Хмельницький національний університет
Факультет інформаційних технлогій
Кафедра комп'ютерної інженерії та системного програмування

ДИПЛОМНА РОБОТА МАГІСТРА

Галузь знань 12 – Інформаційні технології

Спеціальність 123 –Комп'ютерна інженерія

на тему «Інтелектуалізована мобільна розподілена система для побудови блокчейну»

КвРКІ 016002.20.01.07 ПЗ

Виконав: студент 2 курсу, група КІ2м-20-1 _____ Гаврилюк Р.Л.
Підпис Ініціали, прізвище

Керівник кандидат техн. наук, доцент _____ Бобровнікова К.Ю
Науковий ступінь, вчене звання Підпис Ініціали, прізвище

До захисту допускаю:
Зав. кафедри КІСП, д.т.н., проф.
Т. О. Говорущенко
19 05 2022 р.

Хмельницький, 2022

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА СИСТЕМНОГО ПРОГРАМУВАННЯ

Освітній рівень МАГІСТР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма ОСВІТНЬО-НАУКОВА ПРОГРАМА «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ ТА ПРОГРАМУВАННЯ»

ЗАТВЕРДЖУЮ

Зав. кафедри Т.О.Говорущенко

"01" 09 2021 р.

ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ)

Гаврилюк Роман Леонідович

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Інтелектуалізована мобільна розподілена система для побудови блокчейну

Керівник проекту (роботи) Бобровнікова К.Ю., к.т.н., доц.

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 06.01.2021 р. № 1

2. Строк подання студентом проекту (роботи) на кафедру 06.05.2021 р.

3. Вихідні дані до проекту (роботи) Завдання на дипломне проектування

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

Аналіз відомих методів для підвищення ефективності побудови блокчейну в мобільних розподілених системах; Моделювання процесу побудови блокчейну з врахуванням часу побудови та енерговитрат; Метод побудови блокчейну з врахуванням часу побудови та енерговитрат; Апаратно-програмний рішення інтелектуалізованої мобільної розподіленої системи для побудови блокчейну

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень)

Розділ	Прізвище, консультанта	завдання видав	завдання прийн
Нормоконтроль	Лисенко С.М., професор кафедри КІСП		
Антиплагіат	Нічепорук А.О., доцент кафедри КІСП		

7. Дата видачі завдання «_06_»_____09_____2021 р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Приміт
1	Вибір напряму дослідження та узгодження тематики ДРМ з керівником	05.09.2021	виконан
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	05.10.2021	виконан
3	Робота над розділом 1 – аналіз відомих моделей, методів за темою; постановка задачі	05.11.2021	виконан
4	Робота над розділом 2 – розробка моделей для вирішення поставленої задачі	05.12.2021	виконан
5	Робота над науковою статтею та тезами	05.01.2022	виконан
6	Робота над розділом 3 – розробка методів для вирішення поставленої задачі	15.02.2022	виконан
7	Робота над розділом 4 – проектування та розробка системи захисту для вирішення поставленої задачі, експериментальна частина	05.04.2022	виконан
8	Оформлення пояснювальної записки згідно вимог	15.04.2022	виконан
9	Попередній захист ДРМ	18.04.2022	виконан
10	Захист ДРМ на засіданні ЕК	До 10.05.2022	

Студент _____ Підпис Р.Л. Гаврилюк Ініціали, прізвище

Керівник проекту (роботи) _____ Підпис К.Ю. Бобровнікова ініціали, прізвище

РЕФЕРАТ

Тема дипломної роботи: «Інтелектуалізована мобільна розподілена система для побудови блокчейну»

Автор роботи: Гаврилюк Роман Леонідович

Керівник роботи: Бобровнікова К.Ю.

Пояснювальна записка: 80 ст., 20 рис., 8 табл., 4 дод., 84 джерел.

ПЕРЕЛІК КЛЮЧОВИХ СЛІВ: мобільні розподілені системи, блокчейн, побудова блокчейну, PoW, енергоефективність, еволюційні алгоритми.

Об'єктом дослідження є процес побудови блокчейну в мобільних розподілених системах.

Предметом дослідження є моделі, методи та апаратно-програмні засоби для підвищення ефективності побудови блокчейну.

Метою дипломної роботи є розроблення інтелектуалізованої мобільної розподіленої системи для побудови блокчейну для підвищення ефективності побудови блокчейну в мобільних розподілених системах.

Для розв'язання поставлених задач використовувалися методи теорії мобільних бездротових мереж, побудови блокчейну, теорії множин, еволюційні алгоритми.

Наукова новизна одержаних результатів полягає в наступному:

- Удосконалено модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, що заснована на удосконалених моделях: завантаження даних в мобільні вузли, які беруть участь в побудові блокчейну в розподілених системах; вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах; поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах. Удосконалена модель, на відміну від відомих моделей, надає можливість обчислювати ефективність побудови блокчейну для окремих популяцій мобільних вузлів.

- Розроблено інтелектуалізований метод побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, який заснований на удосконаленій моделі. Метод грунтується на еволюційних алгоритмах та, на відміну від відомих підходів, надає можливість на основі наявної множини мобільних вузлів створювати максимально ефективну популяцію мобільних вузлів, які беруть участь у процесі побудови блокчейну, базуючись на часі створення блоків і енерговитратах. Розроблений метод надає можливість підвищити ефективність побудови блокчейну, в порівнянні з відомими методами для побудови блокчейну.

- Побудовано інтелектуалізовану мобільну розподілену систему для побудови блокчейну, яка грунтується на розробленому методі, та надає можливість підвищити ефективність побудови блокчейну, в порівнянні з відомими методами для побудови блокчейну.

Практична цінність полягає у розроблені інтелектуалізованої мобільної розподіленої системи для побудови блокчейну, яка надає можливість підвищити ефективність побудови блокчейну на 12-20%, в порівнянні з відомими підходами.

Теоретичні та практичні результати роботи впроваджено при виконанні студентських науково-дослідних робіт, які виконувались в Хмельницькому національному університеті.

Дослідження, представлені у кваліфікаційній роботі, проводились в рамках держбюджетної НДР Хмельницького національного університету 1Б-2021 «Самоорганізована розподілена система виявлення зловмисного програмного забезпечення в комп'ютерних мережах» (ДР No 0121U109936) 2021-2022 рр.

За темою дипломної роботи опубліковано статтю на тему «The Approach for IoT Malware Detection Based on Opcodes Sequences Pattern Mining» в матеріалах конференції 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021, що індексується в наукометричній базі Scopus, а також опубліковано тези у Всеукраїнській науково-практичній конференції Актуальні Проблеми Комп'ютерних Наук (АПКН-2021). Було взято участь у Всеукраїнській науково-практичній конференції Актуальні Проблеми Комп'ютерних Наук.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

PoW – proof-of-work

p2p – з’єднання типа точка-точка (peer-to-peer)

IoT – Інтернет речей (Internet of things)

PoS – proof-of-stake

ПЗ – програмне забезпечення

ЕА – еволюційний аглгоритм

ДЕ – диференційна еволюція

ВСТУП

На сьогоднішній день блокчейн-технологія, і її реалізації [1-5] стали дуже актуальними в сфері інформаційних технологій. Але все ще існує проблема з енергоефективністю процесу побудови блокчейну, особливо для PoW [6] рішень. Крім того, існує явна проблема з використанням IoT і мобільних пристроїв [7-12] для виконання процесу побудови блокчейну, в зв'язку з їх обмеженими обчислювальними ресурсами. Хоча саме, адаптація блокчейн-технології під побудову мобільними пристроями, або надання мобільних пристроям умов для ефективної побудови блокчейну, могла б вирішити проблему енергоефективності при процесі побудови блокчейну.

Тому метою дипломної роботи є розроблення інтелектуалізованої мобільної розподіленої системи для побудови блокчейну для підвищення ефективності побудови блокчейну в мобільних розподілених системах.

Задачі дослідження можуть бути сформульовані наступним чином:

1. Провести огляд відомих рішень для побудови блокчейну в мобільних розподілених системах.

2. Провести огляд відомих рішень для підвищення ефективності побудови блокчейну.

3. Удосконалити модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат.

4. Розробити інтелектуалізований метод для підвищення ефективності побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат.

5. На основі розробленого методу створити апаратно-програмне рішення мобільної розподіленої системи, застосування якого надасть можливість підвищити ефективність побудови блокчейну, в порівнянні з відомими аналогами.

Об'єктом дослідження є процес побудови блокчейну в мобільних розподілених системах.

Предметом дослідження є моделі, методи та апаратно-програмні засоби для підвищення ефективності побудови блокчейну.

Наукова новизна одержаних результатів полягає в наступному:

1. Удосконалено модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, що заснована на удосконалених моделях: завантаження даних в мобільні вузли, які беруть участь в побудові блокчейну в розподілених системах; вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах; поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах. Удосконалена модель, на відміну від відомих моделей, надає можливість обчислювати ефективність побудови блокчейну для окремих популяцій мобільних вузлів.

2. Розроблено інтелектуалізований метод побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, який заснований на удосконаленій моделі. Метод грунтується на еволюційних алгоритмах та, на відміну від відомих підходів, надає можливість на основі наявної множини мобільних вузлів створювати максимально ефективну популяцію мобільних вузлів, які беруть участь у процесі побудови блокчейну, базуючись на часі створення блоків і енерговитратах. Розроблений метод надає можливість підвищити ефективність побудови блокчейну, в порівнянні з відомими методами для побудови блокчейну.

3. Побудовано інтелектуалізовану мобільну розподілену систему для побудови блокчейну, яка грунтується на розробленому методі, та надає можливість підвищити ефективність побудови блокчейну, в порівнянні з відомими методами для побудови блокчейну.

Практична цінність полягає у розроблені інтелектуалізованої мобільної розподіленої системи для побудови блокчейну, яка надає можливість підвищити ефективність побудови блокчейну на 12-20%, в порівнянні з відомими підходами.

За темою дипломної роботи опубліковано статтю на тему «The Approach for IoT Malware Detection Based on Opcodes Sequences Pattern Mining» в матеріалах конференції 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021, що індексуються в наукометричній базі Scopus [13], а також опубліковано тези у Всеукраїнській науково-практичній конференції Актуальні Проблеми Комп’ютерних Наук (АПКН-2021) [14]. Було взято участь у Всеукраїнській науково-практичній конференції Актуальні Проблеми Комп’ютерних Наук.

1 АНАЛІЗ ВІДОМИХ ТЕХНОЛОГІЙ ДЛЯ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ ПОБУДОВИ БЛОКЧЕЙНУ

1.1 Блокчейн, як технологія

Блокчейн [15, 16] – це технологія, яка являєть по суті повністю розподіленою базою даних [17], в якій кожен вузол блокчейн-мережі, містить копію даних [18]. В криптовалютах, наприклад, блокчейн використовується для забезпечення захищеного та децентралізованого збереження транзакцій [19].

Блокчейн – це технологія, яка дозволяє зберігати інформацію відносно безпечно, і дає можливість переглядати всю історію блоків, а також гарантує, що ці блоки не буде змінено після внесення у блокчейн [20-22].

На рис. 1.1 можна побачити схематичне представлення блокчейну [23], де видно, що, по суті, блокчейн це ланцюг блоків, де в кожному наступному блоці міститься хеш на попередній блок, що в свою чергу дозволяє відслідковувати всю історію цього ланцюжка [24]. Кожен окремий блок складається з заголовку, в якому містяться хеш попереднього блоку, основна інформація про блок, така як, версія, об’єм, та з даних, в яких містяться самі дані [25-26]. Дані можуть бути будь-якими, наприклад, транзакції, як в криптовалютних блокчейнах, або запись чи картинки, єдине що потрібно врахувати дані повинні бути однотипними, і не перевищувати об’єм для них відведений [27].

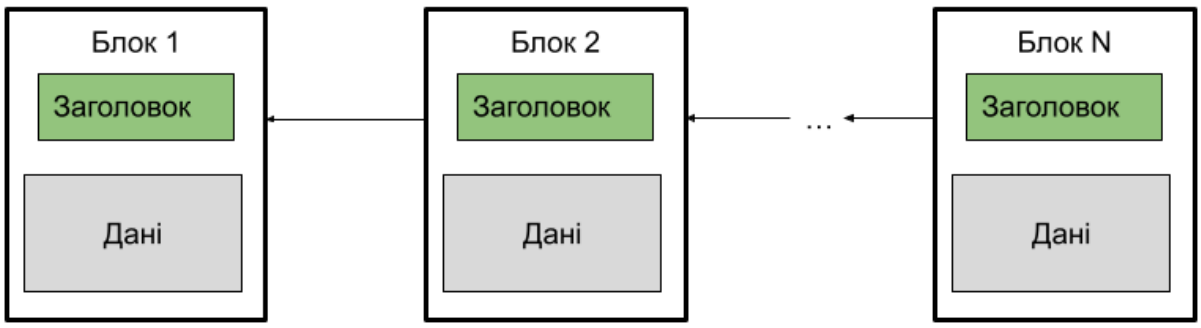


Рисунок 1.1 – Схематичне представлення структури блокчейну

1.2 Відомі технології для побудови блокчейну

1.2.1 Підходи побудови блокчейну в галузі криптовалют

Криптовалютні реалізації біткойну [28-30], зазвичай містять в собі транзакції, на рисунку 1.2 видно, як проходять типові транзакції в блокчейн мережах. Спочатку створюється запис про транзакцію, після чого ця транзакція має бути внесена в блок, і тільки після закриття блоку, тобто його заповнення транзакціями, і внесення блоку в блокчейн, транзакція може вважатись виконаною [31-36]. Деталі можуть відрізнятись в залежності від реалізації, але загалом все виглядає так.

Рисунок 1.2 – Алгоритм виконання транзакцій

Блокчейн, найбільше відомий завдяки вдалим криптовалютним реалізаціям, котрі “вистрілили”, по типу Bitcoin, Ethereum, Litecoin і так далі. Розглянемо методи для побудови блокчейну у цих трьох найпопулярніших криптовалютах.

Як видно з таблиці 1.1, блокчейн займає доволі багато місця, що являється можливою проблемою. Також, можна звернути увагу на різний час створення нового блоку, а також на абсолютно різний розмір блоків, даних блокчейнів, це зв’язано з методами побудови блокчейну і механізмами досягнення консенсусу у них [37].

Таблиця 1.1 – Інформація по криптовалютам, станом на січень 2022.

Назва	К-кість валюти	К-кість блоків	Час створення нового блоку	Розмір блокчейну
Bitcoin [28]	18,936,390 BTC	719,875	10m 40s	448.15 GB
Ethereum [29]	119,263,003 ETH	14,054,560	13.4s	345.17 GB
Litecoin [30]	69,468,608 LTC	2,197,642	2m 37s	72.37 GB

1.2.2 Підходи побудови блокчейну в інших галузях

Загалом, хоч блокчейн і асоціюється з криптовалютами, – це далеко не єдина сфера в якій його можна застосовувати. На сьогодні існує багато сфер використання блокчейну, не пов’язаних з криптовалютами: медицина та охорона здоров’я, Інтернет речей (IoT), цифрова реклама, страхування тощо [38-47]. В цієї технології є великий потенціал, і за умови грамотної реалізації вона може принести значну користь як в сферах бізнесу, так і в державних сферах.

Наприклад, в проекті e-Estonia [48], який тісно співпрацює з естонською владою, використовується блокчейн-технологія, для забезпечення безпеки та гарантії незмінності даних. Сам проект, являє собою серію рішень, які дозволяють взаємодіяти з державними структурами в інтернеті.

Британський стартап Moni [49], який тісно співпрацює з фінським урядом, намагається вирішити проблему з біженцями, з участю блокчейну, шляхом, видачі біженцям допомоги, не готівкою, а грошима на спеціальну карту, що дозволяє

відслідковувати їх розташування, і покупки, що в свою чергу дозволяє вирішити важливу проблему для Євросоюзу.

Варто зазначити, що блокчейн в подібних рішеннях, використовується лише в якості допоміжної технології, для збереження інформації з використанням розподілених систем, а не в якості основи для рішення. Тобто, вони могли використати і інші технології, але використали саме блокчейн, через переваги, які він надає.

Незважаючи, на всі переваги, які може надавати блокчейн технологія, у неї, як і у будь-якого іншого рішення, є недоліки.

Перш за все, це складність самої технології, для розробників. Китайська академія інформаційних та комунікаційних технологій (CAICT) в грудні, 2018, опублікувала дослідження в якому вказано, що близько 92% блокчейн-проектів, являються провальними [50], що порівняно з іншими проектами в індустрії інформаційних технологій, є дуже поганим результатом.

До очевидних недоліків можна віднести необхідність в обчислювальної потужності, для підтримки роботи блокчейну, та побудови нових блоків. З цього також випливає проблема енергоефективності, оскільки для постійної роботи системи потрібна велика кількість електроенергії.

Також існує проблема з постійним збільшенням розміру блокчейну, і об’єму дискового простору, який він займає. На даний момент, ця проблема не є критичною, оскільки завжди можна розширити дисковий простір, але можливо в майбутньому, це призведе до неприємних наслідків.

1.3 Аналіз відомих рішень для підвищення ефективності побудови блокчейну

Існує багато рішень для побудови блокчейну, котрі поліпшують ефективність роботи блокчейн-мережі в залежності від поставленої задачі вони або збільшують

кількість транзакцій в секунду, або зменшують часові затрати на деякі з кроків побудови блокчейну. У цьому розділі будуть розглянуті деякі з них.

У роботі [51] оглядаються деякі з методів для побудови блокчейну, а також пропонуються деякі, можливі, рішення для збільшення маштабованості блокчейну, а саме для поліпшення ефективності побудови блокчейну, а саме збільшення кількості транзакцій в секунду, і зменшення часу необхідного на підтвердження дій.

Автори роботи [52] пропонують перейти з proof-of-work(PoW) на proof-of-stake(PoS), що в свою чергу значно зменшить час необхідний на побудову нового блоку, а також значно зменшить енерговитрати. Це в свою чергу призведе до того, що подібний блокчейн буде менш вразливий до тих атак, до яких вразливі біткойн-подібні блокчейн рішення, які базуються на PoW.

У [53] описується алгоритм selfish mining. Він базується на тому, щоб групка майнерів умисно саботувала роботу блокчейн-мережі, шляхом її уповільнення, з метою отримати винагороду самим. Даний метод також розглядається у роботах [54-56]. Звісно цей метод не збільшує ефективність побудови блокчейну, але все ж його потрібно враховувати при побудові власної блокчейн мережі.

У [57] пропонується відійти від PoW алгоритму для досягнення консенсусу, а використати натомість алгоритм Practical byzantine fault tolerance [58]. Подібний підхід значно збільшує захищеність системи, а також зменшує часові затримки необхідні для досягнення консенсусу і виконання транзакції.

В роботах [59-60] для того ж відходу від PoW, пропонується використовувати алгоритм описаний у [61] – так званий sharding. Це дозволяє отримати механізм для досягнення консенсусу, який не впливає на децентралізованість і захищеність системи, але збільшує ефективність таких ключових компонентів як ефективність пропагації, збереження блоків, час виконання транзакцій.

Одним зі способів покращення ефективності побудови блокчейну, будь то швидкість майнінгу, або кількість транзакцій в секунду, являється зміна розміру блоку.

Наприклад, для поліпшення ефективності побудови блокчейну [62] пропонує використовувати підхід з використанням “відокремленого свідку” (англ. Segregated witness). Він полягає в тому щоб відокремити ключ підпису від основної транзакції, таким чином, збільшивши можливості для блокчейну і кількість транзакцій у блоці, тобто кількість транзакцій в секунду. Оскільки велику частину блоку займають підписи і хеші підписів, приріст ефективності буде вагомим. До недоліків можна віднести те що ключі будуть зберігатись окремо, і складніше буде відслідковувати зміни.

Збільшення розміру блоку також являється одним з можливих способів для покращення ефективності побудови блокчейну [63], зокрема подібний крок значно збільшить кількість транзакцій в секунду, але в той же час деякі дослідження [64] вказують на те що подібне рішення може призвести до ряду інших проблем, наприклад, на кроці пропагації.

Також, одним з варіантів оптимізації розміру блоку являється стиснення надлишкових даних [65]. Ідея полягає в тому що необов’язково повністю зберігати інформацію блоків, особливо якщо вона являється надлишковою, або до неї не потрібно часто звертатись. Подібне рішення значно зменшить розмір блокчейну, і не призведе до значних змін в ефективності роботи блокчейн-мережі.

Іншим способом, який може покращити ефективність побудови блокчейну є використання інших способів досягнення консенсусу. Як, наприклад, вже описані вище [57, 59, 60].

Механізм для досягнення консенсусу PoS допомагає запобігти тій великій кількості обчислень, яка притаманна для класичного PoW механізму. Замість того щоб вимагати обчислювальні ресурси від вузлів для генерації нових блоків, вузли в PoS шляхом голосування, за власним вкладом в блокчейн, виконують задачу досягнення консенсусу, таким чином зменшуючи час на підтвердження транзакцій. Основна ідея полягає в тому що вузли які брали участь в побудові блокчейну, з меншою вірогідністю будуть йому шкодити, або виконувати дії які являються зловмисними по відношенню до інших учасників мережі. Але, оскільки PoS не

передбачає додаткових верифікацій шляхом складних комп'ютерних обчислень, то для забезпечення безпеки блокчейну, необхідно вводити додаткові захисні протоколи, що являється складною задачею.

Наприклад [66], описує створений протокол Snow White для PoS, який передбачає забезпечення захисту шляхом, не просто розділу вузлів на довірених, і не довірених, а також шляхом внесення деякого випадкового значення в процес досягнення консенсусу. У вже згаданому [52] пропонується обирати довіренні вузли на певний проміжок часу(епоху), і закладати нове значення у цій епосі, для вибору нових довірених вузлів у наступній.

Існує також механізм для досягнення консенсусу Delegated proof-of-stake(DPoS). Вперше був згаданий у роботі [67]. Ідея полягає у тому що б шляхом голосування обрати 21 вузол, які будуть брати участь у генерації наступного блоку, де вага голосу визначається вкладом в блокчейн.

Також, можна змінити саму структуру PoW, як це зроблено, наприклад, в [68]. Базова суть залишається незмінною, все ще використовуються комп'ютерні обчислення для додаткових перевірок і досягнення консенсусу, тобто все ще потрібні майнери, але сам блок формується, обраним випадковим чином, лідером, котрий обирається на певний проміжок часу. Таким чином, зменшується час транзакцій, і збільшується загальна надійність блокчейну.

У роботі [69] на основі досліджень було створено таблицю подібну до таблиці 1.2, яка описує найбільш відомі методи для досягнення консенсусу в блокчейн-мережах, а також визначає їх переваги і недоліки відносно одна-одної.

Таблиця 1.2 – Опис механізмів для досягнення консенсусу

Алгоритм	PoW	PoS	DPoS
Вклад	Обчислювальні ресурси	Відсоток блокчейну	Голос за відсотком в блокчейні
Вразливості	Концентрація обчислювальних ресурсів	Недостача активних вузлів	Знищення свідку

Кінець таблиці 1.1 – Опис механізмів для досягнення консенсусу

Алгоритм	PoW	PoS	DPoS
Потреба в ресурсах	максимальна	Менше ніж у PoW, але більше ніж у DPOS	мінмальна
Середній час генерації нового блоку	10 хвилин	64 секунди	3 секунди
Реалізації	bitcoin	peercoin	bitshare
Чесність	Відносно висока	Відносно низька	Низька

Варто наголосити, що вказані в таблиці 1.2 дані являються типовими для визначених механізмів досягнення консенсусу в блокчейн-мережах, але все ще можуть відрізнятись в залежності від конкретної реалізації. Це не означає що ці дані являються ненадійними, але означає що в межах обраного консенсусу все ще можуть бути відмінності між певними рішеннями на його основі.

Також, на основі даних зібраних в згадаій вище [68], було створено таблицю 1.3, яка описує переваги і недоліки менш відомих механізмів для досягнення консенсусу в блокчейн-мережах.

Таблиця 1.3 – Переваги і недоліки інших механізмів для досягнення консенсусу

Алгоритм	Переваги	Недоліки
Raft[69]	Практичний і легкий для розуміння	Низька безпека
RPCA[70]	Низькі затримки	Вузька область для можливих реалізацій

Кінець таблиці 1.3 – Переваги і недоліки інших механізмів для досягнення консенсусу

PoB[71]	Підвищена ефективність	Проблеми з безпекою та практичністю
PoA[72]	Підвищена чесність і ефективність	Низька безпека
Tendermint[73]	Стійкість до певного роду атак	Низька швидкодія
SPC[74]	Безпечний консенсус	Високі часові затримки
Elastico[75]	Збільшення ефективності PBFT	Вузька область для можливих реалізацій
HoneyBadger[76]	Збільшена швидкодія	Зменшена маштабованість
The 2-hop consensus[77]	Стійкість проти атак 51%	Більша централізованість
dbFT[78]	Збільшена пропускна здатність і безпека	Висока складність

Вибір підходящого механізму для досягнення консенсусу являється дуже важливим при створенні власного блокчейну, і може запобігти багатьом проблемам пов'язаних з його експлуатацією. Звісно є можливість переходу між різними механізмами, але це являється часозатратним процесом, хоча і може знадобитись для вирішення певного роду криз в блокчейн-мережі.

Описані вище способи орієнтуються на зміну самої структури блокчейну, але це не завжди являється доцільним, оскільки подібне рішення може призвести до ряду інших проблем. Може скластись так що більш доречним буде оптимізувати вже існуючу блокчейн-мережу, для збільшення ефективності побудови блокчейну.

Для такої задачі може підійти розробка стратегії для покращення ефективності майнінгу. У роботі [79] пропонується RL алгоритм (англ. reinforcement learning), який шляхом машинного навчання підбирає оптимальну стратегію для майнінгу в біткойн-подібній блокчейн-мережі. Автори вказують, що алгоритм розроблений ними дозволяє досягти такої ж ефективності, що і альтернативи, але не маючи знань про повну структуру мережі, що не притаманно аналогам. Тому, даний алгоритм є дуже корисним для динамічних блокчейн-мереж.

1.4 Постановка задачі

Дослідження джерел показало, що проблема збільшення ефективності побудови блокчейну є надзвичайно актуальною, оскільки в деяких випадках процес побудови блокчейну може бути доволі енерговитратним. Отже, метою роботи є підвищення ефективності побудови блокчейну в мобільних розподілених системах шляхом розроблення інтелектуального методу і програмного засобу для побудови блокчейну.

Для досягнення мети роботи, необхідно вирішити наступні завдання:

1) провести огляд відомих рішень для побудови блокчейну в мобільних розподілених системах;

2) провести огляд відомих рішень для підвищення ефективності побудови блокчейну;

3) удосконалити модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат;

4) розробити інтелектуалізований метод для підвищення ефективності побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат;

5) на основі розробленого методу створити апаратно-програмне рішення мобільної розподіленої системи, застосування якого надасть можливість підвищити ефективність побудови блокчейну, в порівнянні з відомими аналогами.

1.5 Висновок

Наш світ бурхливо розвивається, і постійно з’являються технології, достойні уваги, і блокчейн, одна з них. Активний розвиток технології почався відносно нещодавно, тому є широке поле для можливих реалізацій, та ідей, які могли б, покращити загальну ефективність роботи блокчейну.

В першому розділі було зроблено аналіз відомих технологій побудови блокчейну в таких сферах як криптовалюта, медична та інші.

На основі проведеного аналізу було зроблено висновок, що, незважаючи на цікавість блокчейну, як технології, яка дозволяє будувати безпечну, розподілену систему, в якій майже неможливо змінювати блоки, в ній ще можна багато моментів доопрацьовувати. Наприклад, дуже обмежена кількість транзакцій в секунду, щоправда це здебільшого стосується криптовалютних рішень, або величезні витрати електроенергії для підтримання роботоздатності блокчейну.

Саме тому, існує необхідність в розробленні нових методів та систем для підвищення ефективності побудови блокчейну, які би враховували час побудови блоків і енерговитрати.

2 МОДЕЛЬ ПОБУДОВИ БЛОКЧЕЙНУ В МОБІЛЬНИХ РОЗПОДІЛЕНИХ СИСТЕМАХ З УРАХУВАННЯМ ЧАСУ ПОБУДОВИ БЛОКІВ ТА ЕНЕРГОВИТРАТ

З метою підвищення ефективності побудови блокчейну в мобільних розподілених системах необхідно удосконалити модель, яка повинна описувати побудову блокчейну, а також, враховувати час побудови блоків та енерговитрати. Модель базується на вирішенні мобільними вузлами PoW задачі, з метою побудови блокчейну.

Модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат повинна передбачати використання наступних моделей:

- модель завантаження даних в мобільні вузли, які беруть участь в побудові блокчейну в розподілених системах;
- модель вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах;
- модель поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах.

2.1 Модель завантаження даних в мобільні вузли, які беруть участь в побудові блокчейну в розподілених системах

При виконанні PoW задачі першим кроком являється завантаження даних. Дана модель передбачає формування завдання, і його відправлення мобільним вузлам, які беруть участь у процесі. Даний крок повинен передбачати специфіку роботи мобільних мереж, і можливих затримок в них. Модель для завантаження даних, які беруть участь в побудові блокчейну в мобільних розподілених системах передбачає використання і обробку наступних вхідних даних:

$$\rho = \langle N',\ W,\ p_i,\ \sigma,\ H_i,\ D \rangle, \tag{2.1}$$

де ρ – вхідні дані моделі завантаження даних в мобільні вузли, які беруть участь в побудові блокчейну в розподілених системах;

N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну;

W – пропускна здатність серверу;

p_i – потужність передачі даних мобільним пристроєм;

H_i – статус бездротового каналу між мобільним вузлом і сервером;

σ – похибка пов'язана з можливими фоновими шумами в мережі;

D – розмір блоку завдання, який передається мобільному вузлу, що бере участь у процесі побудови блокчейну.

Перш за все, модель передбачає безпосереднє завантаження даних. Одночасно вузли-учасники PoW процесу виконання задач повинні завантажити свої задачі з серверу. Для вирахування швидкості передачі даних в мобільних розподілених системах модель використовує наступну формулу:

$$V_i^t = W \log_2 \left(1 + \frac{p_i H_i}{\sigma + a(i)}\right), i\ \in N', \tag{2.2}$$

де V_i^t – швидкість передачі даних мобільному вузлу, який бере участь у процесі побудови блокчейну;

σ – похибка пов'язана з можливими фоновими шумами в мережі;

α(i) – втручання інших учасників на передачу даних;

W – пропускна здатність серверу;

p_i – потужність передачі даних мобільним пристроєм;

H_i – статус бездротового каналу між мобільним вузлом і сервером;

N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну.

Швидкість завантаження даних мобільними вузлами необхідна для виконання обрахунків можливих часових затрат і енерговитрат, які мали місце при завантаженні даних з серверу. Також, це значення описує те з якою швидкістю

обмінюється даними мобільний вузол і сервер. Значення швидкості передачі даних унікальне для кожного учасника, крім того, повинні бути проведені переобчислення при зміні множини мобільних вузлів учасників мережі.

Для обрахування втручання інших учасників на роботу мережі врахувати те скільки мережевого трафіку займає кожен учасник майнінгового процесу і додати ці значення. Для цього було створено наступну формулу:

$$a(i) = S - p_i H_i, \tag{2.3}$$

$$S = \sum_{j \in N'} p_i H_i, \tag{2.4}$$

де α(i) – втручання інших учасників на передачу даних;

S – сума втручання всіх учасників які беруть участь у процесі побудови блокчейну на мобільну мережу;

p_i – потужність передачі даних мобільним пристроєм;

H_i – статус бездротового каналу між мобільним вузлом і сервером;

N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну.

При обчислені формули 2.3, S перераховується за формулою 2.4 кожен раз як змінюється множина мобільних вузлів, які беруть участь у побудові блокчейну, що потрібно враховувати. Оскільки множина мобільних вузлів, які беруть участь у побудові блокчейну буде змінюватись рідко, кількість виконання повторних обчислень буде відносно малою, а також збільшить загальну точність моделі при виконанні обчислень.

Після того як була передбачена швидкість передачі даних в мобільних розподілених системах (формула 2.1), модель повинна передбачити обрахунок таких значень моделі, як час передачі даних для мобільних вузлів, які беруть участь у побудові блокчейну, а також енерговитрати мобільних вузлів, які беруть участь у побудові блокчейну, під час процесу передачі даних. Для обрахунку описаних величин використовуються наступні формули моделі:

$$T_i^t = \frac{D}{V_i^t},\ i\ \in N', \tag{2.5}$$

$$E_i^t = p_i T_i^t,\ i\ \in N', \tag{2.6}$$

де T_i^t – час передачі блоку завдання мобільному вузлу, який бере участь у процесі побудови блокчейну;

E_i^t – загальні енерговитрати на процес передачі блоку завдання мобільному вузлу, який бере участь у процесі побудови блокчейну;

D – розмір блоку завдання, який передається мобільному вузлу, що бере участь у процесі побудови блокчейну;

V_i^t – швидкість передачі даних мобільному вузлу, який бере участь у процесі побудови блокчейну;

p_i – потужність передачі даних мобільним пристроєм;

N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну.

Оскільки дані про час і енерговитрати обраховують для всіх моделей, які використовуються у запропонованій моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, тому було введено верхній індекс t, який означає що дані значення відносяться до моделі завантаження даних в мобільні вузли, які беруть участь в побудові блокчейну в розподілених системах.

Отже, модель враховує перетворення вхідних даних на наступні вихідні дані:

$$\zeta = \langle T_i^t, E_i^t \rangle, \tag{2.7}$$

де ζ – вихідні дані моделі завантаження даних в мобільні вузли, які беруть участь в побудові блокчейну в розподілених системах;

T_i^t– час передачі блоку завдання мобільному вузлу, який бере участь у процесі побудови блокчейну;

E_i^t – загальні енерговитрати на процес передачі блоку завдання мобільному вузлу, який бере участь у процесі побудови блокчейну;

В подальшому отриманні даною моделлю значення, використовуються в загальній моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат.

2.2 Модель вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах

Процес вирішення PoW задачі, або майнінговий процес, являється виконанням складної математичної задачі тривіальним методом підбору значення. Сам процес являється часозатратним і енерговитратним, що, доречі, і стало причиною даної постановки задачі. Для створення моделі вирішення PoW задачі мобільними вузлами, які беруть участь у процесі побудови блокчейну, було вирішено визначити такий показник як інтенсивність обчислень, який описує середню кількість обчислень на кожен біт завдання, які необхідні для виконання поставленої PoW задачі мобільними вузлами.

Модель вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах враховує отримання наступних вхідних даних, для кожного вузла:

$$\vartheta = \langle N', D, X, f_i \rangle, \tag{2.8}$$

де ϑ – вхідні дані моделі вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах;

N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну;

D – розмір блоку завдання, який передається мобільному вузлу, що бере участь у процесі побудови блокчейну;

X – середня кількість обчислень на кожен біт завдання, які необхідні для виконання поставленої PoW задачі мобільними вузлами;

f_i – обчислювальні ресурси мобільного пристрою.

Як вже було сказано, дана модель повинна обрахувати часові і енерговитрати, які характерні для процесу вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну. Для обрахунку часу виконання завдання мобільним вузлом, модель вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах використовує наступну формулу:

$$T_i^m = \frac{DX}{f_i},\ i\ \in N',\tag{2.9}$$

де T_i^m – час виконання операції вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну;

D – розмір блоку завдання, який передається мобільному вузлу, що бере участь у процесі побудови блокчейну;

X – середня кількість обчислень на кожен біт завдання, які необхідні для виконання поставленої PoW задачі мобільними вузлами;

f_i – обчислювальні ресурси мобільного пристрою;

N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну.

Як видно з формули 2.9, час виконання прямо пропорційний розміру завдання і інтенсивності обчислень конкретного завдання, а також обернено пропорційний обчислювальним ресурсам учасника. Оскільки X визначає середнє значення кількості обчислень на кожен біт завдання, то і обрахований час виконання операції вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну, буде середнім між максимальним і мінімальним часом виконання даної задачі при доступних обчислювальних ресурсах.

Після того як був описаний час виконання PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах, можна обрахувати енерговитрати мобільного вузла за час виконання PoW задачі,

базуючись на обчислювальних ресурсах мобільного пристрою. Оскільки енерговитрати напряму залежать від обчислювальних можливостей мобільного пристрою, тобто в залежності від кількості операцій в секунду, яку виконує процесор мобільного пристрою, енерговитрати будуть значно зростати в порівнянні зі звичайним режимом роботи мобільного пристрою. Модель вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах передбачає опис енерговитрат наступною формулою:

$$E_i^m = k_1 f_i T_i^m,\ i \in N', \tag{2.10}$$

де E_i^m – енерговитрати на виконання операції вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну;

k_1 – коефіцієнт, який описує залежність між обчислювальними ресурсами і енерговитратами;

f_i – обчислювальні ресурси мобільного пристрою;

T_i^m – час виконання операції вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну;

N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну.

Як і в випадку з часом затраченим на процес виконання PoW задачі, формула 2.10 обраховує середнє значення енерговитрат, оскільки використовує середнє значення часу.

Окрім енерговитрат модель вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах, повинна врахувати вірогідність виконати поставлену PoW задачу вчасно. Оскільки одним з пунктів визначення ефективність побудови блокчейну, визначається можливість своєчасного виконання PoW задачі, то модель повинна врахувати вірогідність того що задача була виконана вчасно, в залежності від середнього часу виконання задачі мобільним вузлом, і коефіцієнтом який пов’язаний з допустимим часом виконання PoW задачі. Чим швидше виконується завдання, тим більша вірогідність що воно

було виконано вчасно, саме тому вірогідність вірогідність успішного виконання PoW задачі повинна бути обернено пропорційна часу виконання. Окрім того, враховується коефіцієнт який буде масштабувати це значення, і пов’язаний з максимально допустимим часом виконання задачі.

Для виконання описаних вище пунктів модель вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах використовує наступну формулу:

$$P_i^m = k_2 \frac{1}{T_i^m},\ i \in N', \tag{2.11}$$

де P_i^m– це вірогідність виконати PoW задачу мобільними вузлами, які беруть участь в побудові блокчейну, вчасно;

k_2 – коефіцієнт пов’язаний з максимально допустимим часом виконання задачі;

T_i^m– час виконання операції вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну;

N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну.

Як видно з формули 2.11, чим швидше виконується завдання, тим більше вірогідність що воно буде виконано вчасно. Підбір значення для коефіцієнтів відбудеться при створенні методу.

На виході моделі отримуються наступні значення, для кожного мобільного вузла, який бере участь в побудові блокчейну в розподілених системах шляхом перетворень вхідних даних (формула 2.8):

$$\varphi = \langle T_i^m,\ E_i^m,\ P_i^m \rangle, \tag{2.12}$$

де φ – вихідні дані моделі вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах;

T_i^m – час виконання операції вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну;

E_i^m – енерговитрати на виконання операції вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну;

P_i^m – це вірогідність виконати PoW задачу мобільними вузлами, які беруть участь в побудові блокчейну, вчасно;

Отриманні вихідні значення використовуються в моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат.

2.3 Модель поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах

Модель поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах по своїй суті схожа на модель завантаження даних в мобільні вузли, які беруть участь в побудові блокчейну в розподілених системах. Як і модель модель завантаження даних в мобільні вузли, які беруть участь в побудові блокчейну в розподілених системах, дана модель повинна передбачати передачу даних в мобільних мережах, і враховувати можливі фонові шуми, які наявні у цих мережах.

Модель поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах повинна передбачити передачу блок іншим учасникам мобільної мережі для його затвердження і прийняття, в нашому випадку достатньо передати блок на сервер. Після того як блок поширюється іншим учасникам мережі, механізм досягнення PoW консенсусу передбачає що вузли перевірять блок і приймуть чи відхилять його в залежності від того чи була виконана задача, і чи була вона виконана вчасно. Крім того, потрібно отримати досягнення консенсусу швидше решти учасників.

Також, модель поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах, повинна врахувати часові затримки, які притаманні при поширенні блоку в мобільній мережі. Окрім, вирішення PoW

задачі вчасно, мобільний вузол повинен виконати поширення, отриманого в процесі вирішення задачі блоку, в встановлений час. Якщо, поширення блоку виконується надто повільно є ризик того що PoW консенсус не буде досягнуто в відношенні отриманого блоку, і його відхилять.

Модель поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах передбачає наступні вхідні дані:

$$\lambda = \langle N',\ W,\ p_i,\ \sigma,\ H_i,\ D, t \rangle, \tag{2.13}$$

де λ – вхідні дані моделі поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах;

N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну;

W – пропускна здатність серверу;

p_i – потужність передачі даних мобільним пристроєм;

H_i – статус бездротового каналу між мобільним вузлом і сервером;

σ – похибка пов’язана з можливими фоновими шумами в мережі;

D – розмір блоку завдання, який передається мобільному вузлу, що бере участь у процесі побудови блокчейну;

t – час необхідний для перевірки отриманого блоку, механізмами PoW досягнення консенсусу.

Як було сказано, дана модель схожа на модель завантаження даних в мобільні вузли, які беруть участь в побудові блокчейну в розподілених системах, саме тому щоб не дублювати формули для швидкості передачі даних, було вирішено використати ті ж результати які були отримані при обчислені швидкості передачі даних в моделі завантаження даних в мобільні вузли, які беруть участь в побудові блокчейну в розподілених системах.

$$V_i^p = V_i^t,\ i \in N', \tag{2.14}$$

де V_i^t– швидкість передачі даних мобільному вузлу, який бере участь у процесі побудови блокчейну;

V_i^p – швидкість передачі даних від мобільного вузла, який бере участь у процесі побудови блокчейну.

Для того, щоб обрахувати час затрачений на поширення блоку, модель поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах, передбачає обрахунки, які використовують швидкість передачі даних і розмір даних які поширюються, крім того, передбачається час на перевірку виконання завдання, що описується наступною формулою:

$$T_i^p = \frac{D}{V_i^p} + t,\ i\ \in N' \tag{2.15}$$

де T_i^p – час передачі блоку завдання іншим мобільним вузлам, які беруть участь у процесі побудови блокчейну;

D – розмір блоку завдання, який передається мобільному вузлу, що бере участь у процесі побудови блокчейну;

V_i^p – швидкість передачі даних від мобільного вузла, який бере участь у процесі побудови блокчейну.

t – час необхідний для перевірки отриманого блоку, механізмами PoW досягнення консенсусу;

N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну.

Формула, яка передбачена моделлю поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах, для енерговитрат на етапі пропагації описується наступним чином:

$$E_i^p = p_i T_i^p,\ i\ \in N', \tag{2.16}$$

де E_i^p – загальні енерговитрати на процес передачі блоку завдання іншим мобільним вузлам, які беруть участь у процесі побудови блокчейну;

T_i^p – час передачі блоку завдання іншим мобільним вузлам, які беруть участь у процесі побудови блокчейну;

p_i – потужність передачі даних мобільним пристроєм;

N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну.

Окрім того, необхідно врахувати вірогідність виконати поширення блоку в мобільній розподіленій системі, від чого залежить процес досягнення консенсусу механізмами PoW, і ефективність вузла, згідно моделі поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах. Вірогідність виконати поширення блоку в мобільній розподіленій мережі вчасно визначається наступною формулою:

$$P_i^p = k_3 \frac{1}{T_i^p},\ i \in N', \tag{2.17}$$

де P_i^p – вірогідність виконати поширення блоку в мобільній розподіленій мережі вчасно;

k_3 – коефіцієнт пов’язаний з максимально допустимим часом на поширення блоку;

T_i^p – час передачі блоку завдання іншим мобільним вузлам, які беруть участь у процесі побудови блокчейну;

N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну.

Отже, модель враховує перетворення вхідних даних на наступні вихідні дані:

$$\mu = \langle T_i^p, E_i^p, P_i^p \rangle, \tag{2.17}$$

де μ – вихідні дані моделі поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах;

T_i^p – час передачі блоку завдання іншим мобільним вузлам, які беруть участь у процесі побудови блокчейну;

E_i^p – загальні енерговитрати на процес передачі блоку завдання іншим мобільним вузлам, які беруть участь у процесі побудови блокчейну;

P_i^p – вірогідність виконати поширення блоку в мобільній розподіленій мережі вчасно.

Вихідні дані моделі поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах використовуються в моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат.

2.4 Модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат

Модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат передбачає отримання наступних вхідних даних:

$$\gamma = \langle \zeta, \varphi, \mu \rangle, \tag{2.18}$$

де γ – вхідні дані моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат;

ζ – вихідні дані моделі завантаження даних в мобільні вузли, які беруть участь в побудові блокчейну в розподілених системах;

φ – вихідні дані моделі вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах;

μ – вихідні дані моделі поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах.

Взаємозв’язки між моделями показано на рисунку 2.1.

Рисунок 2.1 – Схема взаємозв'язків між моделями

Спершу потрібно описати загальну формулу для обчислення середнього шансу на виконання всіх кроків попередніх моделей вчасно. Вірогідність того що і вірогідність виконати PoW задачу мобільними вузлами, які беруть участь в побудові блокчейну, вчасно, і вірогідність виконати поширення блоку в мобільній розподіленій мережі вчасно, буде позитивною, згідно теорії вірогідностей, буде являти добуток двох вірогідностей, що і показано наступною формулою:

$$P_i = P_i^m P_i^p, i \in N' \tag{2.19}$$

де P_i^p – загальна вірогідність виконати створення наступного блоку, мобільним вузлом, вчасно межах блокчейн-мережі;

P_i^m – вірогідність виконати PoW задачу мобільними вузлами, які беруть участь в побудові блокчейну, вчасно;

P_i^p – вірогідність виконати поширення блоку в мобільній розподіленій мережі вчасно;

N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну.

Модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, повинна передбачити

обчислення ефективності побудови блокчейну. Для того, щоб описати ефективність побудови потрібно ввести змінну, яка б показувала наскільки є релевантною участь того чи іншого вузла в множині мобільних вузлів, які беруть участь в процесі побудови блокчейну. Для цієї задачі було вирішено ввести поняття вигоди.

Вигода повинна виражатись, як відношення цінності отриманого результату, вірогідності виконати поставлені задачі з достатньою швидкістю до витрачених мобільним вузлом ресурсів. Модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, повинна враховувати коефіцієнти, які описують важливість тих чи інших параметрів по відношенню до мобільного вузла, який бере участь в процесі побудови блокчейну.

Вигода повинна складатись з двох частин: з позитивної частини, яка виражає необхідність в виконанні процесу побудови блокчейну, і з негативної, яка виражає кількість затрачених енергоресурсів, а також використаних обчислювальних потужностей на процес генерації нового блоку. Враховуючи це, для моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, було створено наступну формолу, яка обраховує значення вигоди для кожного окремого мобільного вузла в мережі:

$$R_i = \frac{rP_i * c_1\left(\frac{T_i^{-1}}{\sum T_j}\right)}{c_2(E_i^t + E_i^m + E_i^p)} - R_{min}, \qquad (2.20)$$

де R_i – вигода від використання конкретного вузла в мобільній розподіленій системі для побудови блокчейну;

r – константне значення, яке пов’язане з необхідністю виконання процесу побудови блокчейну;

P_i – загальна вірогідність виконати створення наступного блоку, мобільним вузлом, вчасно межах блокчейн-мережі;

c_1 – коефіцієнт на пов’язаний з цінністю затраченого часу на побудову блокчейну;

T_i – час обчислення блоку мобільним вузлом;

c_2 – коефіцієнт на пов'язаний з цінністю енерговитрат;

E_i^t – загальні енерговитрати на процес передачі блоку завдання мобільному вузлу, який бере участь у процесі побудови блокчейну;

E_i^m – енерговитрати на виконання операції вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну;

E_i^p – загальні енерговитрати на процес передачі блоку завдання іншим мобільним вузлам, які беруть участь у процесі побудови блокчейну;

f_i – обчислювальні ресурси мобільного пристрою;

R_{min} – мінімальне значення вигоди передбачене моделлю;

N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну.

Окрім вигоди яку отримує кожен окремий мобільний вузол в мобільній розподіленій системі для побудови блокчейну, потрібно описати загальну вигоду, яку отримує вся множина мобільних вузлів при побудові блокчейну. Дане значення буде виражати загальну ефективність підібраної множини мобільних вузлів, які беруть участь у процесі побудови блокчейну. Значення загальної вигоди множини мобільних вузлів, описується наступною формулою:

$$R = \sum_{i \epsilon N} R_i, \tag{2.21}$$

де R – загальна вигода вузлів підібраної популяції при побудові блокчейну;

R_i– вигода від використання конкретного вузла в мобільній розподіленій системі для побудови блокчейну;

N'– множина мобільних вузлів, які беруть участь в процесі побудови блокчейну.

Отже, на виході моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, було отримано значення вигоди для певної популяції мобільних вузлів базуючись на значеннях ефективності цієї популяції, і витрачені на процес побудови блокчейну ресурсів.

Подальша максимізація значення вигоди, призведе до підбору, більш ефективної популяції мобільних вузлів, які беруть участь у процесі побудови блокчейну.

2.5 Задача підбору популяції мобільних вузлів

Задача полягає в підборі такої популяції мобільних вузлів, яка є частиною певної множини базових мобільних вузлів, і показує себе значно ефективніше ніж будь які інші популяції, які можна скласти з базової множини мобільних вузлів.

Якщо вважати базову множину вузлів, як N, то підібрана популяція описується наступним чином:

$$N' \subset N, \tag{2.22}$$

де N' – популяція вузлів, які беруть участь в процесі побудови блокчейну;
N – базова множина мобільних вузлів які знаходяться в мережі.

Базуючись на створеній моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, задача по підвищенню ефективності побудови блокчейну виражається наступним чином:

$$\max R = \sum_{i \epsilon N} R_i, \tag{2.23}$$

де R – загальна вигода вузлів підібраної популяції в мобільній розподіленій системі для побудови блокчейну;
R_i – вигода від використання конкретного вузла в мобільній розподіленій системі для побудови блокчейну;
N' – множина мобільних вузлів, які беруть участь в процесі побудови блокчейну.

Тобто задача полягає в підборі такої популяції значення вигоди для якої є максимальним. Окрім вирішення умови вказаній у формулі 2.20, можна

використати наступну систему рівнянь для опису додаткових умов яким повинна підпорядковуватись підібрана популяція:

$$\begin{cases} f^{min} \le f_i \le f^{max} \\ p^{min} \le p_i \le p^{max} \\ T_i^t + T_i^m + T_i^p \le T^{max}, \\ E_i^t + E_i^m + E_i^p \le E^{max} \end{cases} \quad (2.24)$$

де f_i – обчислювальні ресурси мобільного пристрою;

p_i – потужність передачі даних мобільним пристроєм;

T_i^t – час передачі блоку завдання мобільному вузлу, який бере участь у процесі побудови блокчейну;

T_i^m – час виконання операції вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну;

T_i^p – час передачі блоку завдання іншим мобільним вузлам, які беруть участь у процесі побудови блокчейну;

E_i^t– загальні енерговитрати на процес передачі блоку завдання мобільному вузлу, який бере участь у процесі побудови блокчейну;

E_i^m – енерговитрати на виконання операції вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну;

E_i^p – загальні енерговитрати на процес передачі блоку завдання іншим мобільним вузлам, які беруть участь у процесі побудови блокчейну;

Система рівнянь гарантує, що для участі у процесі побудови блокчейну не буде обрано надто слабких, або надто потужних для даної системи пристроїв. Крім того, показано що будь-яка підібрана популяція мобільних вузлів, яка бере участь у побудові блокчейну, повинна виконувати завдання не пізніше ніж вказано у системі, і при цьому не використовувати більше ресурсів ніж це дозволено.

Запропонований метод створений для підвищення ефективності побудови блокчейну шляхом знайдення найбільш ефективної популяції мобільних вузлів, які беруть участь в побудові блокчейну. Алгоритм використовує такий інтелектуальний метод, як алгоритм диференціальної еволюції (ДЕ) [80].

Потрібно враховувати, що більшість відомих аналогів беруть за основу максимізацію швидкості виконання поставлених задач, але не враховують їх енерговитрати. Через це відомі аналоги можуть мати більш значний приріст в швидкості побудови блокчейну ніж запропонований метод, тому було вирішено використовувати саме алгоритм ДЕ, а не будь-який інший з можливих ЕА.

Використання ДЕ дозволить швидше і ефективніше знайти максимально ефективну популяцію мобільних вузлів, які беруть участь у побудові блокчейну, згідно моделі, а також дозволить підібрати кращу популяцію засновану на більшій кількості пунктів, які враховуються, що дозволить зменшити негативний вплив на час побудови блоків, через підвищення енергоефективності, якщо не зовсім його позбутись.

Оскільки подібний алгоритм базується на незначних виконані незначних змін в популяції, то і кожна ітерація алгоритму буде краще впливати на ефективність побудови блокчейну в мережі.

Враховуючи що кожна ітерація алгоритму буде виконувати лише одну зміну, будь то виконання операції додавання, видалення чи заміни, то і зміни в популяції можна буде легко відслідкувати і проаналізувати, що в свою чергу дозволяє підвищити ефективність запропонованого методу, який базується на алгоритмі ДЕ.

Вирішення поставленої задачі максимізації вигоди (формула 2.23) дозволяє знайти таку популяцію яка буде максимально ефективною в побудові блокчейну, оскільки вона враховує як енерговитрати всіх вузлів мережі, так і швидкість створення нових блоків мобільними вузлами, які були обрані для участі у процесі побудови блокчейну. Також, враховується час виконання задачі в мережі, і загальні енерговитрати всіх мобільних вузлів, які беруть участь у процесі побудови блокчейну.

Метод полягає в тому, що необхідно підвищити ефективність обрахунку нових блоків, що в свою чергу збільшить загальну швидкість, і ефективність побудови блокчейну. Для цього використовуються ЕА, який шляхом підбору найбільш ефективної популяції вузлів мережі, підбирає найбільш ефективну популяцію мобільних вузлів, які беруть участь у процесі побудови блокчейну.

Як і для будь-якого ЕА, для порівняння популяцій необхідно зробити схему кодування яка задовільнить наступним умовам:

- буде максимально компактною;
- дозволить відображати вузли які беруть участь у процесі побудови блокчейну;
- буде відображати необхідні дані про вузли.

Тому було запропоновано представлення, яке показано на рисунку 2.2.

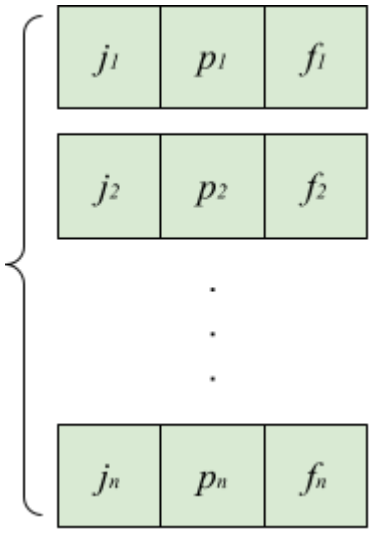


Рисунок 2.2 – Базова схема кодування для загальної популяції

Дана схема кодування (рис 2.2) дозволяє зберігати інформацію про базову множину мобільних вузлів. Відображає участь (j) кожного мобільного вузла у процесі побудови блокчейну, показує потужності передачі (p), і обчислювальні ресурси (f) для кожного з вузлів.

Для збереження похідних популяцій вузлів необхідно створити іншу схему кодування, яка за допомогою моделі побудови блокчейну в мобільних

розподілених системах з урахуванням популяції множини мобільних вузлів та енерговитрат, дозволить порівнювати дві, або більше популяцій між собою, і визначати яка з них є більш ефективною.

Тому, була створена схема кодування (рис 2.3), яка є коротшою. Крім того, дана схема кодування створена виключно для збереження популяції мобільних вузлів які приймають безпосередню участь у процесі побудови блокчейну, а також для тих популяцій, які порівнюються з даною.

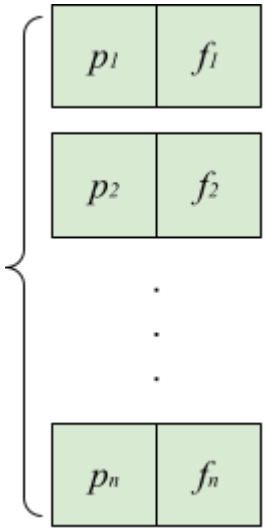


Рисунок 2.3 – Схема кодування, яке використовується для опису підібраної популяції мобільних вузлів

Варто наголосити, що використання подібного компактного представлення можливе лише за рахунок того що використовується алгоритм ДЕ. Також, така компактна реалізація все ще потребує додаткового індексування в коді.

2.6 Висновок

У цьому розділі було представлено і описано модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат. На відміну від аналогічних моделей, дана модель приділяє особливу

увагу часу побудови блоків та енерговитратам, і враховує специфіку роботи мобільних розподілених систем.

Модель передбачає врахування таких показників, як енерговитрати, час виконання операцій, вигода від використання як конкретного мобільного вузла, так і популяції мобільних вузлів, для участі у процесі побудови блокчейну. Врахування цих показників є достатнім для створення методу, який надасть можливість підвищити ефективність побудови блокчейну в мобільних розподілених системах.

Було введено значення вигоди використання певного вузла в процесі побудови блокчейну, яке визначає релевантність використання даного вузла для участі в множині мобільних вузлів, які беруть участь у процесі побудови блокчейну, що, в свою чергу, дозволяє підібрати найбільш ефективну популяцію для виконання поставлених задач.

Саме тому, створена модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат підходить для створення інтелектуалізованого методу побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, і майбутньої побудови, на основі методу, інтелектуалізованої мобільної розподіленої системи для побудови блокчейну.

3 ІНТЕЛЕКТУАЛІЗОВАНИЙ МЕТОД ПОБУДОВИ БЛОКЧЕЙНУ В МОБІЛЬНИХ РОЗПОДІЛЕНИХ СИСТЕМАХ З УРАХУВАННЯМ ЧАСУ ПОБУДОВИ БЛОКІВ ТА ЕНЕРГОВИТРАТ

3.1 Основи інтелектуалізованого методу побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат

Алгоритм інтелектуалізованого методу для підвищення ефективності побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, передбачає в своїй роботі використання такого інтелектуального методу, як ДЕ, що дозволяє підібрати таку популяцію мобільних пристроїв, яка буде найбільш ефективною згідно моделі. Модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, в свою чергу базується на обчислені значень вигоди від використання тої чи іншої популяції.

Загальний алгоритм повинен передбачати наступні моменти своєї роботи:

- створення базової популяції з загальної множини вузлів;
- модифікацію популяції шляхом зміни її складу, не більше однієї зміни за ітерацію;
- порівняння двох популяцій, на основі створеної моделі, і виживання “сильнішої”.

Тому в цьому розділі передбачені алгоритми, які і будуть являти основне тіло майбутнього алгоритму. Ці алгоритми описують введення і аналіз вхідних даних, виконання обчислень на основі створеної моделі, виконання змін в популяціях.

Першим кроком, початку роботи будь-якого алгоритму потрібно ініціалізувати та ввести початкові дані. Першим варто ініціювати такі базові змінні, як кількість елементів у початковому масиві, а також кількість тих елементів з яких буде формуватись перша популяція. Крім того потрібно передбачити ту кількість обчислень після якої алгоритм завершить свою роботу. Як і більшість

інтелектуальних методів, роботу даного методу варто зупиняти, через певну кількість обчислень, оскільки вони являються такими що не дають результатів.

На другому кроці, оскільки в нас не має заданих значень, потрібно сформувати загальний масив елементів, кожен з яких буде представляти собою потенційного майнера. Створений масив повинен передбачати такі змінні як: участь у процесі побудови блокчейну для кожного з елементів, обчислювальні ресурси, потужність передачі. Участь для у процесі побудови блокчейну на першому етапі дорівнює нулю, і використовуються лише для відображення тих вузлів, які будуть обрані для участі у побудові блокчейну, а значення потужності передачі і обчислювальних ресурсів обираються з проміжку який описаний в формулі 2.22.

Далі потрібно сформувати першу популяцію мобільних вузлів, які беруть участь у побудові блокчейну, з елементів які були обраним випадковим чином в кількості обраній раніше. Дана популяція буде постійно змінюватись під час проходження алгоритму, і ставати “сильнішою”.

Також одним з основних елементів алгоритму є вектор вірогідності операцій, який передбачає яким чином популяція буде змінюватись на певному кроці алгоритму. Даний вектор буде постійно адаптуватись в залежності від кількості застосованих до популяції операцій, і від того наскільки доречними були застосовані зміни.

Передбачається три види операцій для зміни популяції:

- додати елемент;
- вилучити елемент;
- замінити елемент.

Дані операції детально описані в подальшому. Можна сказати що вони являють собою той механізм за допомогою якого створюються змінені популяції.

Окрім, вищеописаного, можна створити список тих елементів які будуть негативно відображатись на роботі мережі в цілому. Ініціалізувати його потрібно

як порожній список, в який під час роботи алгоритму повинні записуватись небажані елементи.

Оскільки, оптимальна популяція, являється невідомою, і важко знаходиться, а кількість можливих варіантів популяції, за допомогою якої досягається максимально висока ефективність збільшується пропорційно загальній кількості множини вузлів, то для досягнення поствленої задачі було вирішено обрати ЕА.

На основі цього, було створено наступний алгоритм функції для знайдення максимально ефективної популяції:

1. В якості вхідних даних приймається мінімальна кількість учасників (n_min), а також максимальна кількість обчислень (MaxE), і загальна кількість учасників (n).
2. Ініціалізувати змінні (p, f) для числових значень потужності передачі і обчислювальних ресурсів відповідно, і записати у масив N розміром n.
3. З масиву N випадковим чином обрати n_min учасників для масиву N'.
4. Виконати базові обчислення за формулами 3.1, на основі сформованого масиву N'.
5. Ініціалізувати вектор для вірогідностей op = {op1, op2, op3}, який повинен бути нормалізований.
6. Ініціалізувати список BList для запису неефективних учасників.
7. Поки кількість операцій не досягне MaxE, виконувати кроки (8-10).
8. Використовуючи операції мутації створити змінений масив N''.
9. Використовуючи N'', функцію для оновлення популяції мобільних вузлів, які беруть участь в побудові блокчейну в розподілених системах.
10. Використовуючи N'', функцію для адаптації шансів мутацій популяції мобільних вузлів, які беруть участь в побудові блокчейну в розподілених системах.
11. На виході функції отримуємо N.

Для збільшення ефективності методу, можна виконати алгоритм декілька разів, що створить декілька схожих за своїми характеристиками популяцій, що в

свою чергу дозволить обирати яку саме популяцію мобільних вузлів вигідно використовувати в залежності від ситуації і поставлених задач.

Порівняння двох популяцій і визначення серед них більш ефективної відбувається за рахунок проведення обчислень на основі моделі. Та популяція мобільних вузлів, яка показує кращі результати відносно моделі, тобто значення вигоди використання для якої більше, вважається більш ефективною.

На практиці подібне порівняння виконується емпіричним шляхом, та популяція яка виконує конкретну задачу швидше і з меншими затратами вважається більш ефективною.

В якості вхідних даних обирається загальна популяція, мінімальна кількість вузлів яка повинна брати участь у процесі побудови блокчейну, а також та кількість ітерацій після якої алгоритм завершує свою роботу.

В якості вихідних даних обирається загальна популяція в якій помічені ті мобільні вузли які беруть участь в побудові блокчейну.

Алгоритм же використовує підхід ЕА, для створення і заміни популяцій, який шляхом виконання перевірок на основі моделі, обирає з двох запропонованих популяцій, сильнішої.

Окрім вхідних/вихідних даних і виконання базових обчислень, запропонований алгоритм включає ряд функцій які описані в таблиці 3.1. Частина основних операцій виконується саме в цих функціях і в подальшому вони будуть описані більш детально.

Таблиця 3.1 – Властивості створених функцій

Функція	Призначення
Функція для оновлення популяції мобільних вузлів, які беруть участь в побудові блокчейну в розподілених системах	обирає з двох масивів(популяцій) мобільних вузлів, той який показує себе краще згідно моделі. Тобто, виконує заміну слабшої популяції сильнішою.

Кінець таблиці 3.1 – Властивості створених функцій

Функція	Призначення
Функція для адаптації шансів мутацій популяції мобільних вузлів, які беруть участь в побудові блокчейну в розподілених системах	в залежності від результатів попередньої функції може змінити вірогідності застосування операцій до базової популяції, крім того вносить зміни в чорний список

В процесі роботи алгоритму, базовий масив N' повинен постійно змінюватись шляхом використання операцій додавання, видалення, заміни. Якщо, створений масив N'', виявляється ефективнішим, згідно моделі, ніж N', то популяція N', вважається слабкою і повинна бути замінена більш пристосованою N''. Варто зазначити що розмір масиву N'', повинен бути таким же, або відрізнятись не більше ніж на один елемент від розміру масиву N. Крім того, постійно оновлюються значення m та N', а також значення op та BList, за допомогою функцій алгоритм яких описаний в подальшому. Алгоритм потрібно повторювати поки не буде досягнута максимальна кількість операцій (MaxE).

В результаті роботи алгоритму повинен бути отриманий масив N' який буде являти собою максимально ефективну популяцію, для побудови блокчейну, де ефективністю вважається оптимальне рішення між енерговитратами, швидкістю, і загальними використаними ресурсами, що описано в моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат.

Після того як алгоритм буде виконано, являється допустимим провести ряд додаткових перевірок елементів масиву N', з метою знайдення тих які негативно впливають на результати роботи даної популяції. Варто зазначити що подібні дії не являються обов’язковими, оскільки в результаті роботи алгоритму повинна бути

сформована максимально ефективна множина майнерів, але вони можуть бути корисними в випадку якщо випадково, склалось не так, хоч це і мало вірогідно.

На основі вище перерахованого, було створено наступний алгоритм для перевірки ефективності всіх значень сформованої популяції:

1. Для кінцевої популяції N', для кожного елементу в ній, виконати наступні кроки (2-4).

2. Видалити елемент з популяції таким чином створивши N".

3. Якщо N" ефективніше за N', зберегти N", як N'.

4. Перейти до наступного елементу.

5. Вивести фінальну версію створеної популяції N'.

Після того як було виконане видалення не доцільних елементів, варто зробити перевірку чорного списку на присутність в ньому елементів, які б могли бути додані до основної популяції, після видалення деяких з елементів.

Це необхідно, оскільки існує вірогідність що якийсь з елементів був доданий туди випадково, на основі тої популяції яка була на той момент, тому додаткова перевірка чорного списку дозволить переконатись у тому що дана популяція не потребує додаткових змін, або в іншому випадку внести деякі зміни в склад популяції N'.

Саме з перерахованих вище причин було створено наступний алгоритм перевірки чорного списку, який передбачає перевірку чорного списку і вибору найкращого можливого варіанту для вставки в популяцію:

1. Для кожного елементу чорного списку виконати наступні кроки (2-4).

2. Створити популяцію N" з додаванням обраного елемента

3. Якщо популяція виявляється кращою ніж N' і кращою ніж попередня популяція яка була запам’ятована, запам’ятати популяцію як M'.

4. Перейти до наступного елементу.

5. Замінити N' на M', якщо M' не дорівнює порожній множині.

З запропонованого алгоритму можна зрозуміти, що в основну популяцію додається лише один елемент з чорного списку, який підходить найкраще для

створеної моделі. Подібний підхід застосовується оскільки невідомо скільки точно елементів поміститься в популяцію і яка їх найкраща комбінація, але додавання найкращого, згідно моделі, елементу внесе максимально можливий позитивний ефект для всієї популяції.

Звісно, якщо такого елементу не було знайдено, або якщо в популяції і так знаходиться максимальне число елементів, то жодних дій з нинішньою популяцією проведено не буде.

Отже, виконання двох вище описаних алгоритмів, дозволить за відносно незначну кількість операцій ще більше підвищити ефективність побудови блокчейну.

Останім кроком, буде вивід отриманих результатів, і побудова графіків які покажуть як змінювались значення часу, енерговитрат, загальних обчислювальних ресурсів, і вигода всіх хто бере участь в процесі. Звісно для цього в коді потрібно врахувати і збереження даних, окрім базового алгоритму, що дозволить побудувати графіки, і проаналізувати загальну ефективність запропонованого методу.

Окрім всього вищесказаного, при реалізації запропонованого методу потрібно врахувати необхідність зберігати результати моделі для кожної з популяцій, для можливості відслідковувати зміни в моделі для всіх змін в популяції.

Для збереження подібних даних можна використати будь-яку структуру даних, але рекомендується використовувати список, або чергу для подібної задачі, що в свою чергу зменшить місце і час опрацювання цих даних при створені графіків на їх основі. Звісно це не являється критичним, оскільки вивід допоміжних даних не обов’язковий і несе виключно інформаційний характер, але це теж потрібно враховувати у випадку якщо популяція досить велика, як і кількість виконуваних, над нею, обчислень.

Унікальність даного методу заключається в тому що увага приділяється не лише швидкості, але і на загальних енерговитратах, мінімізація яких є одним з важливих пунктів роботи методу. Оскільки процес досягнення консенсусу в PoW

блокчейнах, являється дуже енерговитратним процесом, задача по зменшенню енерговитрат навіть шляхом незначної втрати в швидкодії являється дуже важливою і даний метод ставить за мету підбір такої популяції мобільних вузлів мережі, яка б витрачала менше енергоресурсів, при незначній втраті в швидкості.

Звісно цю задачу можна було б вирішити шляхом створення методу для PoW, який би спрощував обрахунок хеш-функцій і таким чином зменшував як необхідні на майнінг ресурси, так і збільшував би швидкість, але було вирішено створити метод, який зможе шляхом підбору оптимальної популяції виконувати поставлену задачу ефективніше, що можна використовувати і в інших PoW задачах необов’язково пов’язаних з блокчейном чи криптовалютами.

3.2 Функція для оновлення популяції мобільних вузлів, які беруть участь в побудові блокчейну в розподілених системах

Для виконання оптимізації рішень про використання вузла в побудові блокчейну, потрібно обирати мобільні вузли які беруть участь у процесі, і шляхом оновлення числа і складу вузлів, які беруть участь у процесі побудови блокчейну, тобто N', крім цього потрібно слідкувати за тими хто вже бере участь в процесі, з перспективи загальної популяції N. Для задачі оптимізації необхідно враховувати два момента:

- Як описано в розділі 3.2, кожен елемент в масиві N', відображає виділені ресурси майнера. Тому, розмір масиву рівний кількості мобільних вузлів, котрі беруть участь у процесі побудови блокчейну. Оскільки оптимальне число мобільних вузлів, які повинні брати участь у процесі побудови блокчейну, невідомо, і являється однією з шуканих величин, кількість учасників повинна мати можливість, як збільшуватись у розмірі, додаючи нових учасників, так і з зменшуватись, видаляючи нинішніх. Крім, того потрібно врахувати, що розмір масиву не повинен надто швидко змінювати свій розмір, щоб краще контролювати склад тієї популяції, яка була обрана для побудови блокчейну.

– Важливим пунктом являється і те що б запобігти участь неефективних майнерів у процесі, для цього і потрібний BList. Неефективними можна вважати ті елементи, після додавання/вилучення яких у процес, не змінює загальну ефективність.

Враховуючи перший пункт, було розроблено три операції, а саме, операції, вставки, видалення і заміни, вони створюють три нових масиви, розміри, яких дорівнюють |N'+1|, |N'-1| і |N'|, відповідно. Використовуючи, одну з створених операцій, можна або збільшувати розмір масиву, або зменшувати, або залишати не змінним, в залежності від рішеня алгоритму, яке приймається на основі значення op. Крім того, на кожному кроці алгоритму масив буде змінюватись лише на одного учасника, що в свою чергу гарантує стабільність роботи алгоритму, і його результативність.

Також потрібно враховувати, що впродовж проходження алгоритму всі значення вектору op повинні адаптуватись в залежності від того наскільки часто використовувалась та чи інша операція, тобто та операція яка виконувалась дуже часто і не давала результатів, повинна отримувати менший шанс на повторне використання.

Для врахування другого пункту, створюється чорний список в який будуть включені всі хто не показує достатньої ефективності, і більше не будуть включені в основний масив N'', подібний чорний список не дозволить повторного використання тих вузлів, які негативно показують себе в роботі мережі.

Детальніше про оновлення op і BList описано в наступному розділі. В даному розділі нас цікавить саме алгоритм оновлення популяції, і виконання операцій додавання, вставки, чи заміни.

Алгоритм функції повинен передбачати:

– створення популяції-конкурента, шляхом виконання операцій вставки, видалення чи заміни елемента;

– порівняння двох популяцій;

– повернення сильнішої популяції;

– оновлення складу мобільних вузлів, які беруть участь у процесі побудови блокчейну.

Спершу, потрібно згенерувати випадкове число, і згідно нього обрати операцію яка буде виконуватись на цьому кроці для оновлення m та N', а саме додавання, видалення і вставки. Оновлені значення масиву є модифікацією існуючого N' та називається N''. Варто уточнити, що якщо розмір масиву |N'| рівний n_max або n_min, то операція додавання або видалення елементу, відповідно, не буде виконуватись, а буде обиратись інша випадкова операція. Базуючись на цьому, операції виконуються наступним чином:

1. Операція вставки – індивід, котрий не бере участь у процесі побудови блокчейну, а також не знаходиться у чорному списку BList, випадковим чином обирається для участі. Після чого, він додається у масив N''.

2. Операція видалення – учасник процесу обраний випадковим чином, видаляється з масиву N''.

3. Операція заміни – елемент з N', замінюється на випадковий елемент з N. Змінена популяція записується як N''.

Після чого, виконуються обчислення для N'' на основі створеної моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат. Якщо, N'' показує себе краще ніж N', відбувається заміна N' на N''. Варто зазначити, що даний пункт трішки відрізняється від типових EA, оскільки відбувається заміна лише окремого елементу.

Для виконання описаних задач, було створено наступний алгоритм функції для оновлення популяції мобільних вузлів, які беруть участь в побудові блокчейну в розподілених системах:

Алгоритм для оновлення m та N':

1. Вхідні дані: масиви N' та N'', кортеж вірогідностей op, та чорний список BList.

2. Згенерувати випадкове число в межах [0, 1].

3. Обрати операцію op, в залежності від згенерованого числа.

4. Виконати операцію в залежності від вибору, аби отримати новий N".

5. Виконати обчислення функції 3.1 для створеного в попередньому пункті N".

6. Збільшити лічильник операцій Es.

7. Якщо, N" виявиться краще ніж N', базуючись на правилі доцільності [3], то виконати заміну.

8. Якщо, масив N' було змінено, замінити значення m в основному масиві N в залежності від тих елементів, які було змінено.

9. Вивести N' та N".

3.3 Функція для адаптації шансів мутацій популяції мобільних вузлів, які беруть участь в побудові блокчейну в розподілених системах

Для зміни значення op, і адаптації вірогідності вибору певної операції яка буде виконана наступною, потрібно проаналізувати ту операцію котра була виконана на попередньому кроці, і зробити висновок в залежності від того як змінилась ефективність роботи популяції, згадно моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, після зміни. Наприклад якщо, була виконана операція над N', і при цьому ефективність роботи зросла потрібно збільшити вірогідність вибору такої ж операції на наступному кроці, в іншому випадку, якщо операція була виконана і ефективність не змінилась, або стала гіршою, то, відповідно, потрібно зменшити вірогідність вибору цієї ж операції. Для цієї задачі, було створено наступні функції:

$$rew(op) = (1 - op)e^{-2op}, \tag{3.1}$$

$$pen(op) = ope^{-2(1-op)}. \tag{3.2}$$

Створені функції нагород (функція 3.1), і штрафів (функція 3.2), мають за мету ввести такі поняття, як винагорода і штраф в вірогідність вибору функції. Як

показано на рис. 3.3 значення нагороди rew(op) змінюється від γ до 0, по мірі зміни значення op від 0 до 1. Таким чином, більша нагорода отримується коли op має менше значення, і менша нагорода коли op має більше значення. Це необхідно, оскільки коли окремий елемент op має велике значення, операція має великий шанс на те щоб бути обраною, а значить бажано обмежити його подальший ріст. В цьому випадку, зменшення винагороди може запобігти надто стрімкому росту значення. Додатково, великий штраф накладається у випадку якщо значення op стає надто великим. В результаті rew(op) і pen(op) мають протилежні тенденції росту/спаду, що показано на рисунку 3.1.

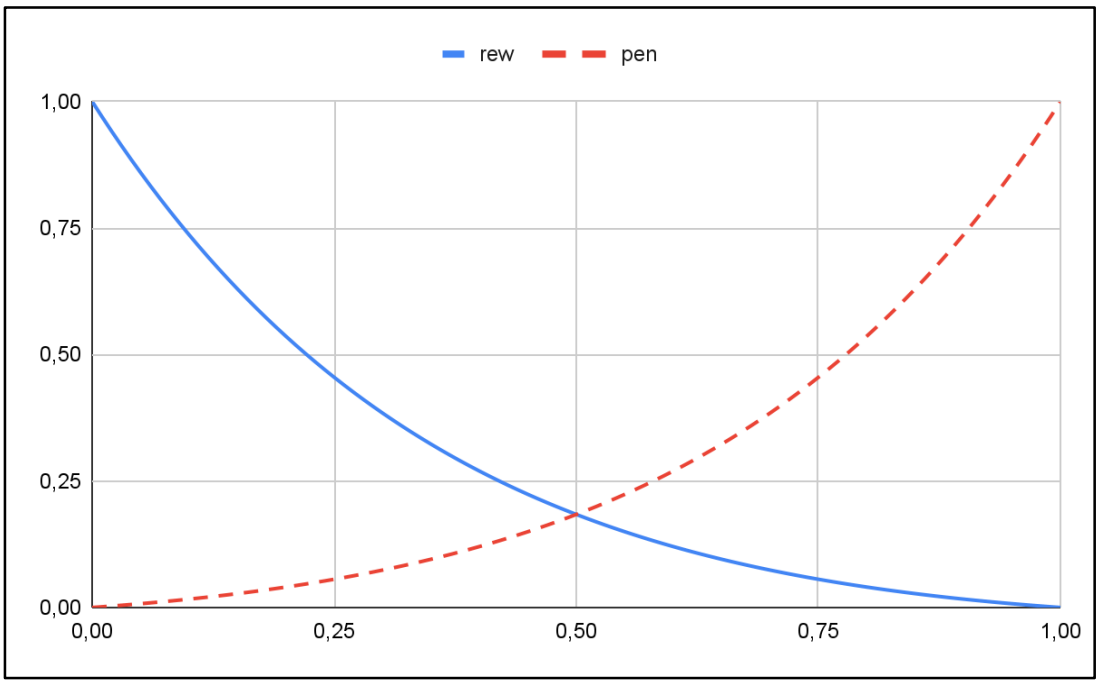


Рисунок 3.1 – Графік функцій 3.1 та 3.2

Початкове значення для n_min може бути досить малим, тому початкові значення для операцій вставки і заміни повинні бути більшими ніж для видалення елемента. Наприклад, можна обрати такі значення для op = [0.6, 0.1, 0.3]. Крім того, при зміні якогось із значень потрібно враховувати, що значення в сумі повинні давати одиницю op1 + op2 + op3 = 1.

Незалежно від того яка операція була виконана, ефективність роботи повинна була змінитись. Якщо зміни були позитивні, тобто ефективність побудови блокчейну, згідно моделі, зросла, то цю зміну в популяції потрібно зберегти. Якщо зміни були негативні, або вони були дуже незначними, то потрібно зменшити вірогідність повторення виконаної операції.

Якщо продуктивність майнерів не збільшується при додаванні якогось елементу, чи при його видаленні не зменшується, то елемент над яким проводилась дія можна вважати неефективним, або навіть шкідливим для всього масиву майнерів. Якщо це так то, втручання таких майнерів у процес потрібно виключити. Для цього створюється BList, в який записуються всі подібні індивіди, аби запобігти їх додавання у майбутні процеси.

Оновлення BList виконується в двох випадках:

− Коли після операції вставки N'' не збільшив свою ефективність в порівнянні з N', доданий елемент додається BList.

− Якщо після операції видалення N'', показує себе краще, або так само, елемент який був видалений додається в BList.

В створеному методі розмір чорного списку обмежений, а саме не більше 30% від всіх можливих учасників. У разі переповнення списку, при додаванні нового елементу, один із старих буде виключений зі списку і зможе знову приймати участь у процесі. Виключення зі списку відбувається за принципом черги, тобто перший зайшов, перший вийшов.

Чорний список BList, повинен не тільки зберігати неефективні мобільні вузли, які не повинні брати участь у процесі побудови блокчейну, але й перешкоджати їх втручанню в процес в майбутньому. Обмеження розміру цього списку дозволяє використовувати ці вузли повторно, але після великої кількості обчислень, що в свою чергу означає зміну старої популяції, а отже і можливий позитивний вклад в нинішню популяцію. Саме тому, чорний список має обмежений розмір, який максимально наближений до розміру початкової популяції вузлів, а також структуру черги, в якій елемент який зайшов першим, першим її покидає.

Враховуючи описані вище пункти було створено наступний алгоритм функції для адаптації шансів мутацій популяції мобільних вузлів, які беруть участь в побудові блокчейну в розподілених системах:

1. Вхідні дані: N', N'', op, і BList.

2. Якщо була виконана операція вставки виконати наступні кроки (3-4).

3. Якщо N'' краще ніж N', то збільшити op1, а саме op1 += rew(op1).

4. Якщо N'' не краще ніж N', то зменшити op1, op1 -= pen(op1), і додати доданий до N'' елемент в BList.

5. Якщо була виконана операція видалення виконати наступні кроки (6-7).

6. Якщо N'' краще ніж N', то збільшити op2, op2 += rew(op2), і додати видалений з N'' елемент в BList.

7. Якщо N'' не краще ніж N', то зменшити op2, op2 -= pen(op2)

8. Якщо виконується операція заміни, то потрібно, в залежності від результату збільшити (op3 += rew(op3)), або зменшити (op3 -= pen(op3)) вірогідність оператору op3.

9. Нормалізувати отриманий вектор.

10. Вихідні дані op і BList.

3.4 Інтелектуалізований метод побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат

Після того як був описаний загальний алгоритм роботи інтелектуалізованого метод для підвищення ефективності побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, потрібно виконати програмну реалізацію, і перевірити ефективність запропонованого рішення.

Варто зазначити, що перевірка методу, побудованою таким чином програмою, відбувається на основі даних моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат.

Створена програма повинна передбачати:

- реалізацію запропонованих у методі алгоритмів функцій;
- заповнення початкових даних випадковим чином;
- можливість обрати розмір початкової популяції;
- вивід проміжних даних у файл;
- вивід фінальної, найбільш ефективної, популяції.

Для написання програми, яка реалізовує інтелектуалізований метод для підвищення ефективності побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, було використану мову програмування C#.

Першим кроком, буде створення класу, який використовується для позначення мобільного вузла. Даний клас повинен зберігати змінні, які являються характеристиками даного мобільного вузла, які використовуються у моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків, а саме потужність передачі і обчислювальні ресурси вузла. Крім того, клас повинен передбачати можливість збереження обчислень зроблених на основі моделі.

Для збереження загальної популяції в програмі використовується такої структури даних, як масив. Загальна популяція мобільних вузлів зберігається як масив з кортежу. Однією змінною кортежу є бінарна змінна, яка показує чи бере даний мобільний вузол участь в основній популяції для побудови блокчейну, інша змінна кортежу представляє сам мобільний вузол.

Для збереження популяції використовується така структура даних як словник. Словник дозволяє співставити мобільний вузол з його місцем в загальній популяції.

Програма передбачає введення вхідних даних, а саме кількість мобільних вузлів, які беруть участь у процесі побудови блокчейну. Оскільки програма лише показує ефективність розробленого методу, мобільні вузли генеруються випадковим чином. Кількість мобільних вузлів в початковій популяції

визначається як 25% від загальної популяції, з округленням в більшу сторону, якщо це необхідно.

Кількість ітерацій програми визначається як n^2, де n – розмір загальної популяції мобільних вузлів.

Основна робота програми передбачає реалізацію алгоритмів запропонованих функцій, і виконання обчислень на основі моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат.

Код розробленої програми представлено в додатку А.

Після того, як була створена програма, яка реалізує інетелектуалізований метод побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків, необхідно провести її дослідження, з метою перевірки результатів роботи.

Для проведення дослідження було вирішено запустити розроблену програму для 50, 100, 150 мобільних вузлів відповідно. Під час роботи програми зберігаються, для подальшого аналізу, наступні дані:

- вигода від використання популяції;
- час створення нового блоку популяцією;
- енерговитрати популяції.

Описані вище змінні використовуються для аналізу того як змінювались параметри популяції під час роботи алгоритму, в залежності від ітерації.

Спершу оцінимо як змінювались енерговитрати популяції в залежності від ітерації, для різних початкових даних. Варто зазначити, що програма завершує роботу на 2500, 10000 і 22500 кроці для 50, 100 і 150 розмірних загальних популяцій.

На графіку енерговитрат (рис. 3.2) зображено зміну енерговитрат в залежності від ітерації для всіх загальних популяцій представлених у програмі. З графіку можна сказати, що приблизно 20% від усіх вузлів, не використовуються взагалі. Також видно що не необхідно приблизно 50% ітерацій, щоб позбавитись від усіх зайвих вузлів, які негативно впливають на ефективність роботи популяції

мобільних вузлів, які беруть участь у процесі побудови блокчейну. Наступні ітерації необхідні лише для збільшення загальної ефективності побудови блокчейну.

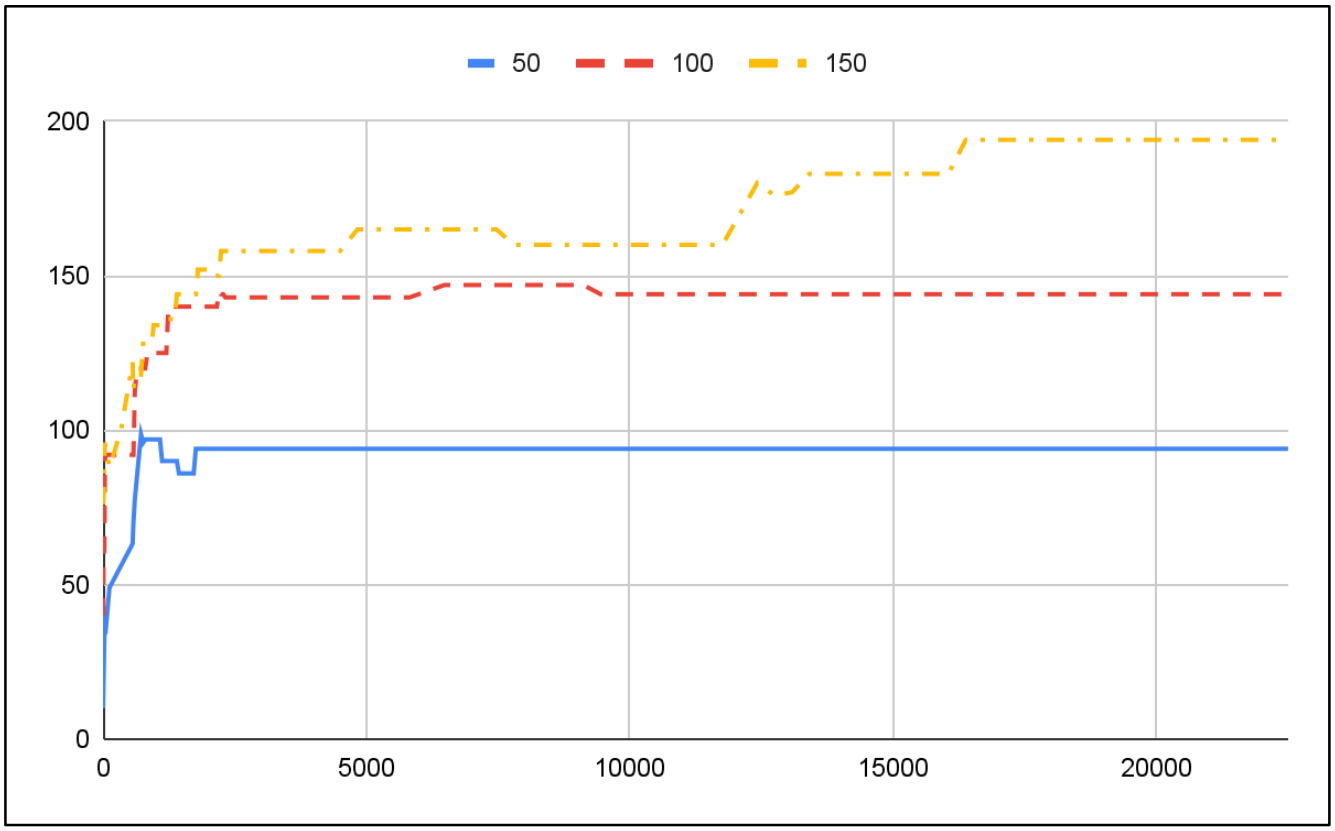


Рисунок 3.2 – Графік залежності енерговитрат від ітерації

На графіку часу побудови нового блоку (рис. 3.3) зображено те як змінювався час виконання процесу побудови блокчейну мобільною розподіленою системою, в залежності від ітерації. З графіку видно, що він частково протилежний до графіку енерговитрат, тобто якщо енерговитрати збільшуються, тобто кількість вузлів в популяції зменшується, то час необхідний для виконання PoW задачі зменшується, і навпаки, якщо енерговитрати зменшуються, то час виконання збільшується.

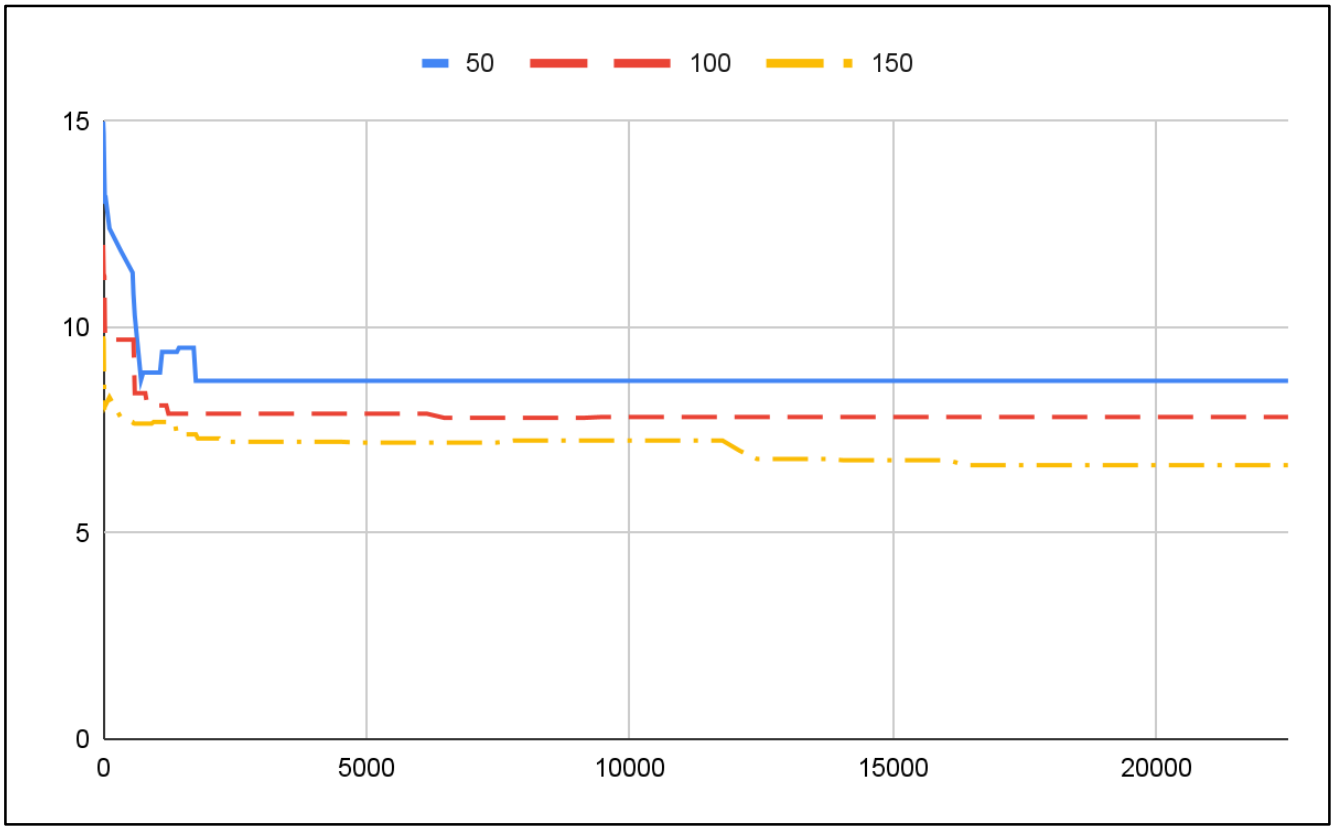


Рисунок 3.3 – Графік залежності часу побудови нового блоку від ітерації

Можна сказати час і енерговитрати змінюються хаотично, але це не так. Кожна зміна яку виконує створений алгоритм призначена для того щоб збільшити загальну вигоду від використання популяції мобільних вузлів. Кожна зміна, яка вносить до популяції, зберігається лише якщо вона була корисною, тобто збільшила ефективність побудови блокчейну.

Коефіцієнти, які пов’язанні з значенням важливості часу і енерговитрат, які враховує модель, обираються на пряму в програмному коді і можуть змінюються в залежності від експерименту який проводиться.

Графік, який зображує ріст вигоди, а отже і ефективності побудови блокчейну (рис 3.4) показує ріст ефективності побудови блокчейну, мобільною розподіленою системою, в залежності від ітерації програми. Крім того варто зазначити, що значення вигоди, згідно моделі, має умовне числове значення, яке відображає відношення часу і необхідності будувати нові блоки, до енерговитрат. Чим більше значення вигоди, тим більш ефективно виконується побудова блокчейну.

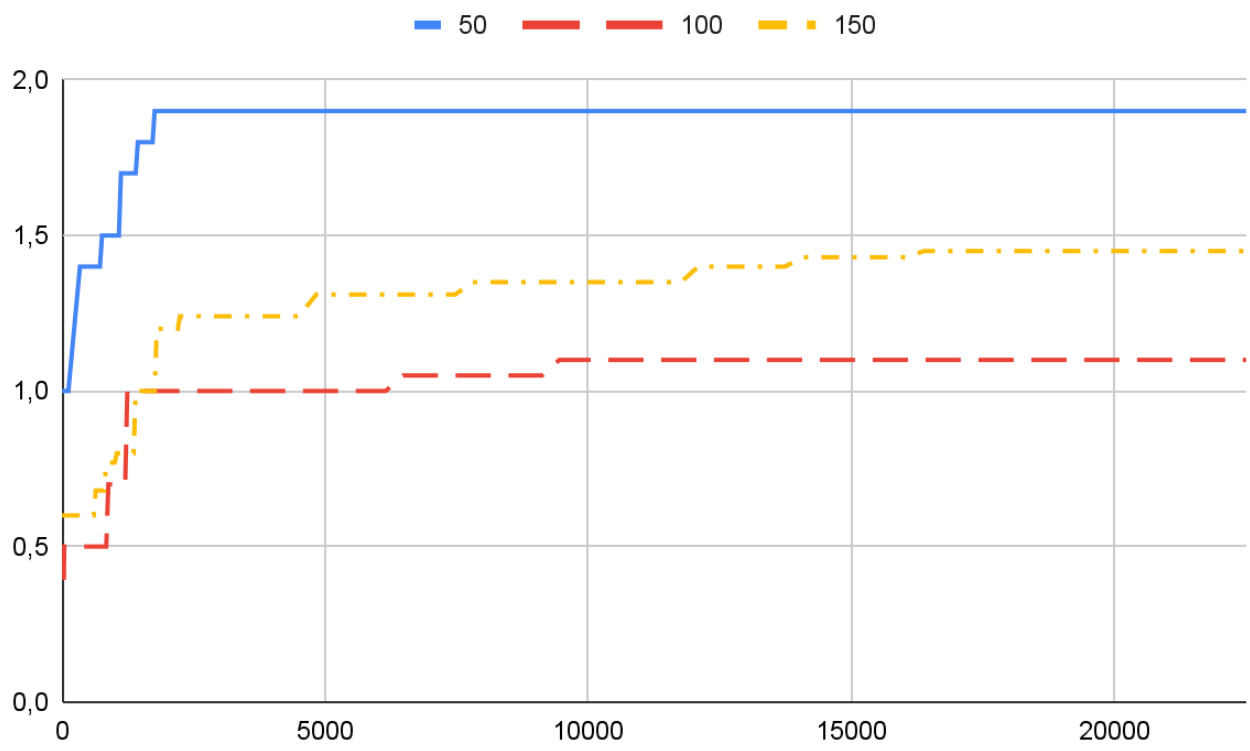


Рисунок 3.4 – Графік залежності значення ефективності побудови блокчейну від ітерації

Отже, після того як були описані зміни значень для вигоди, стають більш зрозумілим зміни значень для енерговитрат і часу побудови нових блоків. З графіку видно, що кожен раз як змінювались значення часу і енерговитрат, загальна ефективність побудови блокчейну зростала.

Важливо розуміти, що при проведенні практичних досліджень, при побудовы апаратно-програмного рішення, створена програма повинна бути змінена. Потрібно враховувати що більшість змінних які використовуються для обрахунку ефективності побудови блокчейну, такі як обчислювальні ресурси чи потужність передачі, повинні знаходитись експериментальним шляхом. Після того як ці змінні були знайдено їх потрібно підставити в модель.

3.5 Висновок

В третьому розділі, був запропонований інтелектуалізований метод побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, який надає можливість підібрати таку популяцію мобільних вузлів, яка буде максимально ефективною для побудови блокчейну. Суть методу полягає в підборі популяції мобільних вузлів, шляхом використання алгоритму ДЕ, яка буде найбільш ефективною для виконання поставленої задачі, а саме вирішення PoW задачі і створенні таким чином нових блоків.

Окрім цього, метод враховує, що обчислювальні ресурси і якість зв’язку в мобільних пристроях можуть бути не дуже потужними величинами, і ті мобільні вузли, які не задовольняють поставленим вимогам, не беруть участь в процесі побудови блокчейну.

На відміну від аналогів, запропонований інтелектуалізований метод побудови блокчейну в мобільних розподілених системах враховує час побудови блоків та енерговитрати.

Запропонований метод характеризується наступними особливостями:

- використання такого інтелектуального методу як алгоритм ДЕ;
- використання чорного списку для зменшення додаткових обчислень з неефективними учасниками;
- обчислення на основі створеної моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат;
- збільшення ефективності підбору виконуваних над популяцією операцій, шляхом адаптації шансів використання тих чи інших операцій;
- додаткові перевірки отриманої популяції на фінальному кроці алгоритму.

Наступним кроком є реалізація апаратно-програмної частини інтелектуалізованої мобільної розподіленої системи для побудови блокчейну.

Подібна система надасть можливість дослідити ефективність запропонованого методу і провести його порівняння з відомими аналогами.

4 ІНТЕЛЕКТУАЛІЗОВАНА МОБІЛЬНА РОЗПОДІЛЕНА СИСТЕМА ДЛЯ ПОБУДОВИ БЛОКЧЕЙНУ

Для створення реалізації запропонованого методу потрібно виконати план який складається з двох пунктів:

1. Побудова апаратної частини, яка включає в себе створення мобільної розподіленої системи, яка, крім того, повинна централізованою. Також, на цьому пункті потрібно передбачити взаємозв’язки між пристроями, і імітувати вплив сторонніх пристроїв на неї.
2. Розробка програмного забезпечення для серверу, який буде центром даної розподіленої системи. Програмне забезпечення засноване на розробленому методі.
3. Проведення експериментальних досліджень, які включають в себе прогін алгоритму для мережі мобільних майнерів, і порівняння результатів з іншими відомими методами, які створені для такої задачі.

В результаті виконання цих пунктів у повинна бути створена інтелектуалізована мобільна розподілена система для побудови блокчейну. Інтелектуалізованість даної системи полягає в розробленому методі, який використовує ЕА.

Крім того, потрібно передбачити збір додаткових даних, та побудову графіків, для порівняння з іншими відомими методами, які являються аналогічними до розробленого.

4.1 Апаратна реалізація інтелектуалізованої мобільної розподіленої системи для побудови блокчейну

Апаратна частина складається з двох основних елементів, а саме, мобільних пристроїв, і сервера, який являє собою персональний комп’ютер зі встановленим на ньому розробленим програмним забезпеченням.

Окрім цих двох елементів потрібно передбачити наявність бездротового зв’язку між ними, шляхом використання Wi-Fi роутерів, чи бездротового мобільного інтернету. Отже розроблена система буде містити 3 елементи наступних типів, які додатково показані на рисунку 4.1:

1. Мобільний пристрій.
2. Сервер.
3. Wi-Fi.

Рисунок 4.1 – Позначення для використовуваних елементів: а) сервер б) wi-fi роутер в) мобільний пристрій

Як видно з рисунку 4.1, хоч сервер і являється по своїй суті персональним комп’ютером, на рисунку він відображається як сервер, що не являється помилкою, а його безпосередньою задачею.

В подальшому Wi-Fi роутер буде пропущено в побудованих схемах, але саме він використовується для поєднання інтелектуалізованої мобільної розподіленої системи для побудови блокчейну в одну мережу.

Мобільні пристрої, які були обрані для експерименту являють собою будь-які мобільні телефони, категорії смартфон, які входять в цінову категорію до 2.5 тисяч гривень. Єдиним обмеженням для даних пристроїв, вони повинні підтримувати операційну систему Android, версії, як мінімум, 8.1. Це необхідно для

того щоб можна було відправляти їм завдання для обчислення і вони його успішно виконували, без використання додаткових програмних засобів, а лише за допомогою своєї системи віддалених обчислень.

Ціна мобільних пристроїв була обрана таким чином щоб задовільти двом вимогам:

− мобільний пристрій не повинен бути надто потужним в своїх обчислювальних можливостях, оскільки суть розробленого методу полягає в саме виконанні обчислень з обмеженими обчислювальними ресурсами мобільних пристроїв;

− мобільний пристрій повинен мати хоч якісь мінімальні обчислювальні потужності і бути відносно сучасним.

Крім того, для чистоти експерименту, необхідно взяти різні мобільні пристрої, з різними обчислювальними потужностіми, і з модулями бездротового зв’язку які також відрізняються. Це дозволить зробити перевірку розробленого методу на різнотипних пристроях, оскільки пристрої, які співпадають за своїми характеристиками будуть призводити до сумнівного результату. Типового представника побудованої мережі можна побачити на рисунку 4.2.

Рисунок 4.2 – Зовнішній вигляд типового представника побудованої мобільної розподіленої мережі

Було обрано 15 мобільних пристроїв (табл.4.1), які беруть участь в побудові інтелектуалізованої мобільної розподіленої системи для побудови блокчейну.

Таблиця 4.1 – Обрані мобільні пристрої для побудови розподіленої системи

№	Назва пристрою
1	ZTE Blade L8 1/16GB Black
2	Tecno Pop 2F (B1G) 1/16GB Dual Sim Dawn Blue
3	Alcatel 1 5033D 1/8GB Dual Sim Bluish Black
4	Doogee X90 Black
5	Tecno POP 4 Pro 1/16GB Cosmic Shine
6	ZTE Blade L9 1/32GB
7	Glofiish GPad U Black
8	Blackview BV5500 Black
9	Blackview A80 2/16GB Black
10	Oukitel C19 2/16GB Black
11	Leagoo M12 2/16Gb Twilight
12	Ulefone Armor X6 Yellow
13	Blackview A10 2/16 Gold
14	DOOGEE X95 2/16GB Black
15	2E F572L 2018 DualSim Silver

Нумерація в таблиці 4.1, показує в якому порядку пристрої будуть підключені до створеної мобільної розподіленої системи. Це допоможе зв’язати результати роботи методу, з тими тими апаратними засобами які були обрані для участі в популяції.

Після того, як були обрані всі мобільні пристрої які будуть включені до складу мережі, потрібно об’єднати їх в єдину інтелектуалізовану мобільну

розподілену систему для побудови блокченйу. На рисунку 4.3 можна побачити максимально детальну картину побудованої мережі.

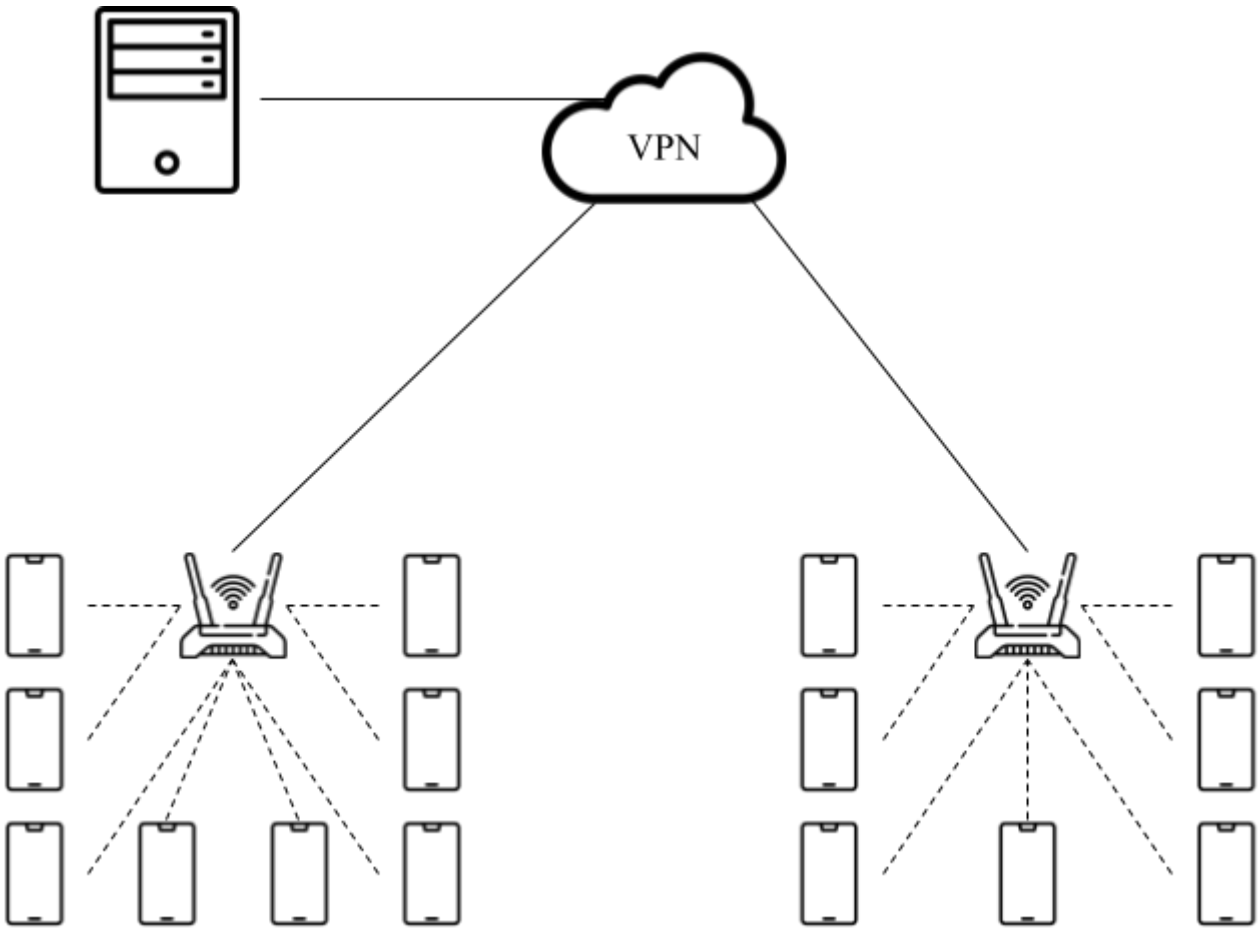


Рисунок 4.3 – Детальна структура мережі

Як показано на рисунку 4.3, побудована система, крім того що було вже перераховано вище, містить VPN-сервер. Цей сервер не виконує жодних функцій окрім об’єднання всіх пристроїв в єдину локальну мережу, саме тому його можна опустити в більш короткому представлені інтелектуалізованої мобільної розподіленої системи для побудови блокчейну(рис. 4.4).

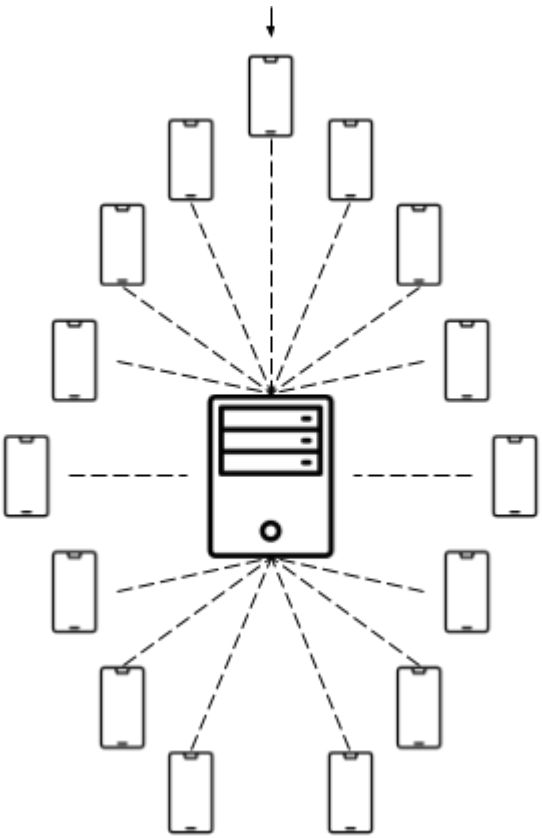


Рисунок 4.4 – Скорочена структура мережі

Єдиний момент який опущений в подібному представлені, це взаємозв’язок між мобільними пристроями, де кожен зв’язаний з кожним, але оскільки це не використовується в виконуваних обчисленнях, даний аспект мережі не відображається в представлені.

Нумерація мобільних пристроїв (рис. 4.4) відбувається починаючи з верхнього елементу, і за годинниковою стрілкою. Нумерація йде від 1 до 15 включно, і відповідає даним вказаним у таблиці 4.1.

Потрібно враховувати що в створеній системі, мобільні пристрої не являються повними вузлами, як це повинно бути в класичних майнінгових задачах, натомість завдання які виконують мобільні пристрої напряму визначаються сервером, і передбачають такі пункти як блок завдання, і початкова точка обчислень. Якщо блок завдання буде однаковий для всіх учасників майнінгового процесу, то точки для початку обчислень повинні відрізнятись, для того щоб

пристрої не дублювали свої обчислення, і таким чином не виконували зайвих операцій.

Після того, як один із вузлів завершить виконання обчислень, він повинен відправити результати на сервер, який, після перевірки, відправляє сигнал про завершення обчислень і формує нове завдання для мобільних пристроїв.

Як вже було сказано в розділі 3, важливим аспектом розробленого методу являється досягнення оптимального значення між енергоефективністю системи, і її обчислювальними потужностями.

Перевірка енерговитрат мобільних пристроїв здійснюється за рахунок моніторингу двох величин:

- відсотку використаного заряду акумулятора;
- загальних енерговитрат мережі під час виконання обчислень.

Відсоток використаного заряду акумулятора потрібно враховувати оскільки, при виконанні подібних обчислень, для мобільних пристроїв буде недостатньою та електроенергія яку вони отримують з електромережі, і більшу частину енергії буде взято саме з акумулятора для виконання подібних обчислень.

Моніторинг загальних енерговитрат в мережі необхідний тому що всі мобільні пристрої будуть під’єднані до електромережі, і будуть активно використовувати електроенергію саме звідти, потрібно також враховувати середні витрати без втручання побудованої інтелектуалізованої мобільної розподіленої системи для побудови блокчейну і врахувати отримане значення як похибку.

Потрібно врахувати що існує можливість більш енергоефективного включення мобільних пристроїв в електромережу, але в даній роботі це не являється пріоритетним.

Після того як були обрані, як загальна схема по якій будуть підключені елементи в мобільну розподілену систему для побудови блокчейну, а також схема підключення мобільних пристроїв в електромережу, можна створити і встановити додаткове ПЗ, яке знадобиться для організації подібної системи. Дане ПЗ не

являється частиною розробленого методу, а служить виключно для експлаутації інтелектуалізованої мобільної розподіленої системи для побудови блокчейну.

Використане додаткове ПЗ включає в себе:

− простий TCP-сервер призначений для передачі даних мобільним пристроям, які вважаються клієнтами, і прийом даних про завершення процесу;

− клієнт, який повинен мати змогу приймати завдання, і відправляти результат назад на сервер;

− програмний код для виконання побудови блокчейну.

Програмний код для серверної і клієнтної частини написаний за допомогою Microsoft Visual Studio. Код для вирішення PoW задачі було взято з [81], і адаптовано під потреби створеної інтелектуалізованої мобільної розподіленої системи для побудови блокчейну.

Після того, як була побудована апаратна частина, а також встановлено додаткове ПЗ, являється доцільним приступити до програмної реалізації створеного методу для інтелектуалізованої мобільної розподіленої системи для побудови блокчейну.

4.2 Програмна реалізація інтелектуалізованої мобільної розподіленої системи для побудови блокчейну

Розроблена програма повинна передбачати реалізацію запропонованого методу, а також враховувати збереження проміжних даних своєї роботи в вигляді окремих файлів, які в подальшому будуть проаналізовані в порівнянні з іншими відомими методами.

Програмна частина побудована на основі створеного інтелектуалізованого методу, мета якого підібрати максимально ефективну популяцію для мобільної розподіленої системи для побудови блокчейну.

Підбір даної популяції виконується за рахунок виконання трьох обчислювальних задач, та популяція майнерів яка виконує задачу швидше і з

меншою витратою ресурсів вважається більш ефективною, важливим пунктом даного методу в порівнянні з аналогами є врахування енерговитрат які споживають мобільні пристрої.

Створене ПЗ передбачає реалізацію певних класів (табл 4.2), з метою збільшення зрозумілості програмного коду, а також для спрощення реалізації інтелектуалізованого методу.

Таблиця 4.2 – Реалізовані класи для створеної програми

Назва Класу	Змінні Класу
Block	int nonce string info
Node	Socket socket string last_message
Pop	Node[] population double profit double electricity_used double time

Клас Block передбачає збереження лише двох змінних, а саме nonce і info. Також даний клас передбачає реалізацію інтерфейсу для роботи з файлами, з метою зчитування поля info з файлу. Варто наголосити що хоч всі блоки завдання обрані до початку процесу, і являються значеннями константними, їх збереження відбувається не безпосередньо в коді, а у файлі, в якому кожен абзац відповідає 1 МБ завдання, яке і буде записане у поле info.

Клас Node зберігає дані про кожен вузол в створеній блокчейн-мережі. Він зберігає інформацію про сокет, за допомогою якого здійснюється зв’язок між

сервером і мобільним пристроєм, а також інформацію про останнє повідомлення отримане сервером.

Клас Pop представляє собою популяцію вузлів, яка бере участь у процесі побудови блокчейну. Зберігає масив вузлів популяції, а також інформацію про ефективність даної популяції, якщо вона вже порахована. Інформація про ефективність необхідна для порівняння двох популяцій і вибору сильнішої популяції.

Окрім реалізації класів, потрібно передбачити створення глобальних змінних, які будуть описувати константні змінні, або змінні які використовуються майже на всіх етапах програми. Крім того, опис глобальних змінних дозволить краще контролювати хід роботи програми. Створенні глобальні змінні описані в таблиці 4.3.

Таблиця 4.3 – Глобальні програмні змінні

Назва змінної	Призначення
double[] const operation_probabilities	Зберігає базові значення вірогідності використання операцій вставки, видалення, або заміни по відношенню до популяції.
Node[] nodes	Зберігає популяцію всіх вузлів в створеній інтелектуалізованій мобільній розподіленій системі для побудови блокчейну.
string[] filenames	Зберігає назви всіх файлів які використовуються для збереження інформації, яку створює дане ПЗ під час своєї роботи.

Кінець Таблиці 4.3 – Глобальні програмні змінні

Назва змінної	Призначення
int const n_min	Зберігає мінімальне значення для кількості вузлів-учасників процесу побудови блокчейну.
int const maxT	Вказує максимальну кількість часу через яку сервер буде вважати задачу не виконаною даною популяцією вузлів мережі.
double const maxE	Вказує максимальну кількість енерговитрат створеної популяції, яку не потрібно перевищити.
int const iterations	Описує кількість ітерацій, яку виконує програма до отримання відповіді.
int max_BList	Вказує на максимальну кількість вузлів які зберігає чорний список.

Мінімальна кількість вузлів-учасників, повинна обиратись, як 20-30% від усієї популяції вузлів мережі. Тому було вирішено що базове значення для мінімальної кількості вузлів-учасників n_min буде обрано – 4 вузли. Тобто, 4 випадково обраних вузли будуть формувати початкову популяцію вузлів-учасників. Варто наголосити що згідно методу, число майнерів які беруть участь у процесі не може опуститись нижче числа обраної початкової популяції, і піднятись вище загальної кількості вузлів в мережі, що очевидно.

Кількість ітерацій потрібно обрати в залежності від складності виконуваних обчислень, враховуючи випадковість виконуваних операцій. Оскільки складність виконуваних обчислень дорівнює n^2, то кількість ітерацій програми було обрано, як 225 ітерацій.

Масив filenames, зберігає повний шлях до файлів, які використовуються програмою. під індексом 0 міститься файл з завданнями, а саме 3 абзаци по 1 МБ кожен, в яких містяться заздалегідь обрані значення для обрахунку. Під індексом 1 містяться шлях до папки, в якій зберігаються проміжні результати роботи програми. Під індексом 2, міститься файл який використовується для збереження фінальних даних програми.

Програмна частина передбачає створення ПЗ для побудованої інтелектуалізованої мобільної розподіленої системи для побудови блокчейну, що в свою чергу передбачає реалізацію деяких функцій (табл 4.4). Розроблені функції повинні бути максимально короткими, і дозволяють розбити створений метод на кроки.

Таблиця 4.4 – Реалізовані функції програми

Назва функції	Вхідні та вихідні дані	Призначення
Main	null	Основна функція програми, використовується як точка входу. Викликає решту функцій і організовує порядок роботи побудованої мобільної розподіленої системи для побудови блокчейну.
StartServer	Вхідні: null Вихідні: Node[]	Створює з’єднання з учасниками мережі, до тих пір поки не буде отримана команда продовжити роботу. Повертає масив всіх вузлів які підключились.
FormBasePop	Вхідні: null Вихідні: Pop	Створює базову популяцію з вузлів які містяться у масиві nodes, в кількості n_min. Повертає створену популяцію.

Продовження таблиці 4.4 – Реалізовані функції програми

Назва функції	Вхідні та вихідні дані	Призначення
SendTask	Вхідні: Block, Pop Вихідні: Block	Відправляє задачу на обрахування всім вузлам в вхідній популяції. Завершує свою роботу, як тільки один з вузлів відправляє обрахований блок завдання. Повертає обрахований блок завдання.
CheckAnswer	Вхідні: Block Вихідні: bool	Перевіряє чи отриманий блок підходить під задану PoW задачу. Дає відповідь у форматі Так/Ні
CancelAllTasks	Вхідні: Pop Вихідні: void	Відправляє сигнал всім вузлам в вхідній популяції про завершення обчислень.
MakeAnalytics	Вхідні: Pop, int Вихідні: void	Виконує обчислення ефективності для даної популяції. Використовує в обрахунках час який минув між початком функції SendTask і завершенням функції CancelAllTasks. Зберігає отримані значення у файл.
GenerateNewPop	Вхідні: ref Pop, double[] Вихідні: int	Генерує нову популяцію на основі отриманої з використанням операцій вставки, видалення або заміни. Повертає індекс використаної операції.

Кінець таблиці 4.4 – Реалізовані функції програми

Назва функції	Вхідні та вихідні дані	Призначення
ComparePops	Вхідні: ref Pop, Pop Вихідні: bool	Порівнює дві популяції і якщо початкова популяція виявилась сильнішою ніж створена, то нічого не відбувається. В іншому випадку початкова популяція замінюється сильнішою. На виході отримується інформація чи була проведена заміна.
AdaptChances	Вхідні: ref double[], int, bool Вихідні: void	На основі даних про нинішні шанси операцій, індекс виконаної операції, а також інформації про те чи ефективною виявилась використана операція, адаптує шанси проведення операцій для наступної ітерації.

Розроблені функції, передбачають не тільки основну роботу методу, але й зберігають додаткову інформацію, про проміжні етапи своєї роботи. Крім того розроблені функції являються достатньо зручними для їх модифікації в майбутньому.

Отже, беручи до уваги розроблене програмне забезпечення, яке описане в таблицях (табл. 4.2-4.4), а також в додатку А, можна сказати що алгоритм передбачає наступні аспекти своєї роботи:

1. Вибір блоку завдання, який був сформований заздалегідь до виконання основного алгоритму, і був записаний у файл, в трьох варіаціях.

2. Створення з’єднання між мобільними пристроями та сервером, шляхом налаштування сокетів по TCP протоколу.

3. Безпосередню реалізацію методу, який підбирає таку популяцію мобільних вузлів мережі, в якій вузли які беруть участь в побудові блокчейну, мають максимальну ефективність, з урахуванням енерговитрат та часу побудови блоків.

4. Запис проміжних результатів у файл, для можливості бектрекінгу ефективності роботи розробленого методу, а також номер тих ітерацій в результаті яких вони були отримані.

5. Створення запису про фінальну популяцію у файл, враховуючи її склад, і збережені значення.

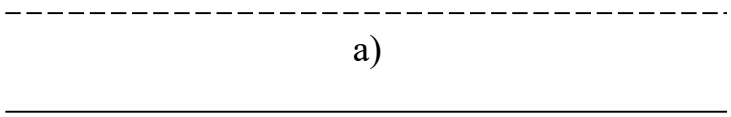
4.3 Експериментальне дослідження інтелектуалізованої мобільної розподіленої системи для побудови блокчейну

Після того, як була побудована мобільна розподелена система для побудови блокчейну, а також був програмно реалізований інтелектуальний метод для побудови блокчейну, для побудованого апаратного рішення, наступним кроком буде виконання експериментальних досліджень.

Експериментальні дослідження побудованої інтелектуалізованої мобільної розподіленої системи для побудови блокчейну передбачають виконання наступних кроків:

- створення популяції шляхом використання реалізованого методу;
- створення альтернативних популяцій, за допомогою цього ж методу. Якщо таким чином будуть отримані альтернативні популяції обрати зі створених популяцій найефективнішу, згідно заданих параметрів;
- створення альтернативних популяцій, з використанням методів-аналогів;
- порівняння результатів роботи створених популяцій між собою. Для порівняння використовуються графіки.

Обрані згідно алгоритму пристрої для участі в процесі побудови блокчейну будуть зображені суцільною лінією, якщо пристрій не був обраний для участі в процесі, то він буде поєднаний з сервером пунктирною лінією (рис 4.6).



а)

б)

Рисунок 4.5 – Зв’язки між мобільним пристроєм та сервером: а) пристрій не бере участь у процесі. б) пристрій обраний для того щоб виконувати процес.

Отже, після того як була запущена програма було отримано наступну популяцію мобільних вузлів мережі, для виконання процесу (рис. 4.6).

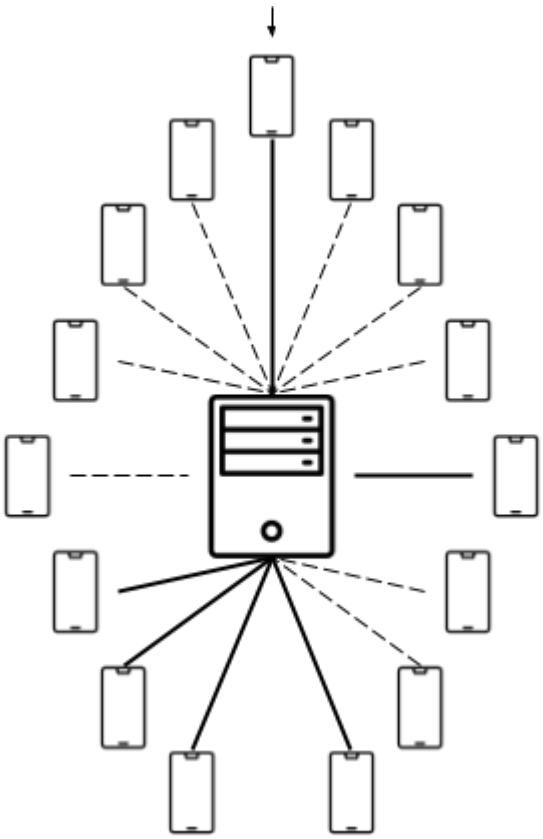


Рисунок 4.6 – Популяція мобільних вузлів отримана після роботи алгоритму.

Дана популяція вузлів, повинна давати якомога більшу швидкість побудови нового блоку, при використанні ефективної кількості ресурсів. Для цього на етапі побудови моделі (розділ 2) було створено формулу яка описує отриману системою вигоду, вигода це умовне значення, яке визначає відношення використаних ресурсів, до важливості отриманого результату, далі змінюючи коефіцієнти для обчислювальних ресурсів і енерговитрат, можна вибрати ті значення коефіцієнтів, які задовольняють поставленій задачі.

Ріст вигоди для підібраних популяцій обрахованої створеним ПЗ і представленою рисунком 4.7, можна побачити на графіку (рис.4.8)

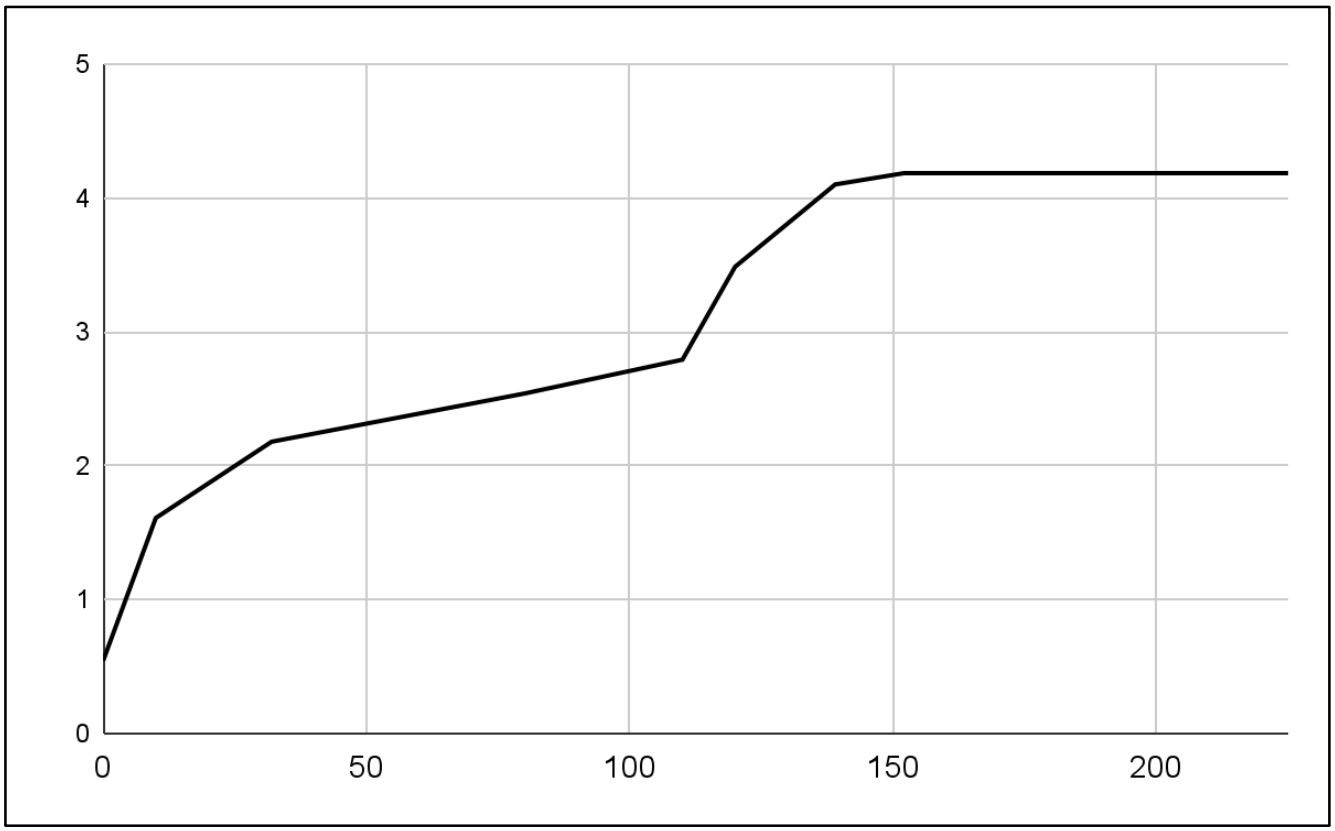


Рисунок 4.7 – Графік зміна вигоди, в залежності від кількості ітерацій

Як видно з представленого графіку, кінцевий результат був отриманий приблизно на 150-ій ітерації, і зберігався протягом решти ітерацій алгоритму.

На відміну від аналогів розроблений метод акцентує увагу на витрачених енергоресурсах, тому ріст енерговитрат, підібраних під час роботи популяцій, у ваттах, можна побачити на графіку (рис. 4.8).

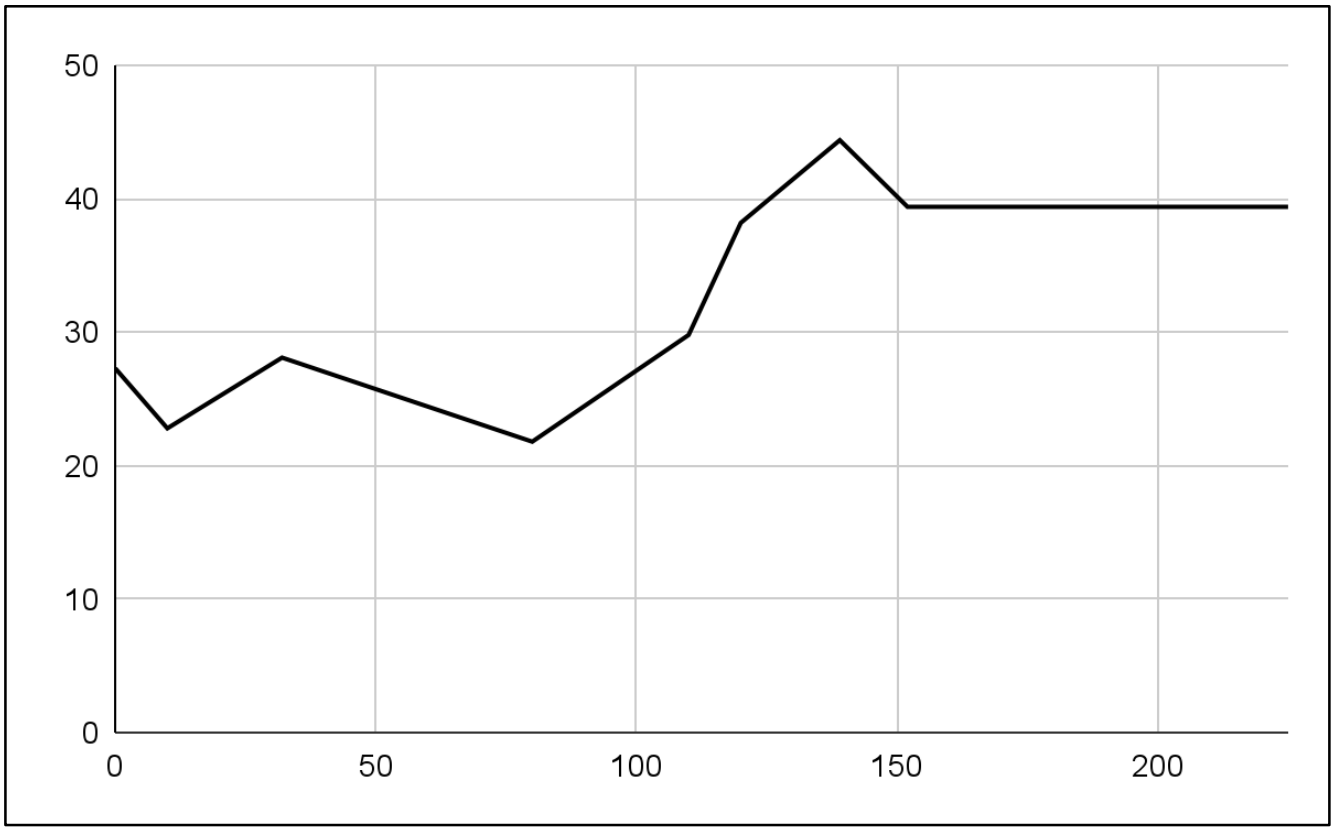


Рисунок 4.8 – Графік зміни значення енерговитрат

Час виконання процесу також є важливим значенням, оскільки саме за рахунок часу виконання процесу досягнення консенсусу в PoW задачах, визначаються загальні енерговитрати. Для того щоб представити зміну часових витрат для різних популяцій, було створено графік (рис 4.10), який показує як змінювався час виконання поставленої задачі для популяцій, під час виконання програми.

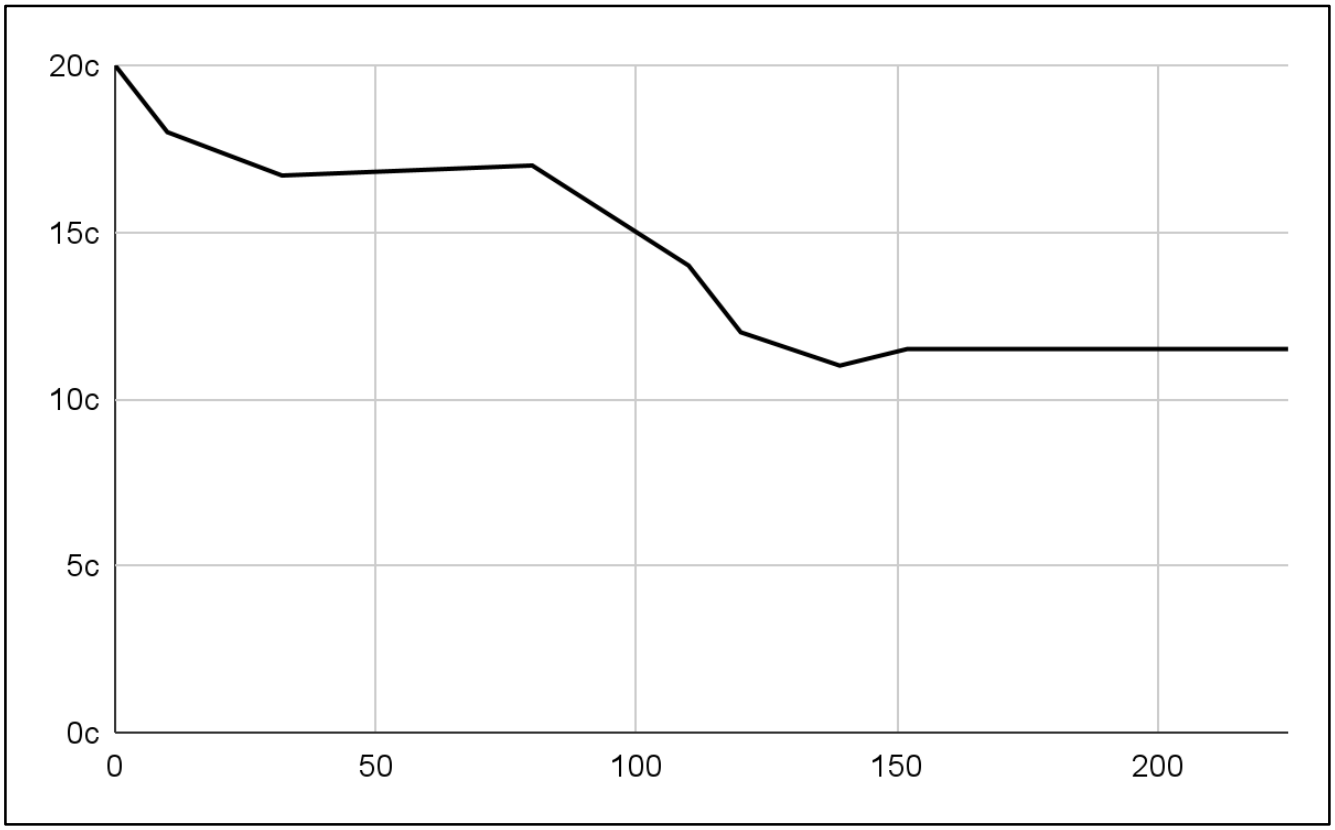


Рисунок 4.9 – Графік часу створення нового блоку, в залежності від ітерації алгоритму

Після того, як була створена популяція мобільних пристроїв, які будуть приймати участь у процесі, за допомогою запропонованого методу, необхідно провести порівняння з популяціями які отримуються за допомогою методів-аналогів.

Для виконання порівнянь було обрано наступні аналоги:

- метод заснований на підборі популяції мобільних вузлів базуючись на нейронних мережах [82] (в подальшому метод А);
- метод, який базується на використанні лише найбільш потужних пристроїв [83] (в подальшому метод Б);
- метод, який використовує інший підхід до еволюційних алгоритмів [84] (в подальшому метод В).

Використовуючи дані методи-аналоги, було сформовано три інших популяції відповідно (рис. 4.10). Після створення даних популяцій мобільної розподіленої

мережі, можна виконати тестові перевірки всіх трьох мереж, для зчитування показників. Результати збережені у вигляді діаграми (рис. 4.11)

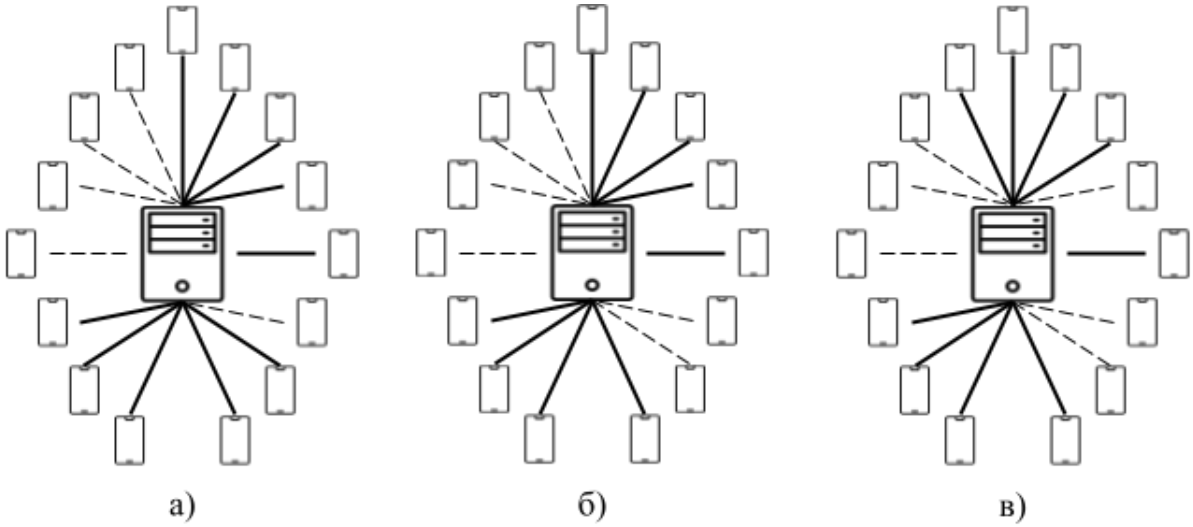


Рисунок 4.10 – Вузли підібрані методами аналогами: а) Вузли підібрані методом А б) Вузли підібрані методом Б в) Вузли підібрані методом В

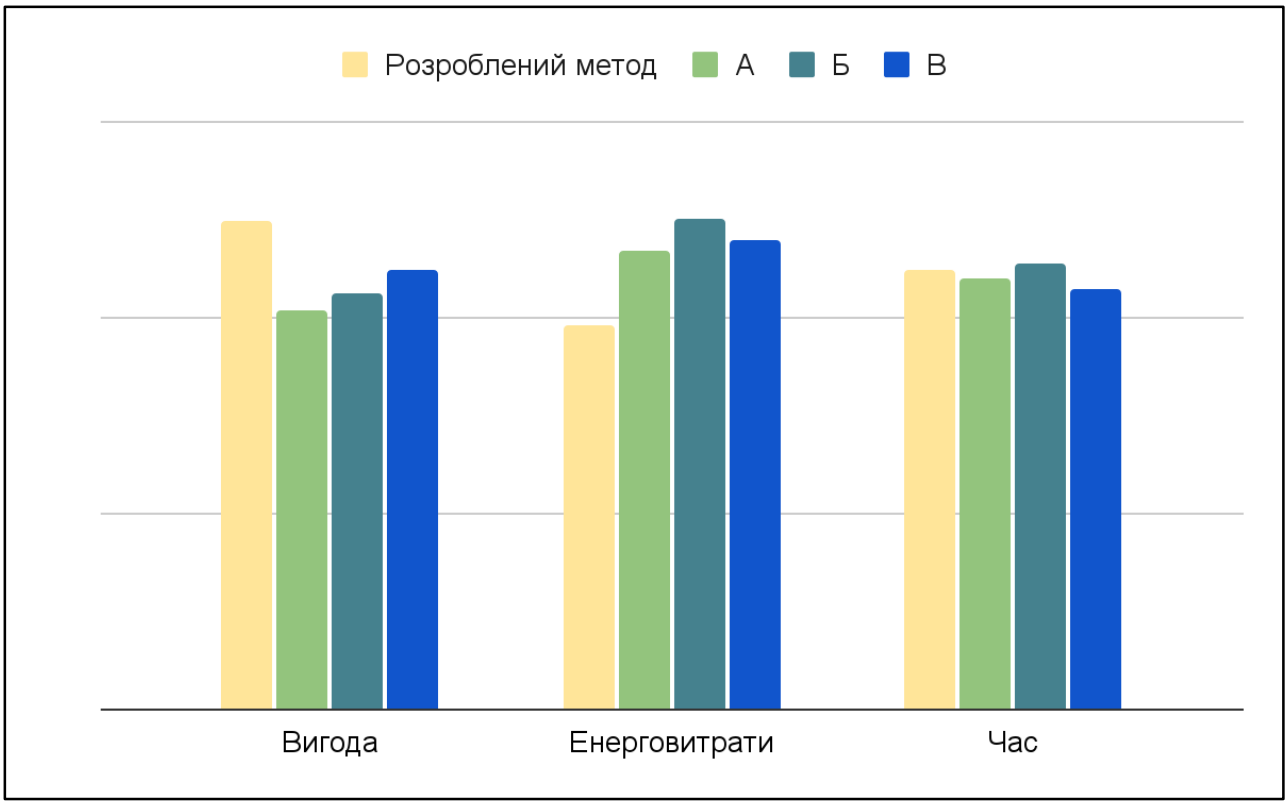


Рисунок 4.11 – Діаграма порівняння результатів роботи методів

4.4 Висновок

В цьому розділі, було розроблено апаратно-програмне рішення інтелектуалізованої мобільної розподіленої системи для побудови блокчейну.

Було підібрано 15 мобільних пристроїв, які брали участь в інтелектуалізованій мобільній розподіленій системі для побудови блокчейну. Ці пристрої було об’єднано в одну мережу за допомогою бездротового зв’язку, з використанням Wi-Fi модулів.

Таким чином було створено мобільну систему, яка є розподіленою і централізованою. Керуючий центр побудованої інтелектуалізованої мобільної розподіленої системи для побудови блокчейну знаходився на сервері. Основна програма, яка реалізує запропонований метод, також функціонувала на сервері.

Таким чином було створено реалізацію інтелектуалізованого методу, який надає можливість формування такої популяції мобільних вузлів-учасників процесу, яка досягає максимальної ефективності роботи даної системи, згідно встановлених параметрів.

Також, було проведено дослідження, яке показує результативність розробленого методу в порівнянні з аналогами. На відміну від аналогів, розроблений метод бере до уваги енергоефективність розробленої системи і показує значно менші витрати електроенергії, що збільшує ефективність побудови блокчейну на 12-20% в порівнянні з відомими аналогами.

ВИСНОВКИ

В даній роботі за результатами виконаних теоретичних та практичних досліджень було розроблено інтелектуалізований метод та апаратно-програмне рішення для збільшення ефективності побудови блокчейну в мобільних розподілених системах.

У першому розділі була досліджена предметна область, досліджені відомі методи побудови блокчейни, а також відомі методи виконання мобільних розподілених обчислень, і методів збільшення ефективності побудови блокчейну.

У другому розділі удосконалено модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, яка базується на використанні трьох інших моделей, і яка, на відміну від відомих моделей, заснована на врахуванні часу побудови блоків та енерговитратах.

Запропонована модель є основою інтелектуалізованого методу побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат.

Третій розділ представляє інтелектуалізований метод побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, який, на відміну від відомих методів, ґрунтується на створеній моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, що дозволяє підібрати максимально ефективну популяцію для побудови блокчейну згідно заданих параметрів.

У четвертому розділі, беручи за основу розроблений метод, було побудовано апаратно-програмну реалізацію інтелектуалізованої мобільної розподілної системи для побудови блокчейну, яка дозволяє підібрати найбільш ефективну популяцію мобільних вуздів, на основі даних отриманих емпіричним шляхом.

Проведено експериментальні дослідження, результати яких показали, що застосування розробленого рішення надає можливість підвищити ефективність побудови блокчейну на 12-20% в порівнянні з відомими методами.

Отже, створена інтелектуалізована мобільна розподілена система для побудови блокчейну надає можливість підібрати максимально ефективну популяцію мобільних вузлів, в залежності від їх загальної множини.

За темою дипломної роботи опубліковано статтю на тему «The Approach for IoT Malware Detection Based on Opcodes Sequences Pattern Mining» в матеріалах конференції 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021, що індексуються в наукометричній базі Scopus, а також опубліковано тези у Всеукраїнській науково-практичній конференції Актуальні Проблеми Комп'ютерних Наук (АПКН-2021). Було взято участь у Всеукраїнській науково-практичній конференції Актуальні Проблеми Комп'ютерних Наук.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Du W. D. et al. Affordances, experimentation and actualization of FinTech: A blockchain implementation study. *The Journal of Strategic Information Systems.* 2019. T. 28. №. 1. pp. 50-65.

2. Kwilinski A. Implementation of blockchain technology in accounting. *Academy of Accounting and Financial Studies Journal*. 2019. T. 23. pp. 1-6.

3. Nathan S. et al. Blockchain meets database: Design and implementation of a blockchain relational database. 2019.

4. Caro M. P. et al. Blockchain-based traceability in Agri-Food supply chain management: A practical implementation. *2018 IoT Vertical and Topical Summit on Agriculture-Tuscany (IOT Tuscany)*. 2018. pp. 1-4.

5. Lewis-Pye A., Roughgarden T. How does blockchain security dictate blockchain implementation? *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security.* 2021. pp. 1006-1019.

6. Bach L. M., Mihaljevic B., Zagar M. Comparative analysis of blockchain consensus algorithms. *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. Ieee, 2018. pp. 1545-1550.

7. Reyna A. et al. On blockchain and its integration with IoT. Challenges and opportunities. *Future generation computer systems*. 2018. T. 88. pp. 173-190.

8. Huh S., Cho S., Kim S. Managing IoT devices using blockchain platform. *2017 19th international conference on advanced communication technology (ICACT)*. IEEE, 2017. pp. 464-467.

9. Panarello A. et al. Blockchain and iot integration: A systematic survey. *Sensors*. 2018. T. 18. №. 8. pp. 2575.

10. Ometov A. et al. An overview on blockchain for smartphones: State-of-the-art, consensus, implementation, challenges and future trends. *IEEE Access*. 2020. T. 8. pp. 103994-104015.

11. Asheralieva A., Niyato D. Learning-based mobile edge computing resource management to support public blockchain networks. *IEEE Transactions on Mobile Computing*. 2019. Т. 20. №. 3. pp. 1092-1109.

12. Suankaewmanee K. et al. Performance analysis and application of mobile blockchain. *2018 international conference on computing, networking and communications (ICNC)*. IEEE, 2018. pp. 642-646.

13. Гаврилюк Р. Л., Бобровнікова К. Ю. Метод розробки емулятора виявлення кібер-загроз типу «фішинг». *Збірник наукових праць XIII Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2021»*. Хмельницький, 2021. с. 57-61.

14. The Approach for IoT Malware Detection Based on Opcodes Sequences Pattern Mining Denysiuk, D., Bobrovnikova, K., Lysenko, S., Havryliuk, R., Boichuk, Y. *Proceedings of the 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*, IDAACS 2021, pp. 779–784.

15. Nofer M. et al. Blockchain. *Business & Information Systems Engineering*. 2017. Т. 59. №. 3. pp. 183-187.

16. Catalini C., Gans J. S. Some simple economics of the blockchain. *Communications of the ACM*. 2020. Т. 63. №. 7. С. 80-90.

17. Ceri S., Pernici B., Wiederhold G. Distributed database design methodologies. *Proceedings of the IEEE*. 1987. Т. 75. №. 5. pp. 533-546.

18. Kushch S., Prieto-Castrillo F. Blockchain for dynamic nodes in a smart city. *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*. IEEE, 2019. С. 29-34.

19. Deepa N. et al. A survey on blockchain for big data: approaches, opportunities, and future directions. *Future Generation Computer Systems*. 2022.

20. Pilkington M. Blockchain technology: principles and applications. 2016.

21. Mermer G. B., Zeydan E., Arslan S. S. An overview of blockchain technologies: principles, opportunities and challenges. *2018 26th Signal Processing and Communications Applications Conference (SIU)*. IEEE, 2018. pp. 1-4.

22. Mukkamala R. R. et al. Blockchain for social business: Principles and applications. *IEEE Engineering Management Review*. 2018. Т. 46. №. 4. pp. 94-99.

23. Di Pierro M. What is the blockchain? *Computing in Science & Engineering*. 2017. Т. 19. №. 5. pp. 92-95.

24. Maxwell D., Speed C., Pschetz L. Story Blocks: Reimagining narrative through the blockchain. *Convergence*. 2017. Т. 23. №. 1. pp. 79-97.

25. Zheng Z. et al. An overview of blockchain technology: Architecture, consensus, and future trends. *2017 IEEE international congress on big data (BigData congress)*. Ieee, 2017. pp. 557-564.

26. Monrat A. A., Schelén O., Andersson K. A survey of blockchain from the perspectives of applications, challenges, and opportunities. *IEEE Access*. 2019. Т. 7. pp. 117134-117151.

27. Feng J. et al. Towards random-honest miners selection and multi-blocks creation: Proof-of-negotiation consensus mechanism in blockchain networks. *Future Generation Computer Systems*. 2020. Т. 105. pp. 248-258.

28. Schilling L., Uhlig H. Some simple bitcoin economics. *Journal of Monetary Economics*. 2019. Т. 106. pp. 16-26.

29. Dannen C. Introducing Ethereum and solidity. 2017. Т. 1. – pp. 159-160.

30. Gibbs T., Yordchim S. Thai perception on Litecoin value. *International Journal of Social, Behavioral, Educational, Economic, Business and Industrial Engineering*. 2014. Т. 8. №. 8. pp. 2613-5.

31. . Gupta S., Sadoghi M. Blockchain transaction processing. 2021.

32. Peters G. W., Panayi E. Understanding modern banking ledgers through blockchain technologies: Future of transaction processing and smart contracts on the internet of money. *Banking beyond banks and money*. Springer, Cham, 2016. pp. 239-278.

33. . Jabbar A., Dani S. Investigating the link between transaction and computational costs in a blockchain environment. *International Journal of Production Research*. 2020. Т. 58. №. 11. pp. 3423-3436.

34. Wang Y., Kogan A. Designing confidentiality-preserving Blockchain-based transaction processing systems. *International Journal of Accounting Information Systems*. 2018. T. 30. pp. 1-18.

35. Nair R. et al. An approach to minimize the energy consumption during blockchain transaction. *Materials Today: Proceedings*. 2020.

36. Niranjanamurthy M., Nithya B. N., Jagannatha S. Analysis of Blockchain technology: pros, cons and SWOT. *Cluster Computing*. 2019. T. 22. №. 6. pp. 14743-14757.

37. Bhosale J., Mavale S. Volatility of select crypto-currencies: A comparison of Bitcoin, Ethereum and Litecoin. *Annu. Res. J. SCMS, Pune*. 2018. T. 6.

38. Radanović I., Likić R. Opportunities for use of blockchain technology in medicine. *Applied health economics and health policy*. 2018. T. 16. №. 5. pp. 583-590.

39. Roman-Belmonte J. M., De la Corte-Rodriguez H., Rodriguez-Merchan E. C. How blockchain technology can change medicine. *Postgraduate medicine*. 2018. T. 130. №. 4. pp. 420-427.

40. Siyal A. A. et al. Applications of blockchain technology in medicine and healthcare: Challenges and future perspectives. *Cryptography*. 2019. T. 3. №. 1. C. 3.

41. Esposito C. et al. Blockchain: A panacea for healthcare cloud-based data security and privacy? *IEEE Cloud Computing*. 2018. T. 5. №. 1. pp. 31-37.

42. Zhang X., Chen X. Data security sharing and storage based on a consortium blockchain in a vehicular ad-hoc network. *IEEE Access*. 2019. T. 7. pp. 58241-58254.

43. Wang X. et al. Survey on blockchain for Internet of Things. *Computer Communications*. 2019. T. 136. pp. 10-29.

44. Ge C., Liu Z., Fang L. A blockchain based decentralized data security mechanism for the Internet of Things. *Journal of Parallel and Distributed Computing*. 2020. T. 141. pp. 1-9.

45. Kshetri N., Voas J. Blockchain-enabled e-voting. *Ieee Software*. 2018. T. 35. №. 4. pp. 95-99.

46. Raikwar M., Gligoroski D., Velinov G. Trends in development of databases and blockchain. *2020 Seventh International Conference on Software Defined Systems (SDS)*. IEEE, 2020. pp. 177-182.

47. Gammon K. Experimenting with blockchain: Can one technology boost both data integrity and patients' pocketbooks? *Nature Medicine*. 2018. Т. 24. №. 4. pp. 378-382.

48. Goede M. E-Estonia: The e-government cases of Estonia, Singapore, and Curaçao. *Archives of Business Research*. 2019. Т. 7. №. 2.

49. Стартап Moni. URL: https://www.eu-startups.com/directory/moni/ (дата звернення 20.01.2022).

50. Blockchain White Paper. URL: http://www.caict.ac.cn/english/yjcg/bps/201901/t20190131_194150.htm (дата звернення 20.01.2022).

51. Zhou Q. et al. Solutions to scalability of blockchain: A survey. *Ieee Access*. 2020. Т. 8. pp. 16440-16455.

52. Kiayias A. et al. Ouroboros: A provably secure proof-of-stake blockchain protocol. *Annual international cryptology conference*. Springer, Cham, 2017. pp. 357-388.

53. Zheng Z. et al. Blockchain challenges and opportunities: A survey. *International Journal of Web and Grid Services*. 2018. Т. 14. №. 4. pp. 352-375.

54. Eyal I., Sirer E. G. Majority is not enough: Bitcoin mining is vulnerable. *International conference on financial cryptography and data security*. Springer, Berlin, Heidelberg, 2014. pp. 436-454.

55. Nayak K. et al. Stubborn mining: Generalizing selfish mining and combining with an eclipse attack. *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2016. pp. 305-320.

56. Sapirshtein A., Sompolinsky Y., Zohar A. Optimal selfish mining strategies in bitcoin. *International Conference on Financial Cryptography and Data Security*. Springer, Berlin, Heidelberg, 2016. pp. 515-532.

57. Kogias E. K. et al. Enhancing bitcoin security and performance with strong consistency via collective signing. *25th usenix security symposium (usenix security 16)*. 2016. pp. 279-296.

58. Castro M. et al. Practical byzantine fault tolerance. *OsDI*. 1999. T. 99. №. 1999. pp. 173-186.

59. Zamani M., Movahedi M., Raykova M. Rapidchain: Scaling blockchain via full sharding. *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. 2018. pp. 931-948.

60. Wang J., Wang H. Monoxide: Scale out blockchains with asynchronous consensus zones. *16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19)*. 2019. pp. 95-112.

61. Corbett J. C. et al. Spanner: Google's globally distributed database. *ACM Transactions on Computer Systems (TOCS)*. 2013. T. 31. №. 3. pp. 1-22.

62. Lombrozo E., Lau J., Wuille P. Segregated witness (consensus layer). *Bitcoin Core Develop. Team, Tech. Rep. BIP*. 2015. T. 141.

63. Javarone M. A., Wright C. S. From Bitcoin to Bitcoin Cash: a network analysis. *Proceedings of the 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems*. 2018. pp. 77-81.

64. Block Size Increase. URL: https://bitfury.com/content/downloads/block-size-1.1.1.pdf (дата звернення 20.01.2022).

65. Ding D. et al. Txilm: Lossy block compression with salted short hashing. *arXiv preprint arXiv:1906.06500*. 2019.

66. Bentov I., Pass R., Shi E. Snow White: Provably Secure Proofs of Stake. *IACR Cryptol. ePrint Arch*. 2016. T. 2016. №. 919.

67. Eyal I. et al. {Bitcoin-NG}: A Scalable Blockchain Protocol. *13th USENIX symposium on networked systems design and implementation (NSDI 16)*. 2016. pp. 45-59.

68. Yang F. et al. Delegated proof of stake with downgrade: A secure and efficient blockchain consensus algorithm with downgrade mechanism. *IEEE Access*. 2019. T. 7. pp. 118541-118555.

69. Ongaro D., Ousterhout J. In search of an understandable consensus algorithm. *2014 USENIX Annual Technical Conference (Usenix ATC 14)*. 2014. pp. 305-319.

70. Schwartz D. et al. The ripple protocol consensus algorithm. *Ripple Labs Inc White Paper*. 2014. T. 5. №. 8. P. 151.

71. P4Titan. Slimcoin: A Peer-To-Peer Crypto-Currency with Proof-of-Burn. – 2014.

72. Bentov I. et al. Proof of activity: Extending bitcoin's proof of work via proof of stake [extended abstract] y. *ACM SIGMETRICS Performance Evaluation Review*. 2014. T. 42. №. 3. pp. 34-37.

73. Kwon J. Tendermint: Consensus without mining. *Draft v. 0.6, fall*. 2014. T. 1. №. 11.

74. Mazieres D. The stellar consensus protocol: A federated model for internet-level consensus. *Stellar Development Foundation*. 2015. T. 32. pp. 1-45.

75. Luu L. et al. A secure sharding protocol for open blockchains. *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. 2016. pp. 17-30.

76. Miller A. et al. The honey badger of BFT protocols. *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. 2016. pp. 31-42.

77. Duong T. et al. 2-hop blockchain: Combining proof-of-work and proof-of-stake securely. *European Symposium on Research in Computer Security*. Springer, Cham, 2020. pp. 697-712.

78. Zhang Z. W. A byzantine fault-tolerant algorithm for blockchains. – 2019.

79. Wang T., Liew S. C., Zhang S. When blockchain meets AI: Optimal mining strategy achieved by machine learning. *International Journal of Intelligent Systems*. 2021. T. 36. №. 5. pp. 2183-2207.

80. Liu J., Lampinen J. A fuzzy adaptive differential evolution algorithm. Soft Computing. 2005. T. 9. №. 6. pp. 448-462.

81. Bcoin URL: https://github.com/bcoin-org/bcoin (дата звернення 23.03.2022).

82. Goel A. et al. DeepRing: Protecting deep neural network with blockchain. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 2019.

83. Jiang S. et al. Blockchain competition: The tradeoff between platform stability and efficiency. *European Journal of Operational Research*. 2022. Т. 296. №. 3. С. 1084-1097.

Додаток А

(обов’язковий)

Лістинг коду програмного забезпечення інтелектуалізованої мобільної розподіленої системи для побудови блокчейну (серверна частина)

```
using System;
using System.IO;
using System.Linq;
using System.Collections;
using System.Timers;
using System.Net;
using System.Net.Sockets;
using System.Text;

public class Pop
{
    public Node[] population;
    public double profit;
    public double electricity_used;
    public double time;

}

public class Node
{
    public Socket socket;
    public string last_message;

}

public class Block
```

```
{
   public int nonce;
   public string info;
}

double[] const operation_probabilities = { 0.5, 0.3, 0, 2 };
Node nodes = new Node[n];
string[] filenames = { @"C:\magstr\задача.txt", @"C:\magstr\Extra\",
@"C:\magstr\результат.txt" };
int const n = 15;
int const n_min = 4;
int const maxT = 30000;
double const maxE = 100;
int const iterations = n * n;
int max_BList = n_min;
Random rand = new Random(Time.Now);

public class mgstrprgrm
{
   public static socket formSocket()
   {
      int recv;
      byte[] data = new byte[1024];
      IPEndPoint ipep = new IPEndPoint(IPAddress.Any, 9050);
      Socket newsock = new Socket(AddressFamily.InterNetwork, SocketType.Stream,
ProtocolType.Tcp);
      newsock.Bind(ipep);
      newsock.Listen(10);
      Console.WriteLine("Очікування...");
      Socket client = newsock.Accept();
```

```
    return client;
}

public static Node[] StartServer()
{
    Node[] nodes1 = new Node[n];
    for (int i = 0; i < n; i++)
        nodes1[i].socket = formSocket();
    return nodes1;
}

public static Pop FormBasePop()
{
    Pop pop = new Pop()
    for (int i = 0; i < n_min; i++)
        pop += Node[rand.Next(0, n)];
}

public static bool SendTask(Block block)
{
    if (SHA256(block.info + block.nonce)) return true;
    return false;
}

public static bool CancelAllTasks(Block block, Pop pop)
{
    foreach (Node node in pop)
    {
        node.sendMessage("Відмінити");
    }
```

```
}

public static Block SendTask(Block block, Pop pop)
{
   string message;
   Timer timer = new Timer();
   timer.Start();
   foreach(Node node in pop)
   {
      node.sendMessage(block);
   }
   foreach (Node node in pop)
   {
      message = node.socket.Listen();
      if (messessage != "")
      {
         node.last_message = message;
         if (CheckAnswer(new Block(message))) break;
      }
   }
   timer.Stop();
   pop.time = timer.Time;
   Console.ReadLine(pop.electricity_used);
   CancelAllTasks(block, pop);
   return block;
}
public static pop GenerateNewPop(Pop pop, double[] op)
{
   double chance = rand.NextDouble(0, 1);
   if (chance < op[0]) pop.add(rand.Next(0, 15));
```

```
    else if (chance < op[0] + op[1]) pop.extract(rand.Next(0, 15));
    else pop.switch(rand.Next(0, 15), rand.Next(0, 15));
    return pop;
}
public static pop ComparePops(ref Pop pop1, Pop pop2)
{
    if (pop1.profit > pop2.profit) retutn;
    pop1 = pop2;
}
public static void MakeAnalytics(Pop pop)
{
    pop.profit = 0.28 * pop.time - 0.12 * pop.electricity_used;
}
public static void Main()
{
    List<Node> BList = new List<Node>();
    double[] op = operation_probabilities;
    nodes = StartServer();
    Pop basePop = FormBasePop();
    SendTask(basePop, blocks);
    MakeAnalytics(basePop);

    for(int i = 0; i < iterations; i++)
    {
        Pop mutantPop = GenerateNewPop(basePop);
        SendTask(ref mutantPop, blocks);
        MakeAnalytics(mutantPop);
        AdaptChances(ref op, ComparePops(basePop, mutantPop));
        basePop.WriteToFile(filenames[1]);
    }
```

```
        Console.WriteLine(basePop);
    }
}
```

Додаток Б

(обов'язковий)

Копія публікації у виданні, що індексується в наукометричній базі Scopus

The 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications
22-25 September, 2021, Cracow, Poland

The Approach for IoT Malware Detection Based on Opcodes Sequences Pattern Mining

Dmytro Denysiuk [1], Kira Bobrovnikova [1], Sergii Lysenko [1], Oleg Savenko [1], Piotr Gaj [2], Roman Havryliuk[1], Yaroslav Boiko[1]

[1] Khmelnytskyi National University, Institutska str., 11, Khmelnytskyi, 29016, Ukraine,
web.developer.den@gmail.com, bobrovnikova.kira@gmail.com, sirogyk@ukr.net, savenko_oleg_st@ukr.net,
http://ki.khnu.km.ua/

[2] Silesian University of Technology, Akademicka str. 2A, Gliwice, 44-100, Poland,
piotr.gaj@polsl.pl

***Abstract* – In recent years, the area of IoT malware detection has been at the forefront of the security community. Due to lack the latest cybersecurity requirements in IoT devices design, lack of security updates and transparency of security posture, as well as due to the IoT devices specific characteristics, such as heterogeneity of the processor architecture, unsafe deployment of IoT devices, as well as exponential growth in the number of IoT devices - all this leads to an increase in the number of malware targeting IoT devices.**

The paper presents the new approach for IOT malware detection based on opcodes (operation codes) sequences pattern mining. The proposed approach uses mining of maximal sequential patterns of operation codes from assembly representation of IoT binary executable. Most distinctive of maximal sequential patterns are selected and for each of them estimations of their relevance are performed. Based on these relevance the feature vectors are built which described assembly representation of IoT binary executable. As a classifier Fuzzy Semi-Supervised C-Means classifier was applied.

The experimental results shows that the proposed approach allows detecting IOT malware with height effectiveness, with accuracy 99,51% and F_1-score at level 99,50%.

Keywords — Internet of Things; malware; detection of malicious software; sequential pattern mining; opcodes analysis

I. Introduction

Today, the Internet of Things has become an important part of modern society and has enormous potential, providing new services needed in everyday life. Every year, new types of devices appear on the IoT device market. The main weakness of these devices is the low level of protection against malicious software and cyberattacks.

Smart devices infected with malicious software can be used to steal a user's personal data or become a source of cyberattacks on other devices on the same network or outside it. A compromised smart device can also affect other critical components in the same network, such as database servers and Intranet, by being able to collect data and monitor other components in the IoT network [1, 2].

The projected rapid growth in the number of IoT devices [3] and the lack of basic monitoring and protection mechanisms for them will continue to contribute to the development and spread of malicious software in the IoT infrastructure.

II. Related Works

Today a number of works on the detection of IoT malware based on the analysis of operating codes are known.

In the study [4] for IoT malware detection and categorization fast fuzzy and fuzzy pattern tree approaches were applied. With this aim IoT programs operation codes were transmute into a vector space.

In the paper [5] a technique based on opcode N-grams analysis to classifying IoT malware was developed. With aim to detection IoT malware TF-IDF, Term frequency-Inverse document frequency for each of N-grams for IoT program is calculated. The feature vectors from obtained TF values for IoT programs are constructed. The machine-learning methods were used to carry out IoT malware classification.

In [6] an approach for malware detection in infrastructure of Internet Of Battlefield Things. This deep learning based approach applies class-wise selection of operation codes sequences as a signs for future classification. These operation codes are transformed into a vector space. Further for each IoT application a graph of selected features was build. To classify IoT benign applications and IoT malware a deep Eigen space learning technique was used.

In the work [7] combining machine learning techniques with the sequential pattern mining algorithm to find most frequent operation codes sequences of IoT malware were applied.

In the paper [8] a multi-view learning approach that applies multiple views such as operation codes, bytecodes, API calls, header information, permission and attacker's intent for IoT malware detection. The proposed approach presupposes automatically definition different weights to

these views for IoT malware detection in different environment.

The work [9] is devoted to IoT malicious programs detection by using the evolutionary algorithm and operation codes analysis. According to this technique the suspicious IoT program is defined based on the most similar graph which is built by using evolutionary algorithm for IoT malware families and IoT benign programs.

In the work [10] a malware-detection approach based on convolutional recurrent neural network which uses opcode sequences analysis is presented. In order to extract opcode sequences from executables, the proposed algorithm is applied. In fine for malware detection a deep learning method based on opcode sequences analysis was proposed. According to this approach an convolutional autoencoder transforms a long opcode sequences to a compressed short sequences. A dynamic classifier performs a prediction task using the obtained sequences.

In [11] a cross-architecture IoT malware detection technique which uses advanced ensemble learning. This technique using stacked ensemble of neural networks, such as recurrent (RNN) and convolutional (CNN) neural network and operation codes based feature selection algorithms for IoT malware detection.

In research [12] RNN was used for the study of process of codes execution in IoT frameworks based on ARM platform. Next, this approach was tested with purpose of analysis of operation codes via Long Short Term Memory (LSTM). It was concluded that LSTM-based solution provides the best result compared with other machine learning methods.

In the paper [13] an approach, analogical to processing of natural language, was proposed. This approach uses modeling of malware as a language to perform static analysis of operation code sequence patterns of malware. To constructing the feature vectors of operation codes the word embedding technique is used. Next a two-stage LSTM malware detection model was used.

Despite the large amount of various approaches for malware detection [14-16] as well as different newel approaches for data analysis [17-22], IoT devices are exceedingly defenseless against the malware infection and their negate consequences. Therefore, there is a need to deploy new methods for detecting IoT malware.

III. The Approach for IoT Malware Detection Based on Opcodes Sequences Pattern Mining

The proposed approach for malicious IoT malware detection uses the analysis of the IoT programs operation codes sequences to further improve the accuracy of IoT malware identification. This approach consists of two stages: training stage and detection stage. The steps of the training stage are presented below:

1. Processing the samples of IoT malicious and benign binary executable and extracting assembly representation from these samples.
2. Constructing the feature vectors based on maximal sequential patterns of operation codes:
   - mining of maximal sequential patterns of operation codes for the all samples of assembly representation;
   - selecting most distinctive of maximal sequential patterns of operation codes and determination the order for each of them as features in the feature vectors;
   - estimation of the relevance for each of the selected maximal sequential patterns of operation codes;
   - constructing feature vectors from estimation values of the relevance of obtained maximal sequential patterns.
4. Building labelled and unlabeled data matrix of feature vectors.
5. Performing semi-supervised learning of the fuzzy c-means classifier with using the labelled data matrix.
6. Testing of the fuzzy c-means classifier with using unlabeled data matrix.
7. Evaluating of the effectiveness of the proposed approach for IoT malware detection based on opcodes sequences pattern mining.

Let us describe the some steps of the training stage of the proposed approach for IoT malware detection based on opcodes sequences pattern mining.

A. Constructing the Feature Vectors Based on Maximal Sequential Patterns of Operation Codes

At the training stage of the proposed approach for examples of IoT malicious and benign binary executable their assembly representations are extracted. From obtained assembly representations the maximal sequential patterns of operation codes are mining for the all samples of assembly representation by applying of sequential patterns algorithm.

At the next step to construct the feature vectors based on maximal sequential patterns of operation codes the order must be determined for each of features in the feature vectors. For this purpose, the following actions are performed. For each of obtained maximal sequential patterns the values of inverse document frequency, IDF, are calculated. Taking into account the IDF values reduces the weight of commonly used maximal sequential patterns and allows selecting most distinctive of them:

$$IDF(MP, E) = \log \frac{|E|}{|\{e_i \in E \mid MP \in e_i\}|}, \qquad (1)$$

where $|E|$ – the total number of the assembly representation of IoT binary executables e, $e \in E$, $E = E_b \cup E_m$, were E_b – the set of assembly representation of benign IoT binary executables, E_m –

the set of assembly representation of malicious IoT binary executables;

$|\{e_i \in E \mid MP \in e_i\}|$ – the number of the assembly representation of IoT binary executables e, in which appears certain maximal sequential pattern MP.

Next, for each of features in the feature vectors the order is determined according to the ascending values of the $IDF(MP,E)$:

$$\Theta = \left\{IDF(MP_i,E)\right\}_{i=1}^{N_{MP}},\ IDF(MP_i,E) < IDF(MP_{i+1},E), \quad (5)$$

where N_{MP} – the total number of different MP.

At the next step the relevance value for each MP estimated as weighted term frequency WTF [23]. WTF is calculated as the result of weighting the term frequency, TF, with the relevance of each operation code o when calculating TF, and is defined as the product of sequence frequency and the weight of every operation code o in MP:

$$WTF(MP,e) = TF(MP,e) \times \prod_{o \in MP} \frac{W(o)}{100}, \quad (2)$$

where $W(o)$ – the weight, by means of mutual information gain, for the operation code o;

$TF(MP,e)$ – the MP frequency measure within the assembly representation of IoT binary executable e.

Term frequency, $TF(MP,e)$, assessed the importance of a MP within an assembly representation of IoT binary executable e and can be calculated as following:

$$TF(MP,e) = \frac{f_{MP,e}}{\sum_{MP' \in e} f_{MP',e}}, \quad (3)$$

where $f_{MP,e}$ – the number of times of the maximal sequential pattern MP appears in the assembly representation of IoT binary executable e;

$\sum_{MP' \in e} f_{MP',e}$ – the total number of different MP in the assembly representation of IoT binary executable e.

Calculation the weight for the operation code o, $W(o)$, is based on the concept of Mutual Information. The Mutual Information $I(F;\Upsilon)$ is measure of the statistical dependence of certain two variables. In our case these variables are the opcode o and whether or not the IoT binary executable was malicious program:

$$I(F;\Upsilon) = \sum_{\gamma \in \Upsilon} \sum_{f \in F} p(f,\gamma) \log\left(\frac{p(f,\gamma)}{p(f) \times p(\gamma)}\right), \quad (4)$$

where F – the frequency of operation code;

Υ – the class of the IoT binary executables (benign or malware);

$p(f,\gamma)$ – the joint probability distribution function of F and Υ;

$p(f)$ – the marginal probability distribution function of F;

$p(\gamma)$ – the marginal probability distribution function of Υ.

Thus, the relevance values of MP are used as features of the feature vector which describes assembly representation of IoT binary executable based on maximal sequential patterns of operation codes. In case where the MP was not presented in the assembly representation of IoT binary executable, then the corresponding feature takes the value 0. Let us denote feature vector as:

$$\overline{V_{d,e}} = (r_i)_{i=1}^{N_{MP_e}}, \quad (5)$$

where $d \in D$ – the IoT device in the IoT network, $D = \{d_i\}_{i=1}^{N_d}$ – the set of IoT devices in the IoT network;

N_d – the amount number of IoT devices in the network;

r_i – the relevance value of MP;

N_{MP_e} – the number of MP in the assembly representation of IoT binary executable e.

B. Building Labelled and Unlabeled Data Matrix of Feature Vectors

From the constructed feature vectors $\overline{V_{d,e}}$ two matrix, a labelled data matrix M_l (with feature vectors labelled as "benign" or "malicious" respectively) and unlabeled data matrix M_{unl} (with unlabeled feature vectors) were built. These feature vectors $\overline{V_{d,e}}$, whose lengths are greater than the median length L of the rest all feature vectors in the data matrix M_{unl} and M_l, are truncated. Feature vectors, whose lengths are less than L, are padded with zeros. Further the labelled data matrix M_l is used as training data, and unlabeled data matrix M_{unl} is used as testing data.

The scheme of the process of constructing the feature vectors based on maximal sequential patterns of operation codes and building labelled and unlabeled data matrix presented in Fig. 1.

C. Data Classification

As a classifier in proposed technique the Semi-Supervised Fuzzy C-Means classifier was used. The benefit of applying the fuzzy clustering is the ability to reduce the requirements for the unambiguous correspondence of the clustering object to a specific

cluster. Fuzzy clustering apply functions of membership clustering objects to the fuzzy clusters which take values in the interval $[0,1]$. It allows increasing the information completeness of the results of clustering in cases where clustering object is located at the boundaries of the different clusters.

The distinctive particularity and main advantage of semi-supervised learning is the ability of identification of the initial centers of clusters. This feature improves the quality of clustering results. At the same time, to determine the initial centers of clusters, a training sample is required, the volume of which does not exceed 10% of the data collected for analysis.

Instead of the Euclidean metric, which is applied in the basic algorithm, it was decided to choose the Mahalanobis distance. The use of the Mahalanobis distance allows to take into account the presence of outliers, or observations, that stand out from the general sample in the classified data. This is possible due to the formation of clusters in the form of hyperellipsoids, whose axes can be oriented in different directions. Thus, a fuzzy partition matrix P is the result of clustering, and each element p_{ij} of the matrix P determines the belonging degree of the i -th clustering objects to the j -th cluster:

$$P=\left[p_{ij}\right], p_{ij}\in[0,1], \quad i=\overline{1,N_{\overline{V_{d,e}}}}, j=\overline{1,N_{\Upsilon}},$$

$\sum_{j=\overline{1,N_{\Upsilon}}} p_{ij}=1$, where $N_{\overline{V_{d,e}}}$ – the amount number of the feature vectors, N_{Υ} – the amount number of the clusters. So, the applying of fuzzy clustering implies that each feature vector $\overline{V_{d,e}}$ with a certain affiliation degree will be referred to each of the N_{Υ} clusters, which denotes malicious or benign IoT binary executable.

Let us denote the set of clusters as $\Upsilon=\Upsilon_b\cup\Upsilon_m$, where Υ_b is a cluster that correspond to benign class (benign IoT binary executables) and Υ_m is a cluster that correspond to malicious class (malicious IoT binary executables).

Let's take λ as the value of threshold that determines whether clustering object belonging to the certain cluster, at which the clustering object is considered as benign or malicious. If $p_{ij}\geq\lambda$, then the clustering object belongs to a j cluster, $j\in\{b,m\}$.

The scheme of the proposed approach for IoT malware detection based on opcodes sequences pattern mining presented in Fig. 2.

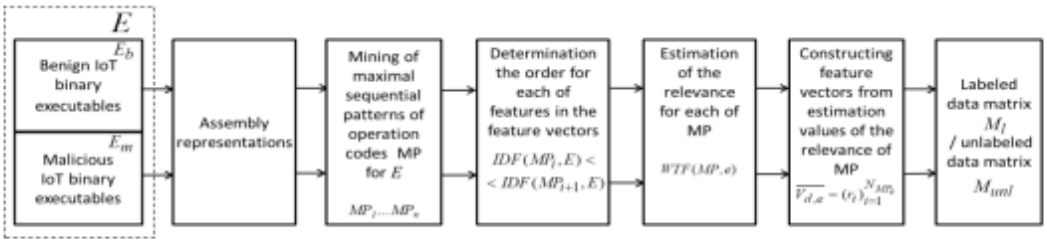


Figure 1. Constructing the feature vectors based on maximal sequential patterns of operation codes and building labelled and unlabeled data matrix

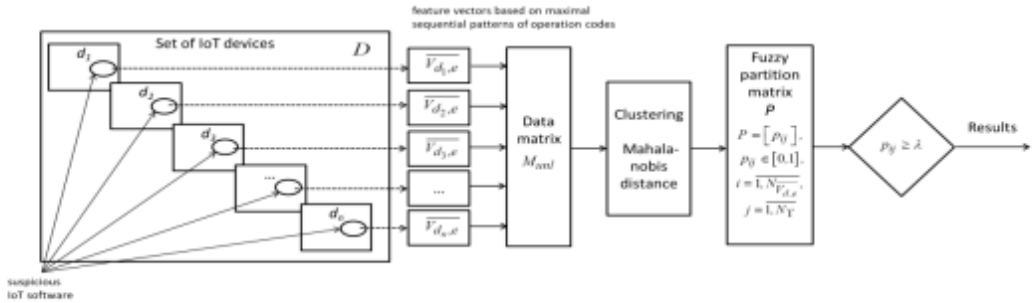


Figure 2. The scheme of the proposed approach for IoT malware detection based on opcodes sequences pattern mining

IV. Experimental Results

In order to evaluate the effectiveness of the proposed approach, a number of experiments were conducted. To conduct the experiment, it was necessary to choose a target IoT platform. For this purpose ARM platform was chosen as it is one of the most widespread IoT platforms.

In this research we have presented the new approach for IOT malware detection based on operation codes sequences pattern mining. Concerning the experimental results, we have used two sets of samples: [24] resource present the set of programs for IoT devices (benign programs for our experiments), while [25] presents the resource with a huge amount of the malware including malware for IoT devices. At this stage of research, the main purpose of the work was not to form the representative of software, but to verify the possibility of developed technique to detect IoT malware.

Thus 314 samples of benign ARM-based IoT software samples from [24] for such IoT devices as camcorders, smart TVs and routers and 250 ARM-based IoT malware samples from [25] (such as Mirai, Stuxnet, Dark Nexus, Gafgyt, Hajime and other) have been used. In addition from used malware 51 polymorphic malware samples were generated by applying the open-source obfuscation tool [26].

According to the proposed approach, the samples of IoT malicious and benign binary executable were processed and assembly representations from these samples were extracted. With the purpose of disassembling software samples the Interactive Disassembler (IDA Pro) [27] was used To mining of maximal sequential patterns of operation codes for the all samples of obtained assembly representation hash-based partition sequential pattern mining algorithm [28] was applied.

About 20% of obtained data were used for training, and about 80% data were used as testing data to get an assessment of the effectiveness of the proposed approach. As classifiers following machine learning methods were applied [29, 30]: Random Forest [31], K-Means [32], C-Means [33], Semi-Supervised Fuzzy C-Means [34] and Support Vector Machine [35].

With aim to evaluate the effectiveness of the proposed technique, two statistical metrics were applied: F_1 score and accuracy.

Accuracy provides a statistical estimation of proportion of correctly predictions among the total number of cases studied:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}, \quad (5)$$

where TP – correctly classified samples of IoT malware, true positive;

TN – correctly classified samples of IoT benign programs, true negative;

FN – samples of IoT malware, falsely classified as benign program, false negative;

FP – samples of benign IoT programs, falsely classified as malware, false positive.

F_1 score (also named as balanced F-score or F-measure) is another measure which was applied for estimation of experiments accuracy. These score is determined as the harmonic mean of recall and precision:

$$F_1 = 2 \times \frac{PREC \times REC}{PREC + REC}, \quad (6)$$

where $PREC$ – precision or positive predictive value, which is defined as part of relevant samples among the classified samples, $PREC = \frac{TP}{TP + FP}$;

REC – recall or sensitivity, which is defined as part of relevant samples that were classified, $REC = \frac{TP}{TP + FN}$.

The results of experiments, which demonstrated values of accuracy and F1 score for proposed approach for IoT malware detection based on operation codes sequences pattern mining presented in Table 1.

As shown in the Table 1, the highest efficiency was reached by applying Semi-Supervised Fuzzy C-Means as classifier.

TABLE I. EXPERIMENTAL RESULTS: ACCURACY AND F1 SCORE VALUES OF APPROACH FOR IOT MALWARE DETECTION BASED ON OPCODES SEQUENCES PATTERN MINING

Classifier	*TP*	*TN*	*FN*	*FP*	*ACC*	*F1*
Random Forest	280	300	21	14	94,31	94,12
K-Means	283	299	18	15	94,63	94,49
C-Means	292	307	9	7	97,40	97,33
Support Vector Machine	292	309	9	5	97,72	97,66
Semi-Supervised Fuzzy C-Means	299	313	2	1	99,51	99,50

V. CONCLUSIONS

Thus, the new approach for IOT malware detection based on operation codes sequences pattern mining was proposed. The proposed approach uses mining of maximal sequential patterns of operation codes from assembly representation of IoT binary executable. Most distinctive of maximal sequential patterns are selected based on inverse document frequency values and for each of these patterns estimations of their relevance are performed. To assess the relevance, the weighted term frequency was used. The feature vectors are built based on obtained relevance for selected maximal sequential patterns. These vectors allow describing assembly

representation of IoT binary executable take into account most distinctive of maximal sequential patterns. As a classifier Semi-Supervised Fuzzy C-Means classifier was used.

Thus, when implementing the proposed approach, it is important to take into account that different IoT platforms have their own specifics. Therefore, for the experiments conduction, attention was paid to the ARM platform as one of the most widespread IoT platforms.

The results of experiments shows that the proposed approach allows detecting IOT malware with height effectiveness, with accuracy 99,51% and F_1-score at level 99,50%.

REFERENCES

[1] Trend Micro. Inside the Smart Home: IoT Device Threats and Attack Scenarios. URL: https://www.trendmicro.com/vinfo/it/security/news/internet-of-things/inside-the-smart-home-iot-device-threats-and-attack-scenarios.

[2] McAfee Labs Threats Report. URL: https://www.mcafee.com/enterprise/en-us/assets/reports/rp-quarterly-threats-nov-2020.pdf

[3] Global System for Mobile Communications. URL: https://www.gsma.com/

[4] E. M. Dovom, A. Azmoodeh, A. Dehghantanha, D. E. Newton, R. M. Parizi, H. Karimipour, "Fuzzy pattern tree for edge malware detection and categorization in IoT". *Journal of Systems Architecture*, 2019, pp. 97, 1-7.

[5] H. Zhang, X. Xiao, F. Mercaldo, S. Ni, F. Martinelli, A. K. Sangaiah, "Classification of ransomware families with machine learning based on N-gram of opcodes". *Future Generation Computer Systems*, 2019, Vol. 90, pp. 211-221.

[6] A. Azmoodeh, A. Dehghantanha, K. K. R. Choo, "Robust malware detection for internet of (battlefield) things devices using deep eigenspace learning". *IEEE transactions on sustainable computing*, 2018, Vol. 4 (1), pp. 88-95.

[7] H. Darabian, A. Dehghantanha, S. Hashemi, S. Homayoun, K. K. R. Choo, "An opcode-based technique for polymorphic Internet of Things malware detection. Concurrency and Computation": *Practice and Experience*, 2020, Vol. 32 (6), e5173.

[8] H. Darabian, A. Dehghantanha, S. Hashemi, M. Taheri, A. Azmoodeh, S. Homayoun, R. M. Parizi, "A multiview learning method for malware threat hunting: windows, IoT and android as case studies". *World Wide Web*, 2020, Vol. 23 (2), pp. 1241-1260.

[9] F. Manavi, A. Hamzeh, "A new approach for malware detection based on evolutionary algorithm". *In Proceedings of the Genetic and Evolutionary Computation Conference Companion*, 2019, pp. 1619-1624.

[10] S. Jeon, J. Moon, "Malware-detection method with a convolutional recurrent neural network using opcode sequences". *Information Sciences*, 2020, pp. 535, 1-15.

[11] D. Vasan, M. Alazab, S. Venkatraman, J. Akram, Z. Qin, "MTHAEL: Cross-Architecture IoT Malware Detection Based on Neural Network Advanced Ensemble Learning". *IEEE Transactions on Computers*, 2020, 69(11), pp. 1654-1667.

[12] G. Radhakrishnan, K. Srinivasan, S. Maheswaran, K. Mohanasundaram, D. Palanikkumar, A. Vidyarthi, "A deep-RNN and meta-heuristic feature selection approach for IoT malware detection". *Materials Today: Proceedings*, 2021.

[13] R. Lu, "Malware detection with lstm using opcode language". arXiv preprint arXiv:1906.04593, 2019.

[14] S. Bezobrazov, A. Sachenko, M. Komar, & V. Rubanau, "The methods of artificial intelligence for malicious applications detection in Android OS". *International Journal of Computing*, 15(3), 2016, pp. 184-190. https://doi.org/10.47839/ijc.15.3.851

[15] I. Obeidat, & M. AlZubi, "Developing a faster pattern matching algorithms for intrusion detection system". *International Journal of Computing*, 18(3), 2019, pp. 278-284. https://doi.org/10.47839/ijc.18.3.1520

[16] G. Markowsky, O. Savenko, A. Sachenko, "Distributed Malware Detection System Based on Decentralized Architecture in Local Area Networks", *Advances in Intelligent Systems and Computing*, 2019, 871, pp. 582-598.

[17] M. Drozd, A. Drozd, "Safety-Related Instrumentation and Control Systems and a Problem of the Hidden Faults" *The 10th International Conference on Digital Technologies 2014*, Zhilina, Slovak Republic, 2014, pp. 137–140. DOI: 10.1109/DT.2014.6868692

[18] A. Cabri, G. Suchacka, S. Rovetta and F. Masulli, "Online Web Bot Detection Using a Sequential Classification Approach," 2018.

[19] S. Ustebay, Z. Turgut and M.A. Aydin, "Intrusion Detection System with Recursive Feature Elimination by Using Random Forest and Deep Learning Classifier", *2018 International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT)*, Ankara, Turkey, 2018, pp. 71-76.

[20] T. Sochor, M. Zuzcak, "Attractiveness Study of Honeypots and Honeynets in Internet Threat Detection". *Communications in Computer and Information Science*, Springer, Cham, 2015, pp. 69-81.

[21] S.Lysenko, K.Bobrovnikova, R.Shchuka, O.Savenko,"A Cyberattacks Detection Technique Based on Evolutionary Algorithms". *In 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies*, 2020, pp. 127-132.

[22] S. Lysenko, K. Bobrovnikova, O. Savenko, R. Shchuka, "Technique for Cyberattacks Detection Based on DNS Traffic Analysis", *CEUR-WS*, Vol 2732. ISSN: 1613–0073, 2020. pp. 171-182.

[23] I. Santos, F. Brezo, X. Ugarte-Pedrero, P. G. Bringas, "Opcode sequences as representation of executables for data-mining-based unknown malware detection". *Information Sciences*, 2013, Vol. 231, pp. 64-82.

[24] Packages Search for Linux and Unix. URL: https://pkgs.org/

[25] VirusTotal. URL: http://www.virustotal.com

[26] Obfuscatoin-for-ARM-disassembled-binary. URL: https://github.com/darabian/Obfuscatoin-for-ARM-disassembled-binary

[27] Hex Rays. IDA Pro. URL: https://www.hex-rays.com/products/ida/

[28] R. Millham, I. E. Agbehadji, H. Yang, "Pattern Mining Algorithms". *Bio-inspired Algorithms for Data Streaming and Visualization, Big Data Management, and Fog Computing*. Springer, Singapore, 2021, pp. 67-80.

[29] S. Nakhodchi, A. Upadhyay, A. Dehghantanha, "A comparison between different machine learning models for IoT malware detection". *Security of Cyber-Physical Systems*. Springer, Cham, 2020, pp. 195-202.

[30] W. Peters, A. Dehghantanha, R. M. Parizi, & G. Srivastava, "A comparison of state-of-the-art machine learning models for OpCode-based IoT malware detection". *In Handbook of Big Data Privacy*, 2020, pp. 109-120. Springer, Cham.

[31] Khammas, B. M., "Ransomware Detection Using Random Forest Technique". *ICT Express*, *6*(4), 2020, pp. 325-331.

[32] Q. Wang, L. Li, B. Jiang, Z. Lu, J. Liu, & S. Jian, "Malicious domain detection based on k-means and smote". *In International Conference on Computational Science*, 2020, pp. 468-481. Springer, Cham.

[33] I. Hafeez, M. Antikainen, A. Y. Ding, & S. Tarkoma, "IoT-KEEPER: Detecting malicious IoT network activity using online traffic analysis at the edge". *IEEE Transactions on Network and Service Management*, 2020, 17(1), pp. 45-59.

[34] S. Lysenko, O. Savenko, K. Bobrovnikova. "DDoS Botnet Detection Technique Based on the Use of the Semi-Supervised Fuzzy c-Means Clustering". *CEUR-WS*, ISSN: 1613–0073, 2018. Vol. 2104. pp. 688-695.

[35] S. Lysenko, K. Bobrovnikova, O. Savenko, A. Kryshchuk "BotGRABBER: SVM-Based Self-Adaptive System for the Network Resilience Against the Botnets' Cyberattacks". In: Gaj P., Sawicki M., Kwiecień A. (eds) *Computer Networks. CN 2019.*

Додаток В

(обов’язковий)

Копія тез доповіді на Всеукраїнській науково-практичній конференції Актуальні Проблеми Комп’ютерних Наук (АПКН-2021)

УДК 004.896

Гаврилюк Р. Л., Бобровнікова К. Ю.

Хмельницький національний університет

ДОСЛІДЖЕННЯ МЕТОДІВ ПОБУДОВИ БЛОКЧЕЙНУ В РОЗПОДІЛЕНИХ СИСТЕМАХ

Незважаючи на те, що блокчейн дуже сильно асоціюється з криптовалютами у головах сучасних людей, це далеко не єдиний спосіб його використання. Перш за все, блокчейн - це технологія, яка дозволяє безпечно і ефективно проводити транзакції між вузлами блокчейн-мережі, з можливістю їх бектрекінгу. В роботі проведено огляд сучасних методів побудови блокчейну в розподілених системах та окреслено їх загальні недоліки.

Despite those who strongly associate the blockchain with crypto-currencies, it is far from being the only way of blockchain-technology usage. First of all, blockchain is a system that permits secured and efficient transactions between the nodes of the blockchain-network, with the possibility of their backtracking. This work reviews modern methods of blockchain-realizations within distributed systems and represents their general disadvantages.

На сьогодні існує багато сфер використання блокчейну, не пов'язаних з криптовалютами: медицина та охорона здоров'я, Інтернет речей (IoT), цифрова реклама, страхування тощо [1]. В цієї технології є великий потенціал, і за грамотної реалізації вона може принести значну користь як в сферах бізнесу, так і в державних сферах.

Оскільки блокчейн – це технологія, яка стала популярною відносно нещодавно, багато організацій намагаються залучити її в свій бізнес, в основному через бажання включити інноваційну технологію у проект [2]. Важливо розуміти, що хоч технологія блокчейн і є цікавою, але вона не підходить для того, щоб виконувати всі задачі, до того ж, в неї є ряд недоліків, які роблять деякі варіанти реалізації недоцільними. Згідно рекомендацій (DHS) Science & Technology Directorate, що є складовою частиною Міністерства національної безпеки США, для прийняття рішення про доцільність використання блокчейну в конкретній задачі може бути використана схема [2], наведена на рисунку 1.

За допомогою простого набору питань та відповідей "Так/Ні", наведена схема дозволяє визначити, чи релевантно використовувати саме блокчейн у задачі, і чи, можливо, є більш ефективне рішення під цю конкретну задачу. До прикладу, в прикладних задачах найчастіше можна обійтись звичайною базою даних. Але, якщо задачу доцільно реалізувати з використанням блокчейну згідно наведеного алгоритму, то ця реалізація може бути використана достатньо ефективно.

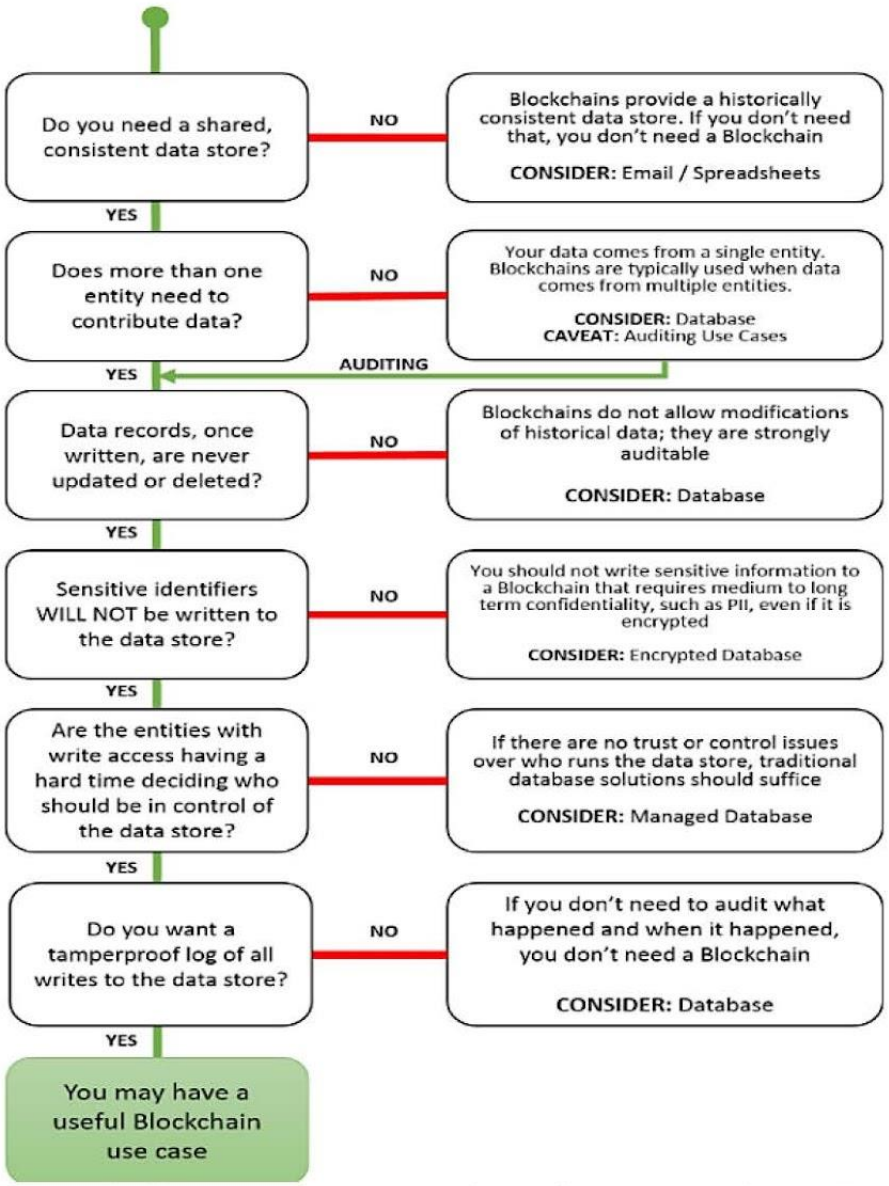


Рисунок 1 – Алгоритм визначення доцільності використання блокчейн технології [2]

Наведемо визначення основних важливих термінів в галузі технології блокчейн [2]. Під *блокчейн мережею* мається на увазі мережа, в якій використовується даний блокчейн. *Блокчейн реалізація* - це конкретний блокчейн.

Вузол – це індивідуальна система в блокчейн мережі. Термін *повний вузол* визначає вузол, який містить в собі повний блокчейн та може перевіряти валідність транзакцій. При цьому *легкий вузол* – це вузол, який не містить в собі повний блокчейн і тому повинен проводити всі транзакції через повні вузли.

Блоки в блокчейні побудовані таким чином, що заголовок кожного наступного блоку містить в собі хеш попереднього. тому можна відслідкувати всю історію до першого блоку. Загальна структура блоку складається з: заголовку, лічильника транзакцій і самих транзакцій (рисунок 2).

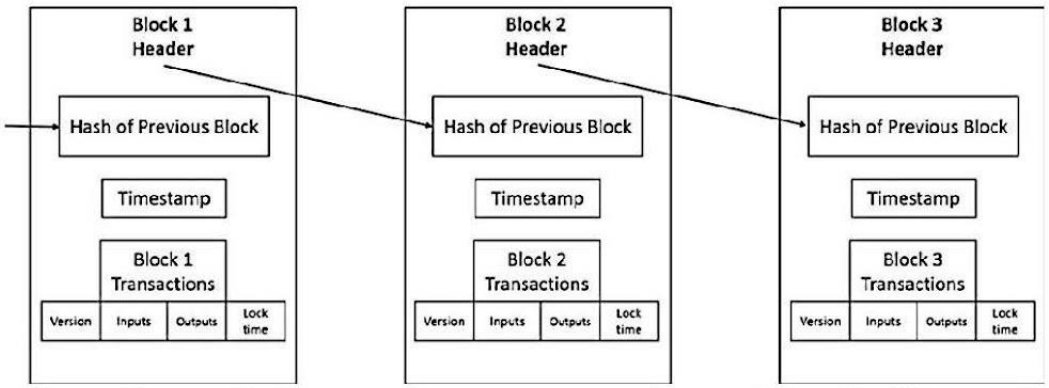


Рисунок 2 – Схематичне представлення структури блокчейну [3]

Як видно з рисунку 2, транзакції займають в блоці найбільше місця, при цьому кожен блок зазвичай містить більше однієї транзакції, а новий блок створюється лише після переповнення попереднього, оскільки кожен блок має обмежений розмір. Транзакція містить дані про відправника, одержувача, час відправлення і номеру блоку.

На рисунку 3 наведено загальну схему виконання транзакцій [3]. Як видно з рисунку 3, для того, щоб провести транзакцію і записати її в блокчейн, потрібно підтвердити її на повних вузлах, і тільки після підтвердження відбудеться виконання транзакції. Тому однією з основних вразливостей технології блокчейн є так звана вразливість 51%, суть якої полягає в тому, що якщо хтось буде контролювати більшу половину повних вузлів мережі, то він зможе одноосібно приймати рішення стосовно блокчейн мережі, які потребують узгодження з усіма вузлами [4].

Відомо багато підходів, спрямованих на усунення вразливостей та недоліків блокчейн технології [5]. В [6] пропонується оптимізована структура дерева Меркла для ефективної перевірки транзакцій у надійних системах IoT з підтримкою блокчейна. Для пришвидшення підтвердження транзакцій в роботі [7] запропоновано метод, заснований на машинному навчанні, який використовує

автоматизоване підтвердження транзакцій блокчейна, включаючи також персоналізовану ідентифікацію аномальних транзакцій.

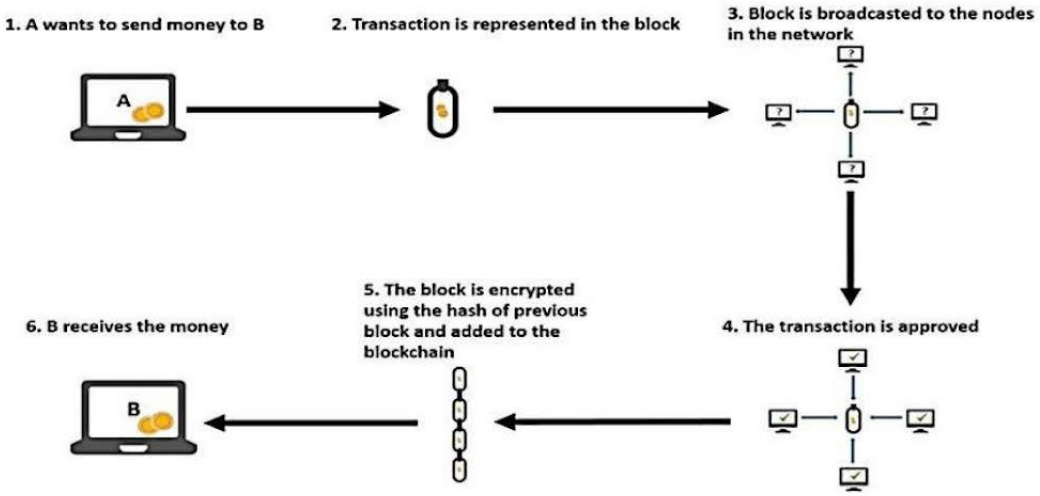


Рисунок 3 – Схема виконання транзакцій [3]

У біткоїн блокчейні реальна ідентичність організації прихована за псевдонімом, так званою адресою. Тому вважається, що біткоїн забезпечує високий ступінь анонімності, що є рушієм його частого використання для незаконної діяльності. На усунення цієї вразливості в [8] представлений підхід до зменшення анонімності блокчейна за допомогою машинного навчання з учителем та алгоритму Gradient Boosting для прогнозування типу ще невпізнаних сутностей.

В роботі [9] запропоновано модель DML (distributed machine learning), що зберігає конфіденційність для дозволеного блокчейна та надає можливість вирішувати проблеми конфіденційності, безпеки та продуктивності. З цією метою було розроблено диференціально приватний метод стохастичного градієнтного спуску та правило агрегації на основі помилок в якості основних примітивів. Розроблена модель може розглядати будь-який тип диференціально приватного алгоритму навчання, де потрібно визначити недетерміновані функції. Запропоноване правило агрегації на основі помилок ефективне для запобігання атакам з боку вузла зловмисника, який намагається погіршити точність роботи моделей DML.

Незважаючи на велику кількість відомих методів побудови блокчейну в розподілених системах, огляд джерел показав, що ця задача не втрачає своєї актуальності. Відомі підходи, спрямовані на вирішення цієї задачі, мають ряд недоліків та обмежень, основними з яких є низька продуктивність, проблеми з

безпекою та конфіденційністю. Таким чином, можна зробити висновок, що існує необхідність у розробленні нових методів побудови блокчейну в розподілених системах, які б усували недоліки відомих підходів.

Перелік посилань

1. AMD и технология блокчейна URL: https://www.amd.com/ru/technologies/blockchain
2. Yaga D. et al. Blockchain technology overview //arXiv preprint arXiv:1906.11078. – 2019.
3. Angraal S., Krumholz H. M., Schulz W. L. Blockchain technology: applications in health care //Circulation: Cardiovascular quality and outcomes. – 2017. – Vol. 10. – No. 9. – pp.. e003800.
4. Ткаченко А. Л., Степанова А. С., Гераева Е. В. Аспекты безопасности системы блокчейн //Advanced science. – 2018. – С. 29-31.
5. Koteska B., Karafiloski E., Mishev A. Blockchain implementation quality challenges: a literature //SQAMIA 2017: 6th Workshop of Software Quality, Analysis, Monitoring, Improvement, and Applications. – 2017. – pp. 11-13.
6. Wang J. et al. An optimized transaction verification method for trustworthy blockchain-enabled IIoT //Ad Hoc Networks. – 2021. – Vol. 119. – pp. 102526.
7. Podgorelec B., Turkanović M., Karakatič S. A machine learning-based method for automated blockchain transaction signing including personalized anomaly detection //Sensors. – 2020. – Vol. 20. – No. 1. – pp. 147.
8. Harlev M. A. et al. Breaking bad: De-anonymising entity types on the bitcoin blockchain using supervised machine learning //Proceedings of the 51st Hawaii International Conference on System Sciences. – 2018.
9. Kim H. et al. Efficient privacy-preserving machine learning for blockchain network //IEEE Access. – 2019. – Vol. 7. – pp. 136481-136495.

Додаток Г

(обов’язковий)

Презентація до дипломної роботи

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
Кафедра комп’ютерної інженерії та системного програмування

Інтелектуалізована мобільна розподілена система для побудови блокчейну

Науковий керівник: к.т.н.
Бобровнікова К. Ю.
Доповідач: Гаврилюк Р. Л.

Об’єктом дослідження є процес побудови блокчейну в мобільних розподілених системах

Предметом дослідження є моделі, методи та апаратно-програмні засоби для підвищення ефективності побудови блокчейну.

Метою дипломної роботи є розроблення інтелектуалізованої мобільної розподіленої системи для побудови блокчейну для підвищення ефективності побудови блокчейну в мобільних розподілених системах.

Задачі дослідження

1. Провести огляд відомих рішень для побудови блокчейну в мобільних розподілених системах.

2. Провести огляд відомих рішень для підвищення ефективності побудови блокчейну.

3. Удосконалити модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат.

4. Розробити інтелектуалізований метод для підвищення ефективності побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат.

5. На основі розробленого методу створити апаратно-програмне рішення мобільної розподіленої системи, застосування якого надасть можливість підвищити ефективність побудови блокчейну, в порівнянні з відомими аналогами

Наукова новизна

1. Удосконалено модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, що заснована на удосконалених моделях: завантаження даних в мобільні вузли, які беруть участь в побудові блокчейну в розподілених системах; вирішення PoW задачі мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах; поширення блоку мобільними вузлами, які беруть участь в побудові блокчейну в розподілених системах. Удосконалена модель, на відміну від відомих моделей, надає можливість обчислювати ефективність побудови блокчейну для окремих популяцій мобільних вузлів.

2. Розроблено інтелектуалізований метод побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, який заснований на удосконаленій моделі. Метод грунтується на еволюційних алгоритмах та, на відміну від відомих підходів, надає можливість на основі наявної множини мобільних вузлів створювати максимально ефективну популяцію мобільних вузлів, які беруть участь у процесі побудови блокчейну, базуючись на часі створення блоків і енерговитратах. Розроблений метод надає можливість підвищити ефективність побудови блокчейну, в порівнянні з відомими методами для побудови блокчейну.

3. Побудовано інтелектуалізовану мобільну розподілену систему для побудови блокчейну, яка грунтується на розробленому методі, та надає можливість підвищити ефективність побудови блокчейну, в порівнянні з відомими методами для побудови блокчейну.

Практична цінність

Практична цінність дипломної роботи полягає у розроблені інтелектуалізованої мобільної розподіленої системи для побудови блокчейну, яка надає можливість підвищити ефективність побудови блокчейну на 12-20%, в порівнянні з відомими підходами.

Модель

Модель

Модель

Модель

$$R_i = \frac{rP_i \cdot c_1 \left(\frac{T_i^{-1}}{\sum_{j \in N'} T_j} \right)}{c_2 \left(E_i^t + E_i^m + E_i^p \right)} - R_{min}, \; i \in N'$$

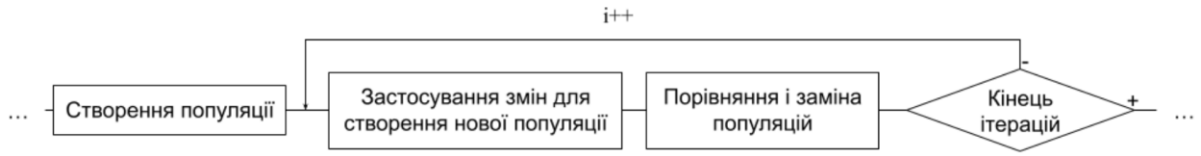
Модель

$$R = \sum_{i \in N'} R_i \begin{cases} f^{min} \le f_i \le f^{max}, \; i \in N' \\ p^{min} \le p_i \le p^{max}, \; i \in N' \\ T_i^t + T_i^m + T_i^p \le T^{max}, \; i \in N' \\ E_i^t + E_i^m + E_i^p \le E^{max}, \; i \in N' \end{cases}$$

Метод

Метод полягає в підборі такої популяції мобільних вузлів, яка була б максимально ефективною для побудови блокчейну, згідно створеної моделі.

Шляхом застосування змін до теперішньої популяції, створюється нова популяція, і шляхом порівняння ефективностей двох популяцій, обирається сильніша для наступної ітерації



Метод

1. В якості вхідних даних приймається мінімальна кількість учасників (n_min), а також максимальна кількість обчислень (MaxE), і загальна кількість учасників (n).

2. Ініціалізувати змінні (p, f) для числових значень потужності передачі і обчислювальних ресурсів відповідно, і записати у масив N розміром n.

3. З масиву N випадковим чином обрати n_min учасників для масиву N'.

4. Виконати базові обчислення за формулами 3.1, на основі сформованого масиву N'.

5. Ініціалізувати вектор для вірогідностей op = {op1, op2, op3}, який повинен бути нормалізований.

6. Ініціалізувати список BList для запису неефективних учасників.

7. Поки кількість операцій не досягне MaxE, виконувати кроки (8-10).

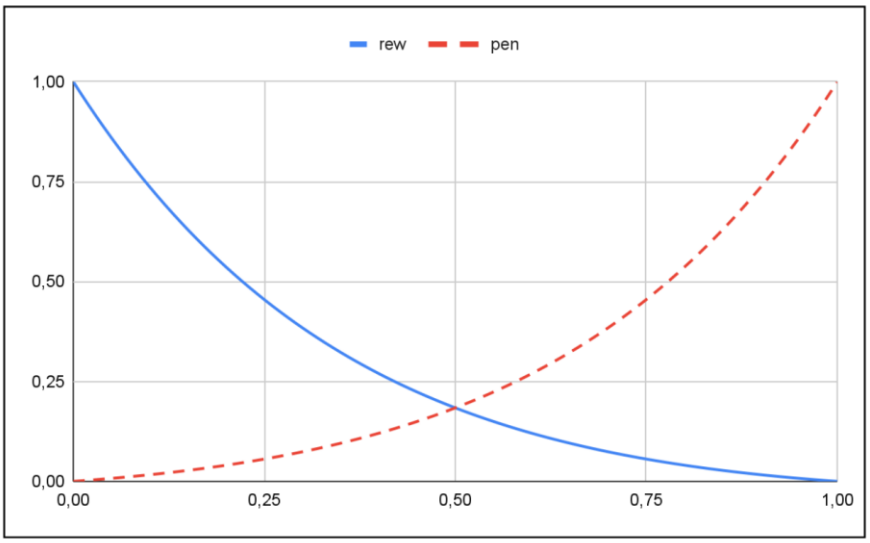
8. Використовуючи операції мутації створити змінений масив N''.

9. Використовуючи N'', функцію для оновлення популяції мобільних вузлів, які беруть участь в побудові блокчейну в розподілених системах.

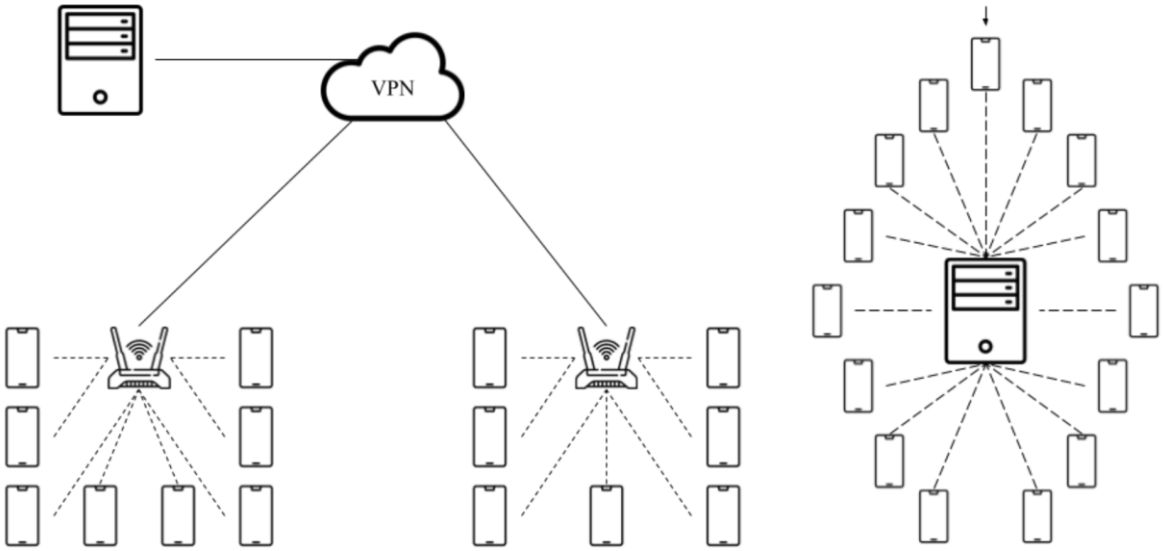
10. Використовуючи N'', функцію для адаптації шансів мутацій популяції мобільних вузлів, які беруть участь в побудові блокчейну в розподілених системах.

11. На виході функції отримуємо N.

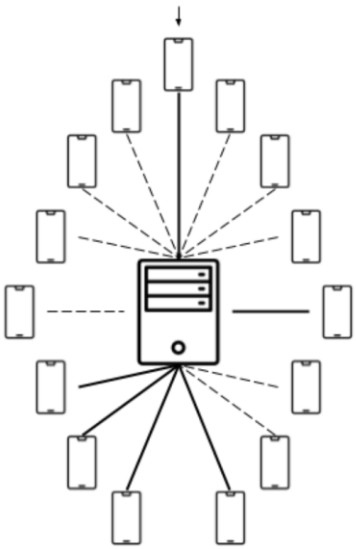
Метод



Апаратна частина



Результат роботи

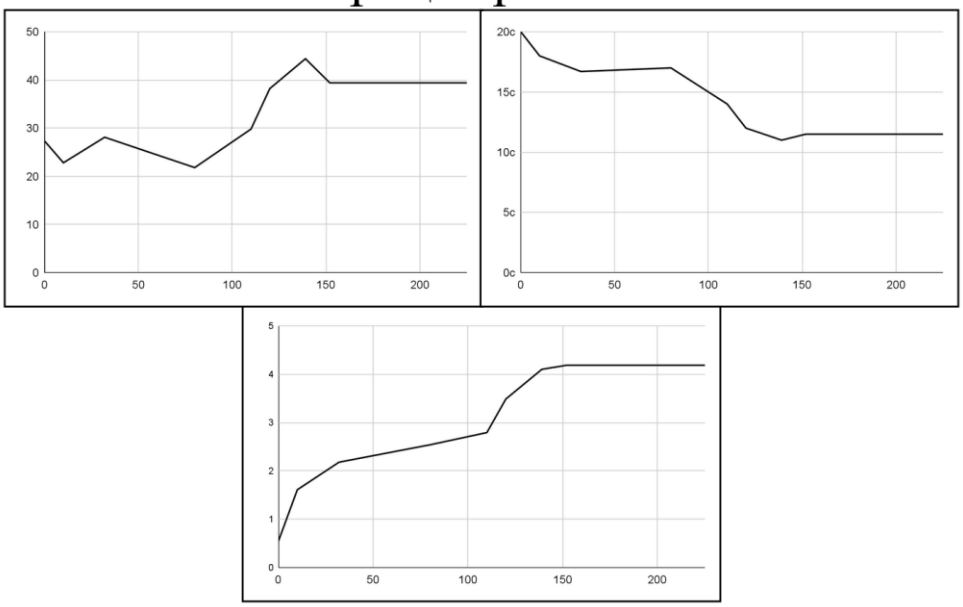


Початкову популяцію сформували 1, 4, 9, 11 мобільні вузли.

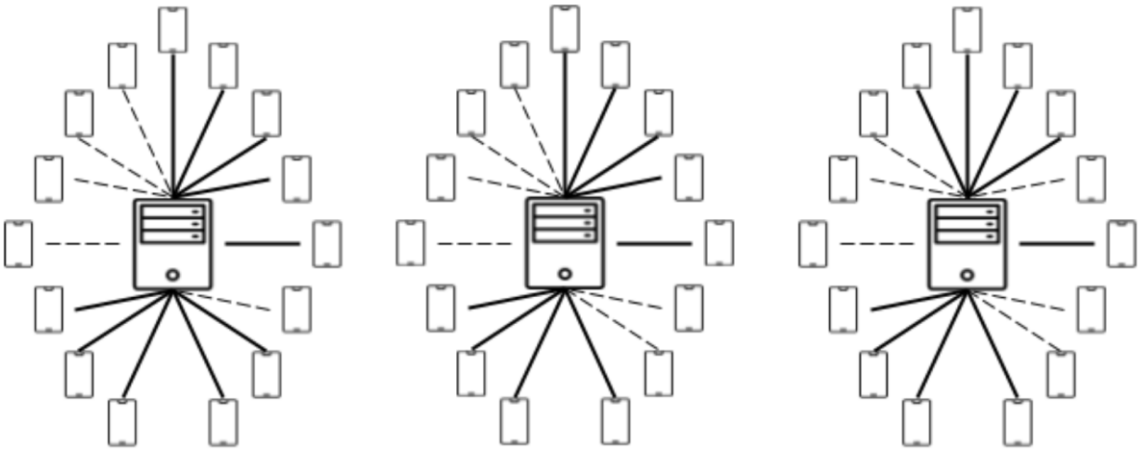
В результаті роботи інтелектуалізованої мобільної розподіленої системи для побудови блокчейну була сформована популяція яка складається з 1, 5, 8, 9, 10, 11 мобільних вузлів.

Сформована популяція вважається найбільш ефективною, серед інших можливих популяцій, для виконання поставленої задачі побудови блокчейну.

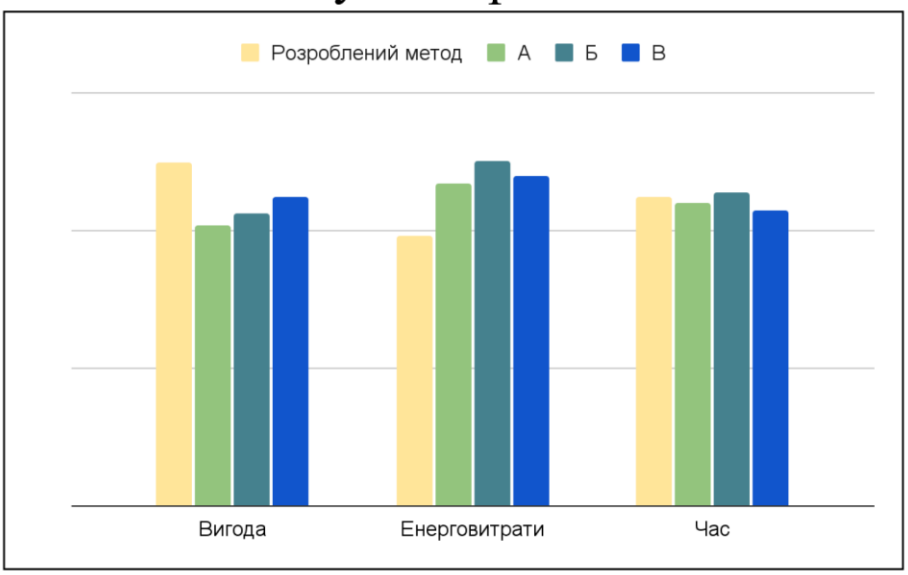
Процес роботи



Результати роботи аналогів



Результат роботи



Наявні публікації

За темою дипломної роботи опубліковано статтю на тему «The Approach for IoT Malware Detection Based on Opcodes Sequences Pattern Mining» в матеріалах конференції 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021, що індексуються в наукометричній базі Scopus, а також опубліковано тези у Всеукраїнській науково-практичній конференції Актуальні Проблеми Комп'ютерних Наук (АПКН-2021). Було взято участь у Всеукраїнській науково-практичній конференції Актуальні Проблеми Комп'ютерних Наук.

Висновки

В даній роботі за результатами виконаних теоретичних та практичних досліджень було розроблено інтелектуалізований метод та апаратно-програмне рішення для збільшення ефективності побудови блокчейну в мобільних розподілених системах.

У першому розділі була досліджена предметна область, досліджені відомі методи побудови блокчейни, а також відомі методи виконання мобільних розподілених обчислень, і методів збільшення ефективності побудови блокчейну.

У другому розділі удосконалено модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, яка базується на використанні трьох інших моделей, і яка, на відміну від відомих моделей, заснована на врахуванні часу побудови блоків та енерговитратах.

Запропонована модель є основою інтелектуалізованого методу побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат.

Висновки

Третій розділ представляє інтелектуалізований метод побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, який, на відміну від відомих методів, ґрунтується на удосконаленій моделі побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, що дозволяє підібрати максимально ефективну популяцію для побудови блокчейну згідно заданих параметрів.

У четвертому розділі, беручи за основу розроблений метод, було побудовано апаратно-програмну реалізацію інтелектуалізованої мобільної розподілної системи для побудови блокчейну, яка дозволяє підібрати найбільш ефективну популяцію мобільних вуздів, на основі даних отриманих емпіричним шляхом.

Проведено експериментальні дослідження, результати яких показали, що застосування розробленого рішення надає можливість підвищити ефективність побудови блокчейну на 12-20% в порівнянні з відомими методами.

Отже, створена інтелектуалізована мобільна розподілена система для побудови блокчейну надає можливість підібрати максимально ефективну популяцію мобільних вузлів, в залежності від їх загальної множини.

Дякую за увагу

UNICHECK

Ім'я користувача:
Кафедра КІ

ID перевірки:
1011239911

Дата перевірки:
18.05.2022 20:33:36 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
18.05.2022 20:40:01 EEST

ID користувача:
100005591

Назва документа: АП Гаврилюк_Інтелектуалізована мобільна розподілена система для побудови блокчейну

Кількість сторінок: 85 Кількість слів: 15364 Кількість символів: 118564 Розмір файлу: 703.86 KB ID файлу: 1011131071

1.09%
Схожість

Найбільша схожість: 0.62% з джерелом з Бібліотеки (ID файлу: 1008222700)

0.41% Джерела з Інтернету	11	Сторінка 87
1.08% Джерела з Бібліотеки	95	Сторінка 87

0% Цитат

Не знайдено жодних цитат

Не знайдено жодних посилань

0%
Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи	3

Wed May 18 19:42:13 EEST 2022, Медзатий Дмитро Миколайович, Хмельницький національний університет, ХНУ

Anti-Plagiarism v-15.257

Максимальное совпадение с одним документом 0.0%

Словари проверки: en_US, ru_RU, ua_UA. **Ошибок в документах: 7%**

ID: 103622 Название: Інтелектуалізована мобільна розподілена система для побудови блокчейну Добавлено в БД: 2022-05-18 Авторы: Гаврилюк Р.Л. Руководители: Бобровнікова К.Ю Консультанты: Оппоненты:	Документ		Суммарное совпадение по Базе Данных	
	Символы	Лексемы	Символы	Лексемы
	104535	698	570 (1%)	8 (1%)

Источник плагиата

ID	Описание	Наличие плагиата в документе	
		Символы	Лексемы

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА ДИПЛОМНУ РОБОТУ

Дипломник: Гаврилюк Роман Леонідович

Тема: Інтелектуалізована мобільна розподілена система для побудови блокчейну

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг дипломної роботи:

Кількість листів креслень ___; кількість сторінок записки 80

1. Короткий зміст роботи та прийнятих рішень В дипломній роботі удосконалено модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат. На основі удосконаленої моделі розроблено інтелектуалізований метод побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат, а також побудована інтелектуалізована мобільна розподілена система для побудови блокчейну, яка надає можливість підвищити ефективність побудови блокчейну для мобільних розподілених мереж, враховуючи такі параметри як час побудови блоків та енергоефективність.

2. Висновок про відповідність роботи дипломному завданню Дипломна робота відповідає виданому завданню як в теоретичній, так і в практичній частині.

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: Розділ 1 – проведено огляд відомих методів для підвищення ефективності побудови блокчейну та визначено їх недоліки. Розділ 2 – удосконалено модель побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат. Розділ 3 – розроблено інтелектуалізований метод побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат та проведено експериментальні дослідження розробленого методу. Розділ 4 – побудована інтелектуалізована мобільна розподілена система для побудови

блокчейну та представлено результати роботи розробленої системи, в порівнянні з відомими аналогами. Всі розділи відповідають завданню.

4. Позитивні сторони роботи: Застосування інтелектуалізованого методу побудови блокчейну в мобільних розподілених системах з урахуванням часу побудови блоків та енерговитрат надає можливість підібрати найбільш ефективну популяцію мобільних вузлів, і таким чином підвищити ефективність побудови блокчейну, в порівнянні з відомими методами.

8. Негативні сторони роботи: _____

_____

_____

_____

_____

6. Оцінка графічного оформлення та пояснювальної записки роботи Пояснювальна записка відповідає нормам оформлення та виконана на достатньо високому рівні.

7. Відгук про роботу в цілому: Дипломна робота заслуговує відмінної оцінки. Дипломна робота присвячена вирішенню актуальної задачі підвищення ефективності побудови блокчейну. Усі розділи роботи йдуть у вірній послідовності, що дозволяє розуміти викладений матеріал.

8. Інші зауваження: _____

_____

_____

_____

_____

9. Оцінка дипломної роботи: Розглянувши позитивні сторони предетавленої дипломної роботи, можна зробити висновок, що вона заслуговує на оцінку «відмінно». _____

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи) Мартинюк Валерій Володимирович, д.т.н., професор, завідудвач кафедрою Автоматизації, комп'ютерно-інтегрованих технологій і телекомунікацій _____

"19" травня 2022 р. _____ (підпис)

Завідувачу кафедри КІІС
д-р.техн.наук, проф. Говорущенко Т. О.

Гаврилюка Романа Леонідовича
ПІБ здобувача вищої освіти

ФПКТС, 2 курсу, групи КІ2м-19-1

ЗАЯВА

З правилами чинного Положення «Про дотримання академічної доброчесності в Хмельницькому національному університеті» від 26.09.2020 (зі змінами від 26.11.2020), згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений (а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

17.05.22
дата

підпис

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ

КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОМАЦІЙНИХ СИСТЕМ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Інтелектуалізована мобільна розподілена система для побудови блокчейну

Автор: Гаврилюк Роман Леонідович

Спеціальність: 123 – Компютерна інженерія

Освітня програма: освітньо-наукова

Науковий керівник: Бобровнікова Кіра Юліївна, к.т.н, доцент

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

1) запозичення розміщені в розділах аналізу існуючих аналогів та прототипів, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;

2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;

3) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості, складає 1.09% і адресується до 81 першоджерела, що, з урахуванням наведених обгрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи _____ К. Ю. Бобровнікова

Гарант ОП _____ О. С. Савенко

Завідувач кафедри КІІСч _____ Т. О. Говорущенко