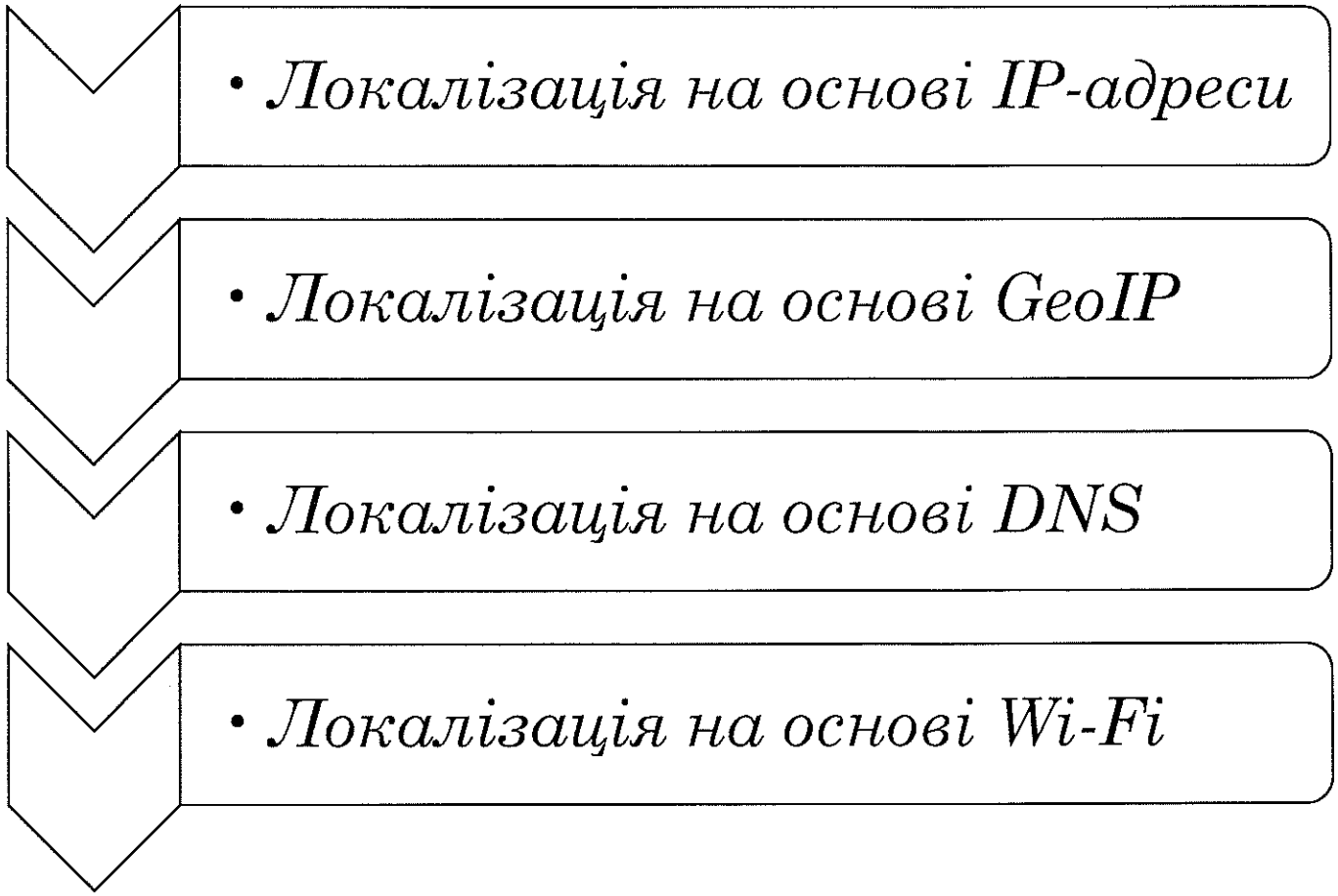
Предмет дослідження - методи геоінформаційного позиціонування для визначення місцезнаходження телекомунікаційного обладнання.

Мета роботи – розробка методу геоінформаційного позиціонування для визначення місцезнаходження телекомунікаційного обладнання.

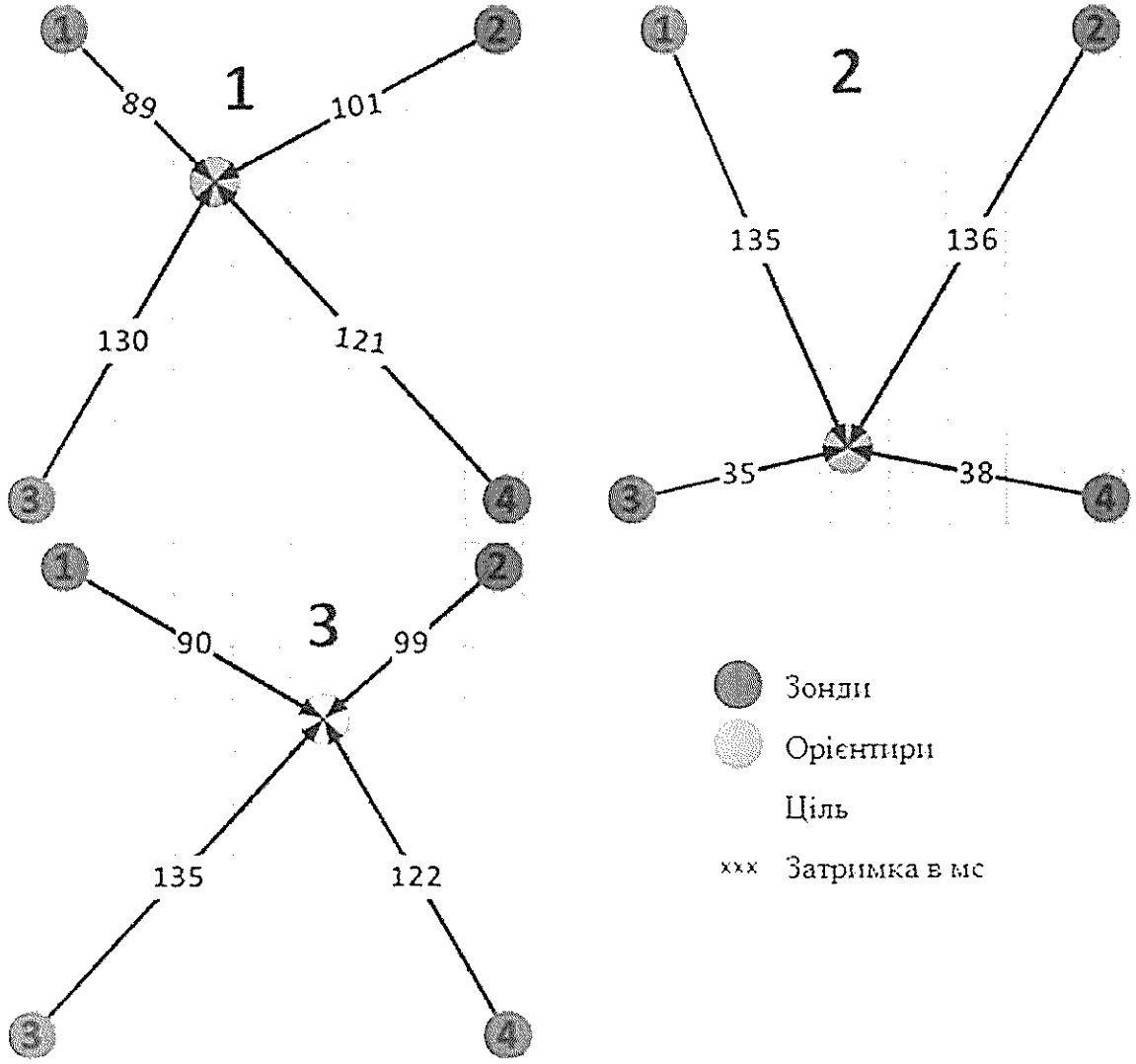
Геоінформаційне позиціонування об'єктів



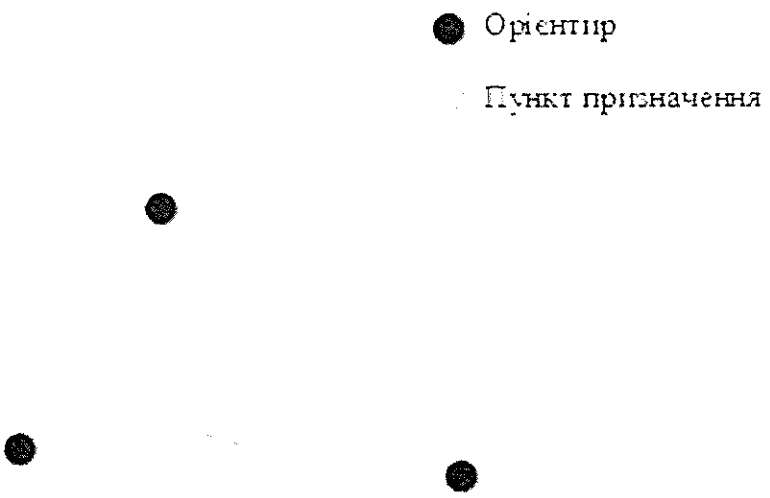
Пасивні методи геопозиціонування ТКО



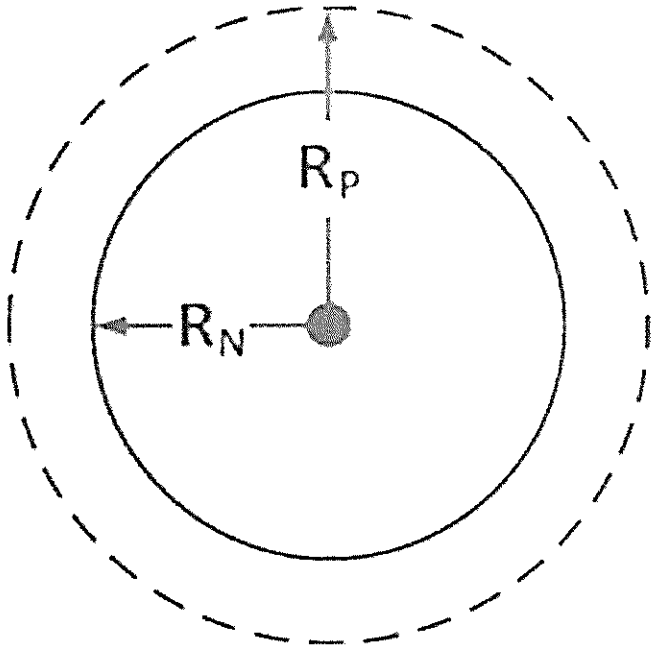
Активні методи геопозиціонування ТКО. Методи визначення цілі як єдиної точки



Активні методи геопозиціонування ТКО. Методи визначення площі розміщення цілі



Метод геоінформаційного позиціонування ТКО (МГТО)



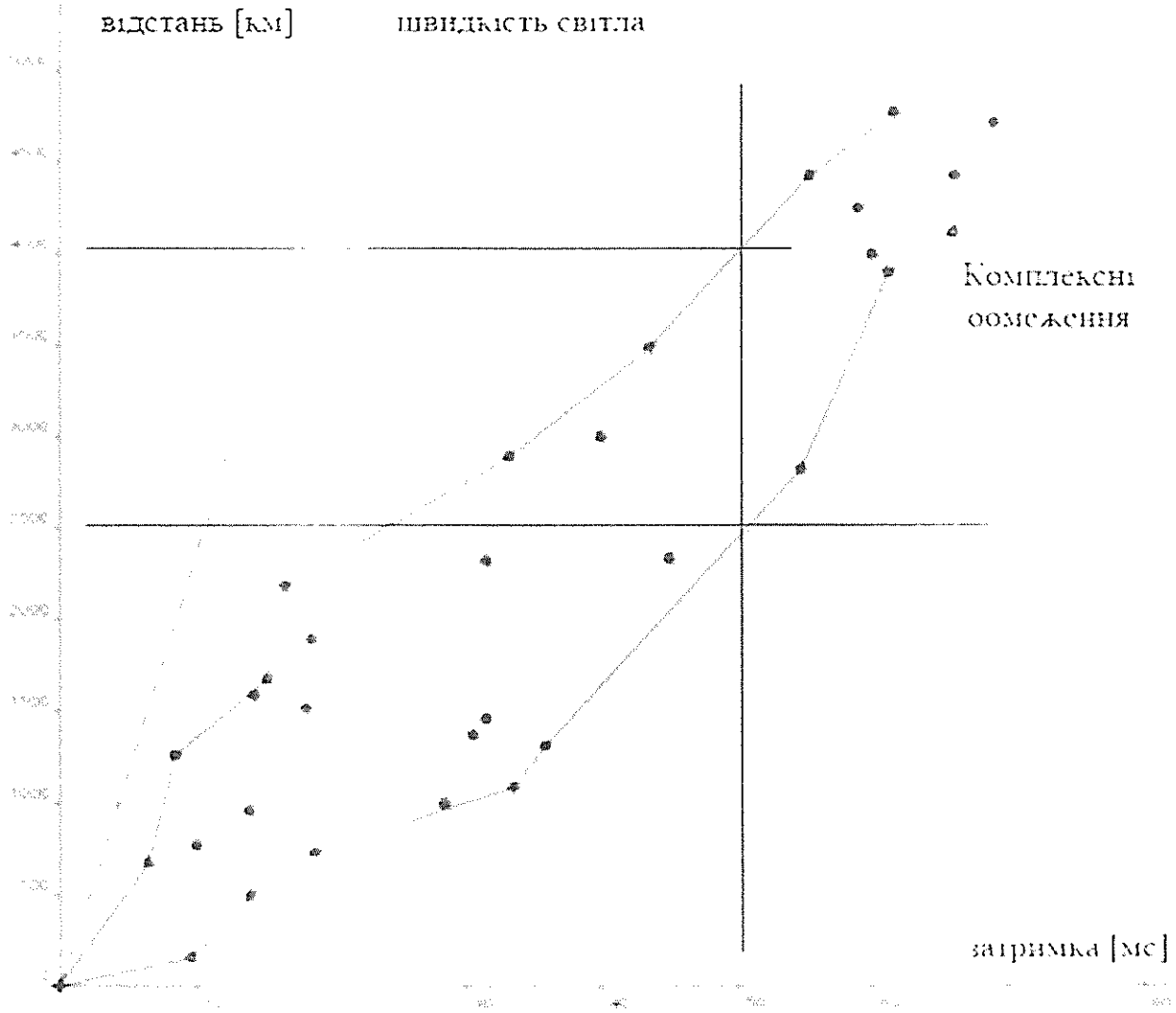
Орієнтир

Область, в якій знаходиться цільове положення

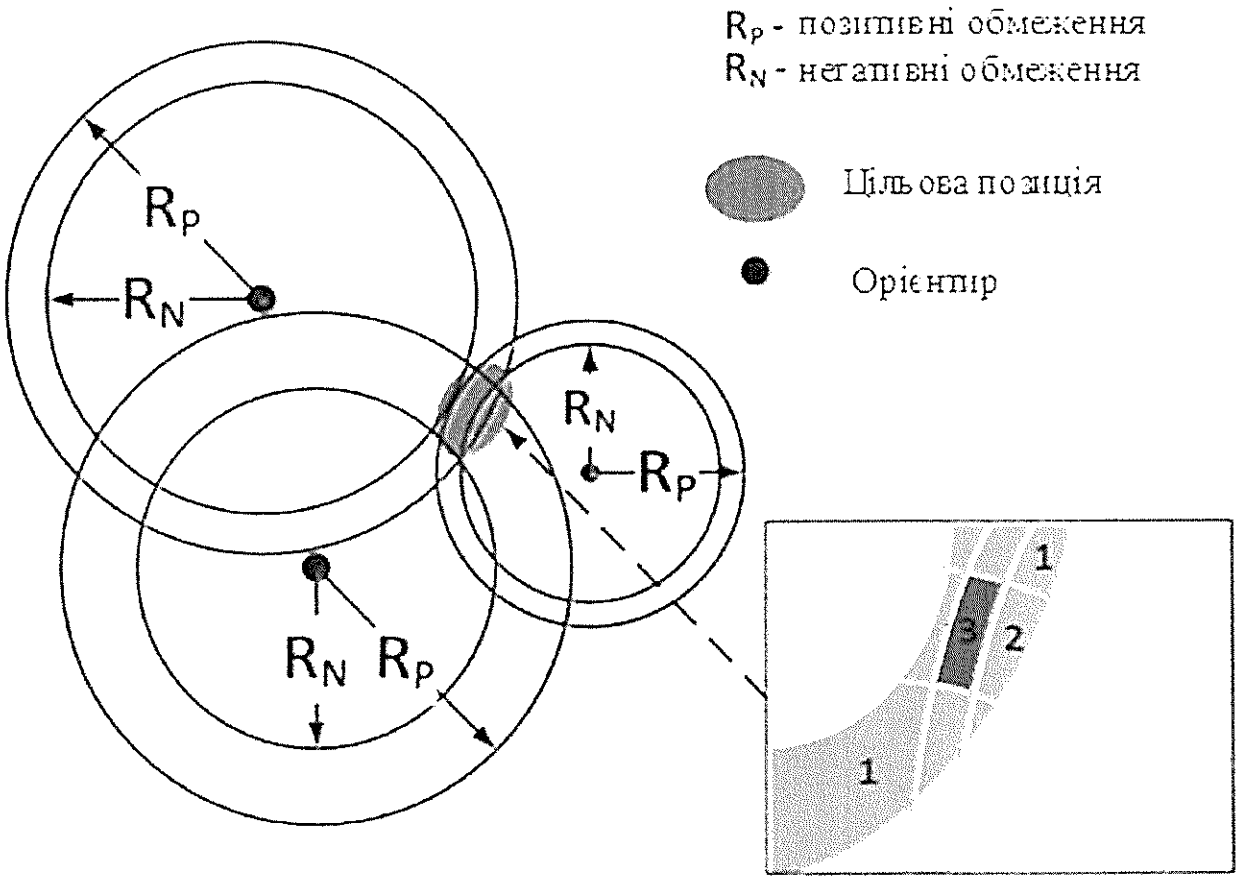
межа обмеження, отримана з позитивної інформації

межа обмеження, отримана з негативної інформації

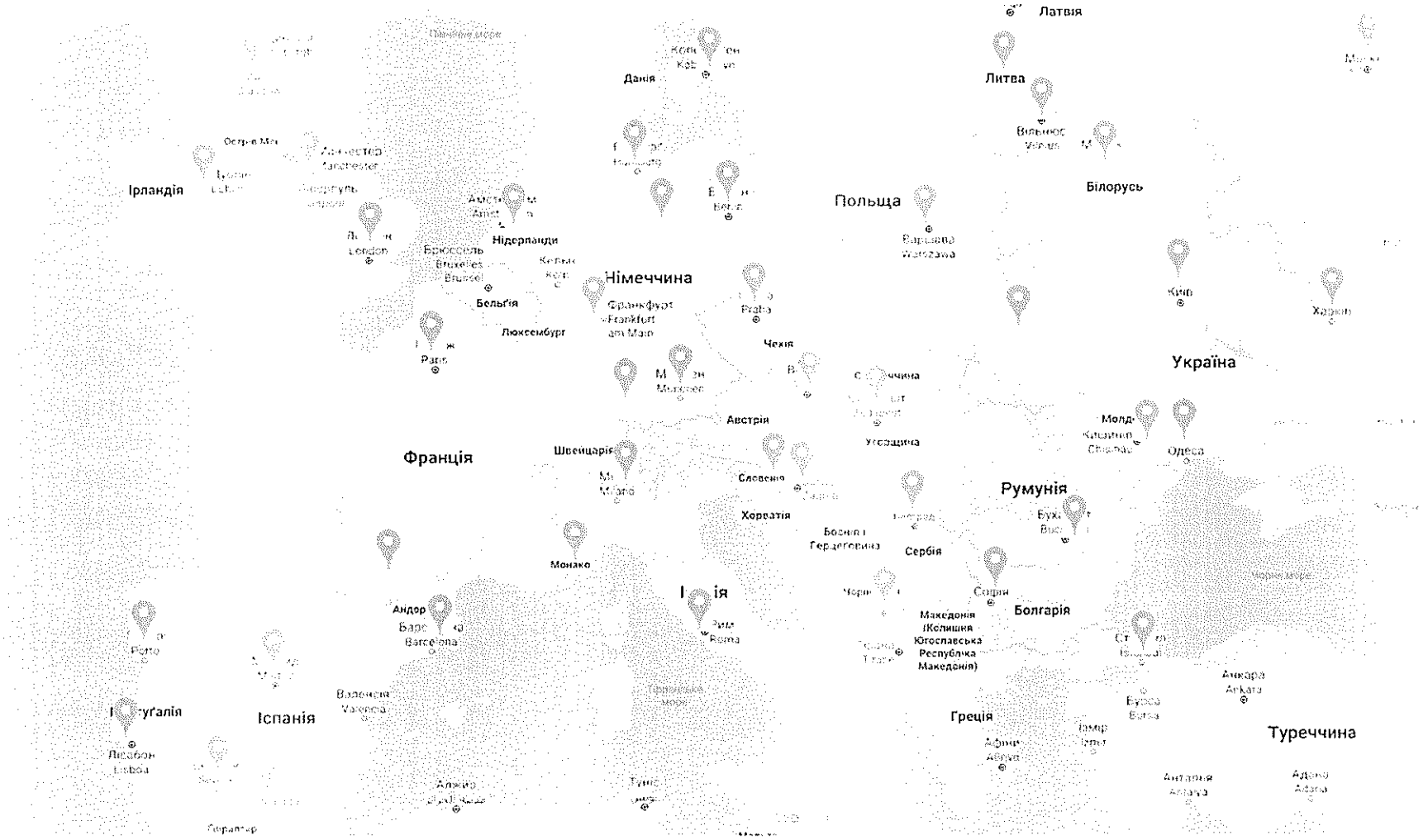
Залежність затримки від відстані



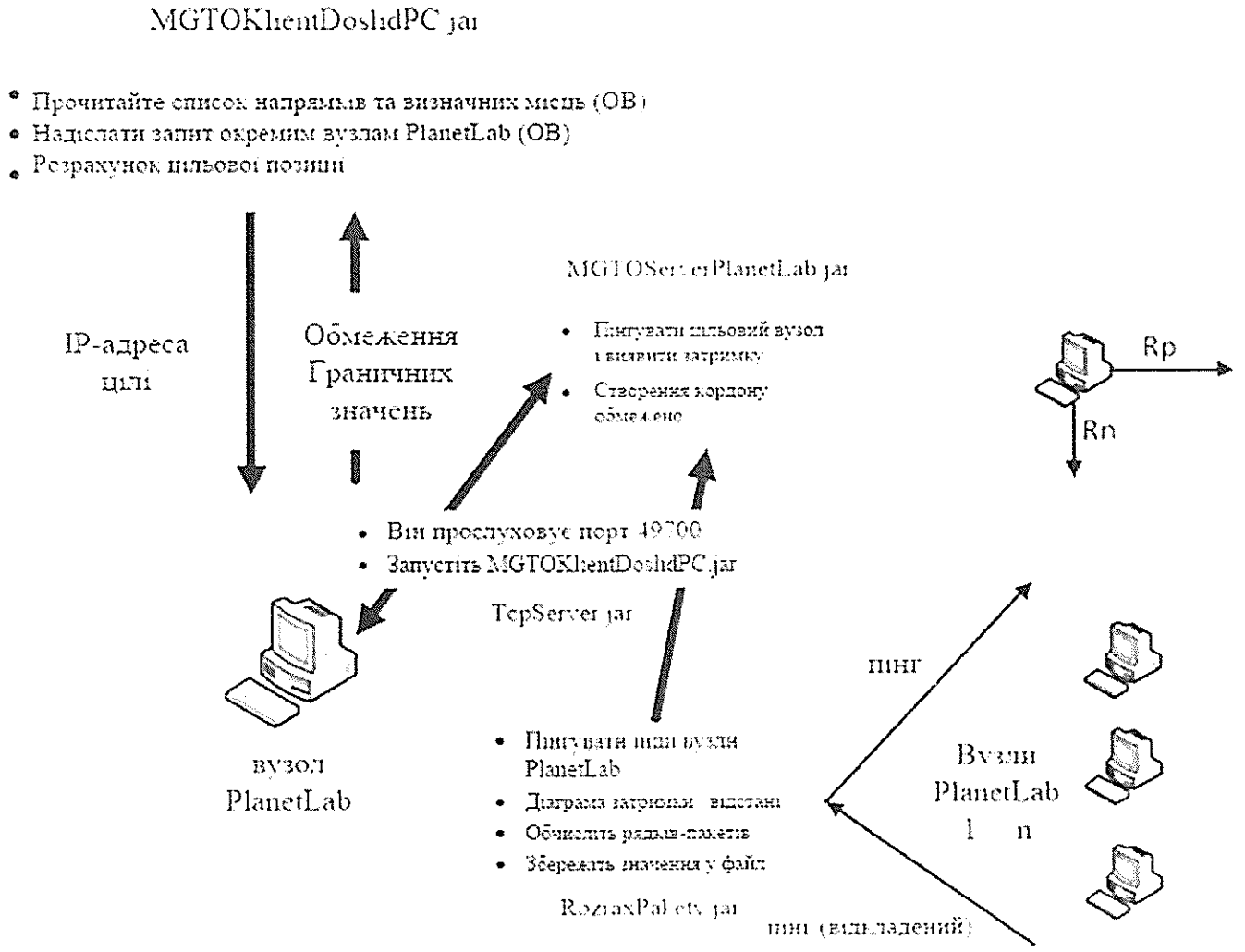
Визначення місця призначення



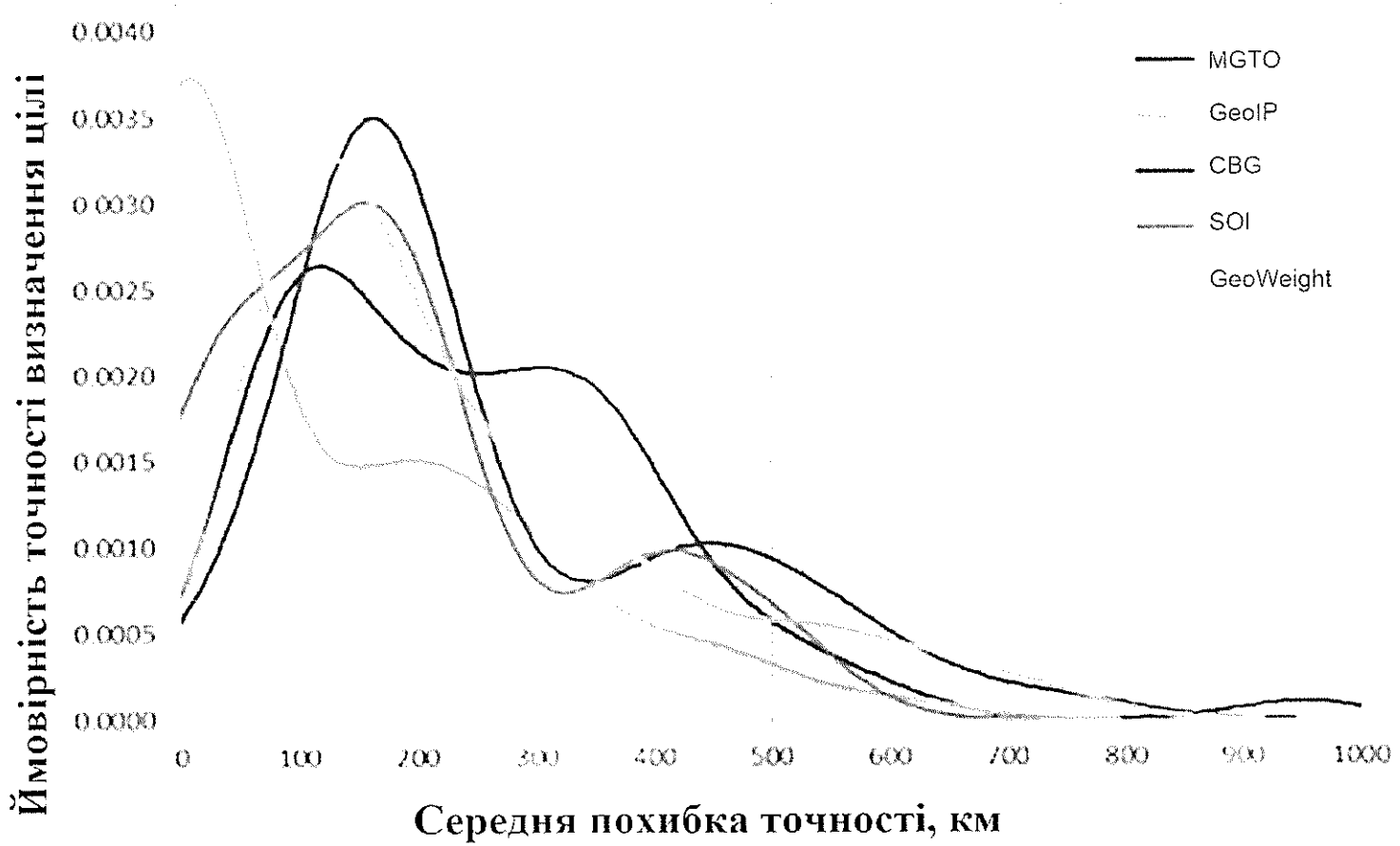
Практична реалізація МГТО. Створення орієнтирів за допомогою PlanetLab



Практична реалізація МГТО. Розробка комплексу програм



Графік ймоврності точності визначення ТКО



Порівняння методів CBG, SOI, МГТО, GeoWeight, GeoIP

Метод	Похибка точності, км	Швидкість локалізації однієї цілі, мс
МГТО	259,70	58,76
GeoIP	120,39	0,00
CBG	284,69	78,48
SOI	183,77	21,61
GeoWeight	239,85	349,48

Висновки

У магістерській роботі розглянуто сучасні методи геолокації та описано основні принципи їх роботи. Робота розділена на 2 частини - теоретичну та практичну.

Перша частина присвячена теоретичному опису цих методів та роз'ясненню основних понять і механізмів, що використовуються при визначенні фізичного розташування станції на основі її IP-адреси.

Більш вичерпна друга частина роботи стосується опису реалізації методу МГТО на мові програмування Java та його розгортання в експериментальній мережі PlanetLab. Важливою передумовою є створення так званого набору опорних точок і цілей, які необхідні для перевірки всього алгоритму методу МГТО.

Досягнуті результати в точності оцінки положення цілей методом МГТО та їх порівняння з результатами активних методів CBG, SOI та пасивних GeoIP представлені в кінці роботи.

Апробація результатів магістерської роботи

Основні результати магістерської роботи доповідалися на Всеукраїнській науково-практичній конференції «Telecommunication: problems and innovation», Київ: ДУІКТ, 20 грудня 2023 р.

Основні наукові результати магістерської роботи опубліковано в науковому журналі “Наукові записки ДУІКТ, – 2023, – №2”.

Протокол аналізу звіту подібності науковим керівником

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Максим ЛЬВОВ

Назва: Розробка методу геоінформаційного позиціонування телекомунікаційного обладнання

Координатор: Андрій ЛЕМЕШКО

Підрозділ: ННІТ

Коефіцієнт подібності 1 :11.1

Коефіцієнт подібності 2 :6.8

Тривога: 3

Після аналізу Звіту подібності констатую наступне:

☑ виявлені в роботі запозичення є сумлінними і не мають ознак плагіату. Тому робота визнається самостійною і допускається до захисту;

☐ виявлені в роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і самостійності її автора. Роботу направити на доопрацювання;

☐ виявлені в роботі запозичення є недобросовісними і мають ознаки плагіату або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень. У зв'язку з чим, робота не допускається до захисту.

....

Дата *Підпис Наукового керівника*