НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені Ігоря СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра математичних методів захисту інформації

«До захисту допущено»
В.о. завідувача кафедри
_____ Михайло САВЧУК
«__» _____ 2021 р.

Дипломна робота

на здобуття ступеня бакалавра

зі спеціальності: 113 Прикладна математика
на тему: **«Новий метод віддаленого оцінювання гешрейта майнера»**

Виконав: студент 4 курсу, групи ФІ-74
Волинський Євгеній Олександрович

Керівник: д.т.н., проф. кафедри ММЗІ Ковальчук Л.В._____

Консультант: _ _____

Рецензент: к.т.н, доц. кафедри ІБ Стьопочкіна І.В. _____

Засвідчую, що у цій дипломній роботі немає запозичень з праць інших авторів без відповідних посилань.

Студент _____

Київ — 2021

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені Ігоря СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра математичних методів захисту інформації

Рівень вищої освіти — перший (бакалаврський)
Спеціальність (освітня програма) — 113 Прикладна математика,
ОПП «Математичні методи криптографічного захисту інформації»

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри
_____ Михайло САВЧУК
«__» _____ 2021 р.

ЗАВДАННЯ
на дипломну роботу

Студент: Волинський Євгеній Олександрович

1. Тема роботи: «*Новий метод віддаленого оцінювання гешрейта майнера*»,

керівник: д.т.н., проф. кафедри ММЗІ Ковальчук Л.В.,

затверджені наказом по університету №__ від «__» _____ 2021р.

2. Термін подання студентом роботи: 4 червня 2021р.

3. Вихідні дані до роботи: Протокол консенсусу PoW, алгоритм гешування мережі Bitcoin.

4. Зміст роботи: Під час дослідження буде проведено огляд опублікованих джерел за тематикою дослідження, виконанано аналіз наведених у літературі наявних методів, розроблено та реалізовано новий ефективний метод оцінюванная гешрейта майнера та підтверджено його коректність.

5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо): Презентація доповіді.

6. Дата видачі завдання: 10 вересня 2020 р.

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання	Примітка
1	Узгодження теми роботи із науковим керівником	01-15 вересня 2020 р.	Виконано
2	Огляд опублікованих джерел за тематикою дослідження	Вересень-жовтень 2020 р.	Виконано
3	Аналіз протоколів консенсусу, способів майнінгу, схем винагороди та методів оцінки гешрейта	Жовтень-грудень 2020 р.	Виконано
4	Робота над новим методом оцінки гешрейта	Січень-березень 2021 р.	Виконано
5	Формальне описання та обґрунтування запропонованого методу	Березень-квітень 2021 р.	Виконано
6	Формальне визначення похибок оцінювання за запропонованим методом	Квітень-травень 2021 р.	Виконано
7	Робота над програмною реалізацією запропонованого методу	Травень 2021 р.	Виконано
8	Формулювання результатів дослідження	Травень-червень 2021 р.	Виконано

Студент _____ Волинський Є.О.

Керівник _____ Ковальчук Л.В.

РЕФЕРАТ

Кваліфікаційна робота містить: 54 стор., 13 рисунків, 1 таблицю, 19 джерел та 1 додаток. У роботі описано проблему децентралізації, що виникає під час об'єднання майнерів у пули, та розглянуто вже запропоновані варіанти оцінки гешрейтів майнерів. Зокрема, метод оцінки за використанням шар та метод, заснований на середньому значені порядкових статистик геш-значень. Також детально розглянуто основні схеми розподілу винагороди в пулах. Запропоновано новий метод оцінки гешрейта майнера та його формальне обґрунтування, у тому числі визначено похибки оцінювання, практично підтверджено отримані результати а також проведено кількісний та якісний аналіз.

БЛОКЧЕЙН, КРИПТОВАЛЮТА, BITCOIN, МАЙНІНГ, POW, ГЕШРЕЙТ

ABSTRACT

Qualification work contains: 54 pages, 13 figures, 1 table, 19 sources and 1 appendix. The work describes the problem of decentralization, that arises when miners are connected to the pool, and considered the already proposed options for estimating miners hashrates. In particular, the method of estimation using a shares and the method, based on the average value of ordinal statistics of hash-values. The main schemes of reward distribution in pools are also considered in detail. A new method for estimating a miner's hashrate and its formal substantiation are proposed, including estimation errors, practically confirming the obtained results and performing quantitative and qualitative analysis.

BLOCKCHAIN, CRYPTOCURRENCY, BITCOIN, MINING, POW, HASHRATE

ЗМІСТ

Перелік умовних позначень, скорочень і термінів 7
Вступ 8
1 Основні поняття, означення та теоретичні підстави 10
1.1 Поняття блокчейну та його властивості 10
1.2 Що розуміється під криптовалютою 13
1.3 Майнінг та його значення для блокчейнів 14
1.4 Протоколи консенсусу 14
1.4.1 Вимоги до протоколів досягнення консенсусу 15
1.4.2 Огляд існуючих протоколів консенсусу 16
1.5 Атаки на блокчейни 19
Висновки до розділу 1 21
2 Існуючі методи визначення гешрейта та розподілу винагороди в пулах 22
2.1 Соло-майнінг 22
2.2 Майнінговий пул 23
2.2.1 Проблема децентралізації 24
2.3 Схеми розподілу винагороди 25
2.4 Складність, гешрейт та його розрахунок пулом 28
2.5 Огляд альтернативного методу визначення гешрейта 32
Висновки до розділу 2 35
3 Новий метод дистанційного визначення гешрейта 36
3.1 Формалізація нового методу визначення гешрейта 36
3.2 Визначення можливих похибок оцінки 39
3.3 Практичне підтвердження отриманих результатів 41
Висновки до розділу 3 48
Висновки 49
Перелік посилань 50
Додаток А Тексти програм 52
А.1 Програма симуляції результатів, отриманих у розділі 3 52

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

DLT — група методів спрямованих на створення розподілених баз даних і забезпечення несуперечності, синхронізації, незмінності і прозорості зберігаємої в них інформації;

форк — розділення безперервного ланцюга блокчейна на два ланцюжки;

гешрейт — величина, яка показує обчислювальну потужність обладнання, вимірюється в гешах за секунду (г/с);

шара — можливе рішення блоку, геш-значення, яке є меншим за таргет;

таргет — геш-значення, яким має бути геш блоку, щоб блок був доданий до блокчейну;

nonce — число, яке майнери підбирають щоб видобути блок;

мемпул — набір всіх транзакцій, які очікують підтвердження в мережі;

баунті — винагорода, яку пул виплачує майнеру згідно деяких правил;

POW — алгоритм консенсусу, заснований на доведенні виконання роботи;

SHA-256 — одностороння геш-фунція.

ВСТУП

Актуальність дослідження. У зв'язку з динамічним розвитком технології блокчейн і тісно пов'язаного з нею ринку криптовалют, які з кожним роком залучають все більше і більше уваги з боку різного роду стартапів, технологічних компаній і інвесторів — підвищується і актуальність майнінгу. Майнінг, в свою чергу, є механізмом для забезпечення функціонування блокчейн-мереж. Так як найвищою ліквідністю володіють майнінгові пули, практично повністю девальвуючи соло-майнінг, в кваліфікаційній роботі мова піде, в основному, саме про пули.

Незважаючи на очевидні переваги блокчейна — децентралізацію (відсутність єдиного центру управління або місця зберігання, підтримання працездатності безпосередньо за рахунок всіх учасників мережі), високий рівень безпеки (гешування і цифровий підпис є основними криптографічними примітивами, на яких будується блокчейн), прозорість (кожен учасник мережі має доступ до всієї історії транзакцій), високу швидкість і низьку вартість — нинішній блокчейн не позбавлений недоліків. На жаль, користувачі мережі не готові надавати свої обчислювальні потужності на чистому ентузіазмі, з цього виникає первинна проблема — розподіл винагороди за виконану роботу, пропорційно внеску кожного з майнерів, в умовах повної недовіри, і супутні їй: визначення гешрейта та частки цього гешрейта в загальній потужності, для подальшого розрахунку винагороди. Основним вектором дослідження кваліфікаційної роботи є саме оцінювання гешрейта, величина якого є ключовим фактором якості та прибутковості майнінгу.

Метою дослідження є розробка нового методу віддаленого оцінюванная гешрейта майнера. Для досягнення мети необхідно розв'язати наступні **задачі дослідження**:

1) провести огляд опублікованих джерел за тематикою дослідження;

2) виконати аналіз наведених у літературі наявних методів;

3) розробити новий ефективний метод оцінюванная гешрейта;

4) реалізувати новий метод та переконатися у його коректності.

Об'єктом дослідження є процес майнінгу в пулі за протоколом PoW.

Предметом дослідження є віддалене оцінювання гешрейта майнера та визначення похибок наближення.

При розв'язанні поставлених завдань використовувались такі *методи дослідження*: методи криптографії, теорії ймовірностей, математичної статистики, комп'ютерного та статистичного моделювання.

Наукова новизна отриманих результатів полягає у тому, що буде запропоновано новий, ефективний як для операторів пула, так і для майнерів, метод визначення гешрейта, який скорочує часові та економічні витрати майнера.

Практичне значення полягає у тому, що отримані результати, враховуючи їх точність, можуть бути застосовані в роботі реальних майнінгових пулів, що дозволить з боку пула підключати нових майнерів швидше, а з боку майнерів — підвищить безпечність та економічну ефективність.

1 ОСНОВНІ ПОНЯТТЯ, ОЗНАЧЕННЯ ТА ТЕОРЕТИЧНІ ПІДСТАВИ

У даному розділі розглядаються необхідні для дослідження теоретичні відомості з криптографії та основні поняття, які використовуються при побудові блокчейн-систем.

1.1 Поняття блокчейну та його властивості

Блокчейн — це технологія децентралізованої системи проведення різних операцій з даними, в якій вся інформація представлена у вигляді ланцюга блоків даних. Блокчейн ще називають технологією розподілених реєстрів, тому що весь ланцюг зберігають на своїх комп'ютерах безліч незалежних користувачів. Застосування сучасних алгоритмів шифрування дозволяє захищати окремі записи, що належать конкретній людині, від копіювання чи редагування іншими користувачами системи.

Концепція технології блокчейн запропонована Сатоші Накамото в 2008 році, а вперше застосована на практиці при появі біткойну в 2009-му. Через походження її відносять до транзакцій криптовалют, але сфера застосування технології помітно ширше. На прикладі проекту Bitcoin було показано, як організувати масове обчислення геша (нині загальновизнана міра цілісності електронного блоку).

Незважаючи на розподіл блоків в інтернеті, зашифрований доступ до кожного з них дозволяє утримувати в безпеці дані, що в них зберігаються. Сам ланцюг блоків може вільно передаватися будь-якому користувачеві інтернету без ризику втрати вмісту. На цьому базуються криптовалюти, що мають матеріальну цінність в національній валюті. Ключова особливість технології блокчейну полягає в децентралізації системи.

Якщо базу даних, розташовану на єдиному сервері, зламати теоретично можна — то з блокчейнами жоден з таких методів не спрацює. Блокчейн фіксує всі операції, які здійснюються з даними, і надає доступ до цих відомостей. Операції, які проводяться всередині блокчейн-системи, також називають транзакціями. Проведення транзакцій всередині системи влаштовано таким чином:

– користувач пересилає в систему запит на проведення транзакції, генеруючи і відправляючи особливий ключ, в якому зашифрована інформація про операції: тип, мета, сторони та інше;

– запит потрапляє в мережу, де аналізується і підтверджується достовірність;

– мережа верифікує транзакцію і підтверджує статус користувача за певним алгоритмом, після чого відбувається сам процес передачі інформації, наприклад, угода в криптовалюті або підписання контракту;

– після успішного проведення транзакції в ланцюг додається новий блок, який містить всі відомості про операції.

Блокчейн дозволяє вирішити відразу кілька проблем: скорочення часу проведення операцій та матеріальних витрат, позбавлення монополії великих компаній, які можуть маніпулювати ринком.

В роботі увага зосереджена на публічних блокчейнах, таких як Bitcoin або Ethereum, повністю децентралізованих, учасники яких є анонімними. Публічні блокчейни відрізняються від приватних блокчейнів, в яких центральний орган може санкціонувати учасників та засвідчувати транзакції. Кожний блок у блокчейні пропонує оновлену версію реєстру, з урахуванням нещодавніх транзакцій і прикутий до попередньої версії реєстру, тобто попереднього блоку. В ідеальному блокчейні існує одна послідовність блоків, що реєструє динаміку мережі, на яку всі учасники погоджуються. В реальному блокчейні можуть виникати конкуруючі гілки, кожна з яких реєструє потенційно іншу версію реєстра. Це явище називається форком.

Ключові учасники блокчейну — майнери. Саме майнери вирішують, який з ланцюгів є валідним і таким чином створюють єдиний ланцюг. У кожний момент часу майнери намагаються перевірити новий блок. Це передбачає вирішення чисто числової задачі, а сам процес називається майнінгом. Майнер, який вирішує задачу, отримує доведення роботи, яке він прикріплює до свого блоку, розповсюдженого через мережу.

Як стверджував Накамото, доведення роботи генерує стабільний консенсус, або іншими словами, єдиний ланцюг, якщо майнери завжди беруть останній знайдений блок як батьківський для наступного. Цей ідеальний блокчейн проілюстровано на рисунку 1.1:

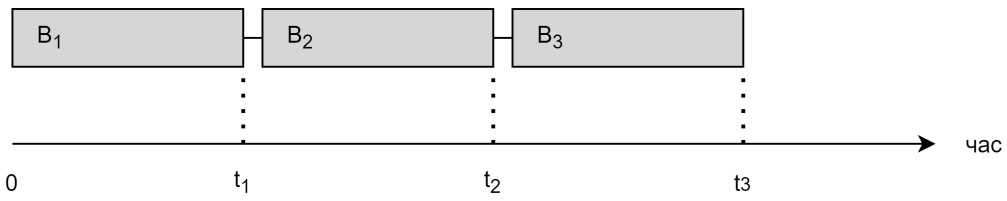

Рисунок 1.1 – блокчейн

При $t = 0$ існує початковий блок B_0 і запас транзакцій, включених в блок B_1, прикутий до B_0. Майнери працюють над валідацією блоку B_1 при t_1. B_1 транслюється для всіх вузлів, які перевіряють підтвердження роботи та висловлють прийняття, прив'язавши наступний блок до B_1.

Однак майнери можуть відмовитися від певних блоків. Припустимо, наприклад, що останнім вирішеним блоком є B_n, але майнер m прив'язує свій наступний блок до батькового B_n, тобто B_{n-1}. Це викликає форк, як показано на рисунку 1.2.

Якщо деякі майнери обирають за валідний ланцюг останній блок майнера m, а інші продовжують прикріплювати свої блоки до оригінального ланцюга — виникають конкуруючі версії реєстру. Навіть якщо, врешті-решт, всі майнери погоджуються приєднати свої блоки до одного ланцюга, поява форку не є безшкідливою. Блоки в недійсному ланцюгу були видобуті даремно, і відповідна обчислювальна потужність

та енергія були витрачені даремно. Більш того, транзакції, записані в неліквідних блоках, можуть бути поставлені під сумнів. Форк може також статися, коли деякі майнери приймають нову версію програмного забезпечення для майнінгу, несумісну з поточною версією.

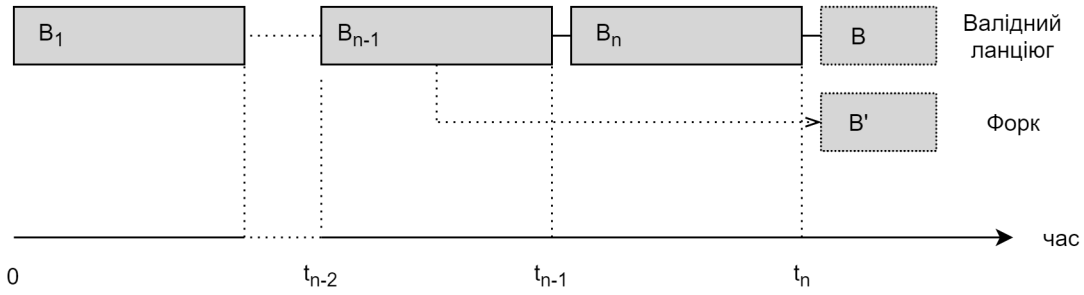

Рисунок 1.2 – форк

1.2 Що розуміється під криптовалютою

Криптовалюта — це нерегульований цифровий актив, що використовується в якості аналога валюти в обмінних операціях. Криптовалюта не має фізичної форми, вона існує тільки в елетронній мережі у вигляді даних, її створюють і передають переважно на базі технології блокчейн.

Проведені транзакції захищаються криптографічними методами. Без підтвердження транзакції мережею неможливо провести фінансову операцію, якщо ж це зроблено — ніхто не зможе скасувати переведення, що виключає шахрайство при оплаті криптовалютою товарів і послуг.

До переваг криптовалюти можна віднести наступні фактори: анонімність, незалежність, надійність, прозорість, швидкість, кількісна обмеженість, саморегуляція. До недоліків — волатильність, незабезпеченість і недостатню безпечність збереження.

1.3 Майнінг та його значення для блокчейнів

Майнінг — це процес видобутку криптовалюти шляхом створення нових блоків в блокчейні. Зазвичай майнінг зводиться до серії обчислень з перебором параметрів для знаходження геша з заданими властивостями, такі обчислення використовуються алгоритмами криптовалют для забезпечення їх функціонування. Учасник мережі отримує винагороду у вигляді комісійних зборів або за рахунок емітованих монет криптовалюти.

По суті, майнери займаються перевіркою нескінченного потоку транзакцій, використовуючи обчислювальні ресурси, в основному фізичного обладнання. Чим цих ресурсів більше, тим краще результат, тому не дивно, що він може відбуватися в промислових масштабах. Спочатку, майнінг працює виключно на досягнення самоокупності, термін якої залежить від складності алгоритму (безпосередньо від криптовалюти). З одного боку, криптовалюти породили такий вид діяльності, як майнінг. З іншого боку, емісія нових монет неможлива без майнінгу, так як часто мережею підтримується тільки одна технологія створення нових блоків та емісії.

Існують два різновиди майнінгу — соло-майнінг та майнінг у складі пула, які будуть детально розглянуті в наступному розділі. Так як соло-майнінг наразі не є реальним для рядового користувача (майнера), предметом подальшого дослідження є саме оцінювання гешрейта майнера, що працює віддалено в складі пула.

1.4 Протоколи консенсусу

Поява, розвиток і стрімке зростання популярності проекту Bitcoin дозволили зробити значний ривок технології DLT, основою якої є протокол консенсусу [1]. У широкому сенсі, консенсус — це угода, яка задовольняє всі залучені сторони. Це ключ до децентралізації в цілому, і

до технології розподіленого реєстру зокрема. Консенсус є процедурою прийняття рішення. Його мета — забезпечити узгодження поточного стану після додавання в мережу нової інформації — блоку або транзакцій, між усіма учасниками. Введемо поняття задачі консенсусу:

Означення 1.1. Задача консенсусу: є декілька процесів, у кожного з них є певні дані — пропозиції *(proposal)*, вони мають виконати деякий розподілений алгоритм і прийти до вирішення *(decision)*. Необхідно:

– узгодженість *(agreement)*: всі працюючі процеси повинні завершитися з рішенням *(decide)* і всі ці рішення повинні співпадати;

– нетривіальність *(non-triviality)*: повинні існувати варіанти виконання, які призводять до різних рішень;

– завершеність *(termination)*: всі процеси повинні завершитися за кінечний час.

1.4.1 Вимоги до протоколів досягнення консенсусу

Відсутність центральної довіреної сторони. Мережа складається з рівноправних вузлів. Якщо зловмисники спробують вивести з ладу певну кількість вузлів — мережа повинна продовжувати нормально працювати, поки доброчесні учасники контролюють більшість вузлів.

Чесні учасники не знають, які вузли контролюються зловмисниками. Передбачається, що деяка кількість вузлів може в довільний момент часу виходити з ладу або координуватися зловмисниками для проведення атаки на мережу, при цьому чесні учасники не знають, які з вузлів чесні, а які ненадійні або підконтрольні.

Передбачається, що мережа *завідомо ненадійна.* У мережі можливі затримки, мережа може піддаватися атаці, але тим не менш, в таких умовах децентралізований консенсус повинен нормально функціонувати: всі чесні вузли повинні приходити до одного й того ж стану.

Протоколи повинні бути повністю формальними. Не повинно бути ніякого додаткового впливу людини і ніяких додаткових даних не потрібно. Всі чесні вузли повинні повністю дотримуватися одного і того ж алгоритму, таким чином приходячи до спільного рішення.

1.4.2 Огляд існуючих протоколів консенсусу

У блокчейн проектах представлено досить багато алгоритмів консенсусу, що дозволяють вибрати, хто є найбільш придатним вузлом для валідації наступного блоку.

Proof-of-Work (PoW) [2]. Алгоритм консенсусу PoW є найбільш відомим способом підтвердження транзакцій, в силу його використання в проекті Bitcoin. Основна ідея полягає в тому, щоб вузли мережі, виконували досить складну обчислювальну роботу, результат якої може бути легко і швидко перевірений іншими вузлами. Перший вузол, який повністю провів всі необхідні обчислення, отримує винагороду.

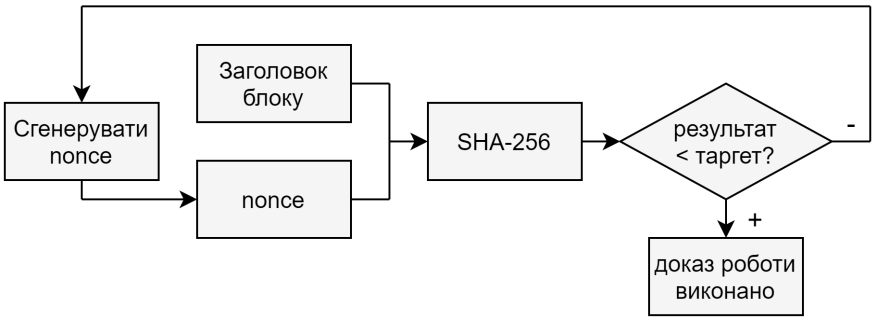

Рисунок 1.3 – граф потоку керування для алгоритма PoW

Proof-of-Stake (PoS) [3]. Один з найпопулярніших алгоритмів в блокчейн мережах. У цьому алгоритмі валідатором наступного блоку вибирається вузол, який володіє найбільшим балансом. За створення блоку вузол винагороду не отримує, вона виплачується за проведення транзакції.

Delegated Proof-of-Stake (DPoS) [3]. Різновид алгоритму PoS, в якому блоки підписують обрані представники. Користувачі мережі вибирають представників, кожен з яких отримує право підписувати блоки в мережі. Кожен представник, що володіє одним або більше відсотками від всіх голосів, потрапляє до «ради». З сформованої ради по колу вибирається наступний представник, який і підпише наступний блок. Учасники пула, делегуючи свої голоси, ні в якому разі не втрачають над ними контролю, так як можуть відкликати їх в будь-який момент.

Leased Proof-of-Stake (LPoS) [3]. Ще одна модифікація алгоритму PoS. На даний момент підтримується тільки платформою Waves. В рамках цього алгоритму, будь-який користувач має можливість передавати свій баланс в оренду PoS-майнінг-вузлів, а за це майнінг-вузли діляться частиною прибутку з користувачами.

Proof-of-Capacity (PoC) [4]. PoC працює за наступним принципом:

– кожен майнер обчислює досить великий обсяг даних, який записується на дискову підсистему вузла;

– для кожного нового блоку в блокчейні, майнер читає невеликий набір даних від свого загального обсягу і повертає результат, як минулий час з моменту створення останнього блоку;

– майнер, який отримав мінімальний час, підписує блок і отримує винагороду за транзакції.

Серед найвідоміших проектів, що використовують консенсус PoC (його також називають *Proof of Space*) — Chia Coin, що спричинив справжній дефіцит швидкісних SSD-накопичувачів.

Proof-of-Importance (PoI) [3]. Використовується блокчейн платформою NEM. Значимість кожного користувача в мережі визначається кількістю засобів наявних у нього на балансі і кількістю

проведених транзакцій. Такий підхід залучає користувачів не просто тримати кошти у себе на рахунку, а й активно використовувати їх.

Proof-of-Activity (PoA) [5]. Алгоритм створений як потенційно новий і більш надійний алгоритм для біткоїну. Автори спробували об'єднати два найбільш популярних алгоритма, з метою збільшення рівня захисту. У PoA процес майнінгу починається за стандартним алгоритмом PoW, коли новий блок знайдено, система переключається на PoS, при цьому знову знайдений блок містить тільки заголовок і адресу. Як тільки всі валідатори підписують блок, він набуває статусу повного блоку, ідентифікується і додається в мережу.

Proof-of-Burn (PoB) [6]. При використанні даного алгоритму майнер здійснює транзакцію на випадкову адресу згенерованого геша. За таке спалювання ресурсів, майнер отримує постійний шанс знайти PoB блок і отримати за нього винагороду.

Proof-of-Research (PoR) [3]. Алгоритм Proof of Research був розроблений в проекті GridCoin для того, щоб направити обчислювальні потужності PoW-мереж на розв'язок наукових завдань на платформі BOINC, тим самим зробивши виконання роботи корисним не тільки для існування мережі. У Proof of Research одночасно використовується PoW для винагороди учасників за виконані обчислення і PoS для заохочення довгострокової участі в проекті.

Більшість платформ використовує PoW і PoS, а також їх модифікації. Одним з найбільш оптимальних є алгоритм DPoS, перевагами якого є збільшена швидкість підписання блоків і велика енергоефективність, проте слово *delegated* в назві протоколу вже ставить під питання істинність децентралізації в подібних системах, через необхідність вибирати і довіряти. Цікавим з точки зору економіки є PoI.

1.5 Атаки на блокчейни

В даному підрозділі розглянуто атаки, які можливо провести на блокчейн мережі. Важливо зауважити, що атаки на блокчейни не мають такого впливу, як в разі централізованих мереж, тому що в розподіленій мережі можна атакувати лише окремі вузли або майнінгові пули. Ще один аргумент на користь децентралізованих мереж — на сьогоднішній день, для успішного проведення атаки, зловмисник повинен володіти дуже значними ресурсами, як обчислювальними так і матеріальними.

Атака Сивіли [7] — це атака, при якій зловмисник заповнює мережу безліччю підконтрольних вузлів і намагається захопити всі сусідні вузли жертви, оточуючи її. Отримавши контроль над усіма сусідніми вузлами можна:

– блокувати всі вхідні і вихідні транзакції;

– приєднувати жертву тільки до блоків, які створює атакуючий в альтернативному ланцюжку;

– уникати справжніх вузлів в мережі, якщо створюється достатня кількість підроблених або ідентичних.

Блокчейни, що використовують консенсус PoW, PoS або DPoS фактично не захищені від атаки Сивіли, вони просто роблять її недоцільною навіть в разі успішного виконання.

Атака 51% [7] — це атака, яка передбачає, що у зловмисника зосереджено більше половини обчислювальної потужності мережі блокчейна, тобто більше, ніж у всій решті мережі. Глобально, при виконанні цієї умови, захоплюється контроль над підтвердженням транзакцій і генерацією блоків. Маючи в розпорядженні 51% потужності, атакуючий може:

– не давати іншим майнерам знаходити блоки *(selfish mining)*;

– проводити подвійну витрату монет *(double-spending)*;

– робити форк основного блокчейну;

– не дозволяти транзакціям або блокам підтверджуватися;

– отримувати всі нагороди за блок і комісії з транзакцій;

– зменшувати і маніпулювати складністю мережі;

– видаляти історію транзакцій за рахунок відкату старих блоків і редагування списку транзакцій, переотримувати нагороди за старі блоки.

Будь-яка блокчейн мережа, незалежно від консенсусу, який використовується, потенційно вразлива до атаки 51%. Захиститися від такої атаки неможливо, її можна тільки запобігти шляхом моніторингу мережі.

Егоїстичний майнінг — стратегія, що дозволяє майнеру збільшити свій прибуток за рахунок приховування блоків від публічної мережі. Майнер не відправляє блок в мережу кожен раз після генерації, а продовжує майнити нові блоки, поверх будь-яких самостійно знайдених. Поки конкуренти майнять поверх старіших блоків, егоїстичний майнер отримує перевагу. Якщо зловмиснику вдалося зробити таємний ланцюжок довшим за публічний — він випускає його в мережу, щоб отримати нагороди за знаходження блоків і комісії користувачів.

Атака подвійної витрати [8] — ситуація в децентралізованих системах, коли користувач намагається використовувати раніше витрачені кошти з метою отримання особистої вигоди. Зазвичай, мережа не прийме таку транзакцію як дійсну, але при накопиченні достатньої потужності, зловмисник може створити форк, і в ланцюгах можуть перебувати транзакції, які по різному розпоряджаються одними і тими ж ресурсами.

Уявімо, що зловмисник розплачується за товар або послугу з продавцем, той приймає оплату в криптовалюті і угода майже завершена. Транзакція відправляється в загальний блокчейн і підтверджується. Коли

покупець переконаний, що угода завершена, він відкатує блокчейн до попереднього стану, тим самим повертаючи монети на свій рахунок. Інший, більш прихований варіант — атакуючий майнить паралельний ланцюжок блоків, куди замість чесної транзакції була включена транзакція подвійної витрати. Така транзакція, найчастіше, відправляє ті ж монети на іншу адресу, яка також належить шахраю. Залишається надати валідному ланцюгу альтернативну послідовність блоків, в очікуванні, що мережа їх прийме.

Нехай транзакція має N підтверджень, якщо вона включена в блок, який є частиною основного ланцюга, і існує N блоків, включаючи даний, і всі наступні, що йдуть від нього. Убезпечити транзакцію від double-spending можна при достатній кількості таких підтверджень. Наприклад, в мережі Bitcoin на сьогоднішній день потрібно 6 підтверджень, таким чином, якщо атакуючий буде володіти 10% потужності мережі, ймовірність успішної атаки подвійної витрати становитиме всього 0.06%. Якщо ж зловмисник буде мати більше половини потужності мережі — він завжди зможе успішно здійснити атаку, незалежно від кількості підтверджень.

Також, до ряду атак можна зарахувати розповсюдження шкідливого майнінгового програмного забезпечення та DoS/DDoS атаки.

Висновки до розділу 1

У цьому розділі були описані базові поняття блокчейну, криптовалюти, майнінгу. Сформульовано та описано вимоги до протоколів консенсусу, проведено огляд найбільш популярних серед них. Також були описані найпоширеніші атаки на блокчейн-мережі, можливості атакуючого в разі успіху та наслідки для мережі. Подальше дослідження буде розглядатися в контексті протокола Proof of Work, в якості мережі буде обрано Bitcoin, а майнер працюватиме у складі пула.

2 ІСНУЮЧІ МЕТОДИ ВИЗНАЧЕННЯ ГЕШРЕЙТА ТА РОЗПОДІЛУ ВИНАГОРОДИ В ПУЛАХ

У даному розділі поглиблено розглядається процес майнінгу, як соло, так і в складі пула, а також проблема «децентралізованого майнінгу в централізованих пулах». Розглянуто поняття гешрейта, визначення якого є центральною проблемою роботи, та методи його оцінки, наявні на теперішній час. Зокрема, був проведений аналіз наукової роботи на тематику кваліфікаційного дослідження.

2.1 Соло-майнінг

Процес розрахування геш-значень в індивідуальному порядку, з метою знайти валідний блок, винагорода за який буде повністю виплачена особі, що є власником пристрою, на якому було знайдено блок, називається соло-майнінгом. Станом на сьогоднішній день, соло-майнінг — це скоріше азартна гра, але вона може бути і більш вигідною, особливо для тих, хто займається майнінгом в промислових масштабах. По статистиці, соло-майнер, який хоче генерувати 1BTC за добу, повинен володіти трохи больше 0.11% від загального гешрейта біткойна (за даними blockchain.com приблизно 148 Ег/с). Оскільки навіть промисловий майнінг, що експлуатує приблизно 153 Пг/с, потребує на знаходження одного блока майже тиждень, майнери, ферми яких складаються всього з декількох пристроїв, скоріш за все, за десятки років не знайдуть ні одного блока, що робить таку діяльність абсолютно безглуздою.

2.2 Майнінговий пул

Поодинці, ймовірність отримати хоч якусь винагороду, навіть за великий проміжок часу дуже і дуже низька. Щоб підвищити шанси на отримання винагороди, майнери об'єднуються в пули [9]. Особливістю обчислень, які проводять майнери, є можливість застосувати максимальне розпаралелювання процесів, коли кожен учасник пула шукає свій варіант вирішення, без узгодження своїх дій з діями інших учасників. Для цього достатньо лише забезпечити виключення дублювання розрахунків одних і тих же параметрів різними учасниками.

З точки зору криптовалютної мережі, пул виглядає як дуже потужний майнер, але за рахунок своєї сукупної потужності, ймовірність отримання нагороди у пулі дорівнює сумі ймовірностей отримати нагороду кожного з його учасників. Винагорода розподіляється серед учасників пула в залежності від ефективності витрачених ресурсів.

На рисунку 2.1 представлений розподіл обчислювальної потужності пулів, що працюють в мережі Bitcoin станом на момент написання роботи (за даними blockchain.com).

Майнінг-пул зазвичай підтримується оператором пула, який отримує певну долю винагороди за блок в якості комісії за підтримку функціонування пула. На практиці всі майнери платять однакову комісію, незалежно від їх гешрейтів або надісланих в пул шар. Іншими словами, не існує цінової дискримінації з точки зору сплати пул-комісії. Крім того, різні пули також відрізняються за тим, як вони розподіляють комісію за транзакції в блоці. Хоча більшість пулів забирають комісію за транзакції та розподіляють лише винагороду за створений блок, враховуючи зростання комісій за транзакції останнім часом, більше пулів тепер також розподіляють комісії за транзакції.

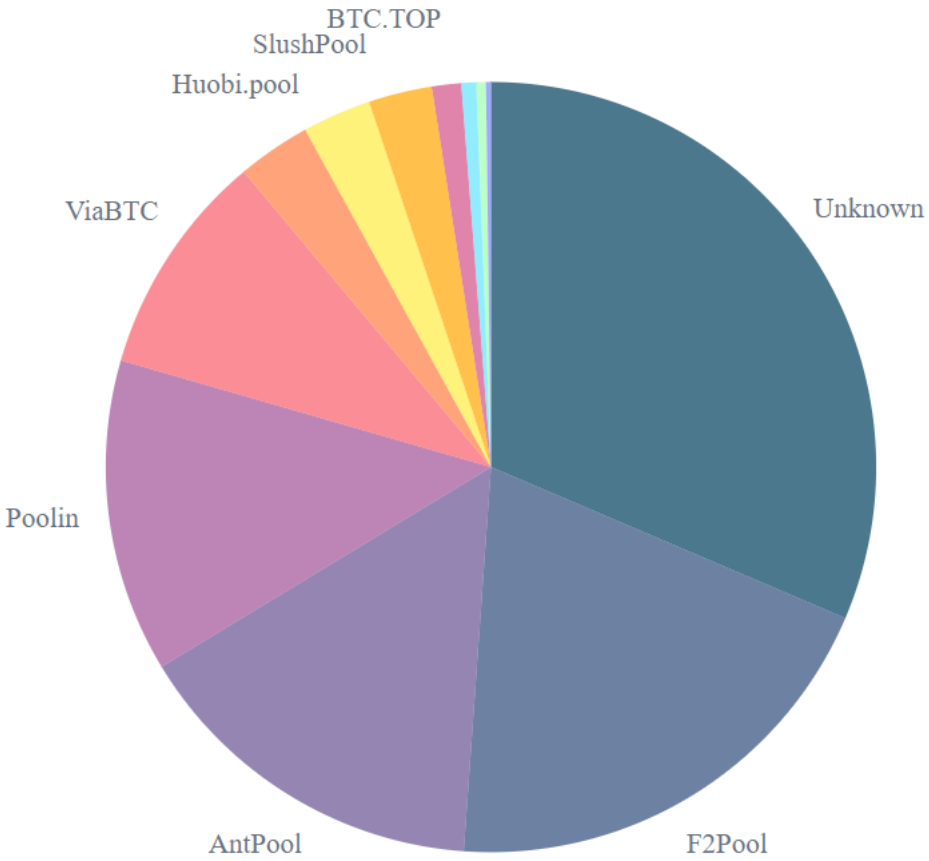

Рисунок 2.1 – Розподіл обчислювальної потужності пулів

2.2.1 Проблема децентралізації

Однією з найважливіших переваг блокчейнів є децентралізація. Але враховуючи описані вище недоліки соло-майнінгу, майнери з невеликими потужностями змушені об'єднуватися в пули. Так як кількість майнерів в пулі може необмежено зростати — це ставить під загрозу децентралізованність блокчейн-систем, що використовують алгоритм консенсусу PoW. Це явище називається проблемою *«децентралізованого майнінгу в централізованих пулах»* [10].

Це питання вже не перший рік турбує крипто-спільноту, яка просуває ряд причин, за якими розмір майнінг-пула повинен бути контрольованим:

– *ідеологія*: майнери, принаймні на початку історії Bitcoin, як правило, притримуються «криптоанархізма» — для них централізація суперечить ідеології. Але ідеологія навряд чи буде причиною першого порядку, оскільки Bitcoin перетворюється на індустрію з трильйонною капіталізацією;

– *саботаж*: так само, як відмова одного контролюючого вузла в традиційних централізованих системах, надвеликі майнінгові пули також потенційно можуть зашкодити мережі в разі відключення, або навіть проводити атаки на мережу, такі як атака розподіленої відмови в обслуговуванні (DDoS) та Атака 51%;

– *криза довіри*: аргументовано, що вартість криптовалют базується на тому, що це децентралізована система. Надмірна централізація в єдиному пулі може призвести до зниження вартості криптовалюти мережі, що не відповідає інтересам пула в тому числі. Однак емпіричних доведень цього аргументу недостатньо.

2.3 Схеми розподілу винагороди

Кожний майніговий пул розподіляє і виплачує винагороду майнерам, дотримуючись певної стратегії виплат [11]. Існує досить багато таких стратегій оцінки роботи майнера, кожна з яких має певні переваги і недоліки. Далі будуть розглянуті найбільш популярні і широко розповсюдженні.

Введемо наступні позначення:

– B (Block reward) — величина винагороди за блок;

– F (Fee) — комісія пула, фіксована частина від нагороди за блок;

– D (Difficulty) — складність, періодично корегується мережею;

– p (probability) $= \frac{1}{D}$ — ймовірність того, що шара буде рішенням блоку.

Pay Per Share (PPS). Пул виплачує нагороду відразу після отримання шари від майнера. Для майнера дана система цікава тим, що пул в будь-якому випадку винагороджує за виконану роботу. Для пула ризик полягає в тому, що до знаходження блока, йому необхідно робити виплати зі свого балансу. Коли учасник відправляє шари — він негайно отримує винагороду, $(1-F)pB$. Щоб компенсувати свій ризик, такий пул буде виставляти значно вищу комісію, ніж при інших методах, і це недолік PPS для майнерів. Якщо пул неправильно врівноважує комісію зі своїми фінансовими резервами — пул має шанси збанкрутувати.

Щоб зберегти ймовірність банкрутства на рівні θ, пул повинен мати резерв не менше ніж:

$$R = \frac{B \ln \frac{1}{\theta}}{2F}$$

Приклад 2.1. При нагороді за блок $B = 6.25$BTC та комісії $F = 0.05$, для підтримання ймовірности банкрутства $\theta = 1\%$ необхідно володіти резервом не менше ніж:

$$R = \frac{6.25 \ln 100}{2 \cdot 0.05} = 62.25 \ln 100 \approx 288 BTC \approx 10.3 \text{ млн. дол.}$$

Pay Per Last N Shares (PPLNS). В цій стратегії нагорода розраховується після знаходження нового блоку, але виплати проводяться після знаходження наступного блоку та оцінюється процентний внесок кожного майнера в розрахунок останніх N шар. При цьому, не має значення кількість шар, яка виявилася необхідною для знаходження нового блоку. Всі попередні «досядження» по знаходженню блоку також враховуються у розрахунку. Така система виплат стійка до *pool hopping* [12], на відміну від PPS, і сума винагороди може бути більше, за рахунок внеску у знаходження попередніх блоків. Технічно, дана система вважається однією з найскладніших, але вона вигідна як для пула, так і для майнерів, які вважають за краще не змінювати пул. Найпростіший варіант — вибрати фіксоване число шар N і виплатити винагороду в

розмірі $\frac{(1-F)BL}{N}$ за кожну з них, де L — кількість рішень блоків, знайдених в наступних N шарах. Кожна шара має ймовірність p бути рішенням, незалежно від минулої, тому L слідує розподілу Пуассона з параметром $\lambda = pN$.

Зауваження. *pool hopping* — це перехід з пула на пул з метою знайти *короткі блоки*, тобто такі блоки, для знаходження яких потрібна менша кількість шар, за допомогою спеціальних програмних засобів. Якщо блок буде коротким, а потужність пула — невеликою, прибуток майнера за цей блок буде більшим, за рахунок проведення меншої кількості обчислень.

Maximum PPS (MPPS). У цьому методі для кожного учасника зберігається два баланси — баланс PPS і пропорційний баланс. Коли майнер відправляє шару, його відповідний PPS-баланс збільшується, так наче це б був PPS-пул. Коли пул знаходить блок, пропорційний баланс кожного учасника збільшується, як якщо б це був пропорційний пул. Ідея цієї системи полягає в тому, що, оскільки загальний пропорційний баланс всіх учасників дорівнює загальній винагороді, а загальна виплачена сума менше ніж ця кількість — пул завжди отримує прибуток.

Зауваження. Пропорційний пул — це найпростіша та найперша система винагород. Коли блок знайдено, пул отримує винагороду B, оператор зберігає за собою комісію в розмірі FB, і решта $(1 - F)B$ винагороди розподіляється між майнерами, прямопропорційно кількості шар, яку вони надіслали за час від попереднього до поточного блоку. Якщо майнер надіслав n шар, і загальна кількість шар під час цього раунду рівна N — його виплата буде визначатися як $\frac{n}{N}(1 - F)B$.

Shared Maximum PPS (SMPPS). Суть цієї стратегії в тому, що майнери отримують виплати також, як по системі MPPS, але із затримками в нарахуванні нагороди. Вони необхідні для того, щоб пул зміг знайти нові блоки. Якщо в данний момент пул володіє позитивним балансом — він розраховується в повній мірі з учасниками за новий блок. Але якщо баланс пула порожній, то винагороди виплачуються

пропорційно між усіма учасниками. Якщо в пулі після виплат по MPPS залишаються кошти, вони зберігаються на балансі пула для майбутніх виплат. Різниця між SMPPS заробітком майнера і заробітком по MPPS зберігається за ним і оплачується в подальшому, при знаходженні нових блоків.

2.4 Складність, гешрейт та його розрахунок пулом

Складність. У відповідності до роботи Накамото [2], час t, необхідний майнеру m для знаходження блоку — випадкова величина з експоненційного розподілу з параметром θ_m ($t \sim Exp(\theta_m)$). Таким чином, θ_m — це миттєва ймовірність того, що m знаходить блок, який він видобуває. θ_m визначається не тільки індивідуальною обчислювальною потужністю майнера, або гешрейтом, а й складністю майнінгу [13], що встановлюється мережевим протоколом (D):

$$\theta_m = \frac{h_m}{D} \tag{2.1}$$

Складність встановлюється протоколом блокчейну так, щоб очікуваний час знаходження нового блоку був константним — X (для Bitcoin $X = 10$ хвилин, тоді як для Ethereum це наразі 13 секунд).

Таким чином, використовуючи властивості експоненціальних розподілів, складність D встановлено так, що:

$$X = \frac{1}{\sum\limits_{i \in M} \theta_i} = \frac{1}{\sum\limits_{i \in M} \frac{h_i}{D}} = \frac{D}{\sum\limits_{i \in M} h_i} = const, \text{ де } M \text{ — множина майнерів} \tag{2.2}$$

Означення 2.1. Складність — це змінна, яка регулює швидкість знаходження блоку. З рівняння (2.2) випливає, що складність D

$$D = X \sum_{i \in M} h_i \tag{2.3}$$

Для мережі Bitcoin складність перераховується кожні 2016 блоків, або приблизно кожні два тижні, щоб забезпечити збереження середнього часу між блоками рівному 10 хвилин. На рисунку 2.2 представлений графік складності мережі Bitcoin, за даними blockchain.com (ціна поділки на осі ординат — 5 трильйонів):

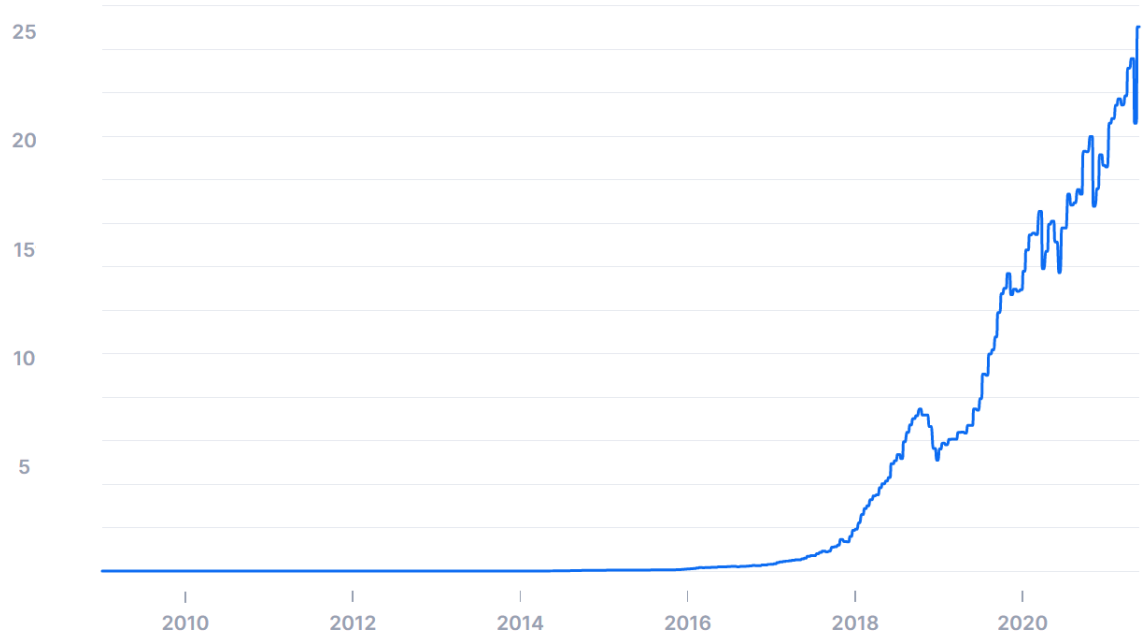

Рисунок 2.2 – масштабування складності з часом

Разом з (2.1) та (2.3) можна отримати наступне співвідношення:

$$\theta_m = \frac{h_m}{D} = \frac{1}{X}\frac{h_m}{\sum\limits_{i \in M} h_i} \tag{2.4}$$

З рівняння (2.4) випливає, що миттєва ймовірність знайти блок для майнера m зростає при збільшенні власних обчислювальних потужностей, але при цьому зменшується ймовірність інших майнерів при сталих умовах.

Гешрейт. Для того, щоб зрозуміти технічні аспекти, потрібно знати, як працює геш-функція Bitcoin, які дані пул отримує від майнерів, і як пул оцінює їх відповідні гешрейти.

Означення 2.2. Шара — це одиниця, яку майнінг-пули використовують для підтвердження роботи, виконаної майнером, що являє собою можливе рішення блоку.

Коли майнінгове обладнання підключене до пула, воно отримує обчислювальну задачу, яку потрібно вирішити, а саме — обчислення геш-значень з певними властивостями (вихідний геш повинен бути меншим за цільовий, що визначається складністю). Геш-значення, які задовольняють вимозі, надсилаються назад в пул і використовуються в якості доведення роботи майнера. Кількість «робіт» майнера реєструється в одиницях, що називаються шарами.

Функція гешування, в контексті Bitcoin, представляє собою функцію подвійного $SHA-256$ [14], яка приймає блок даних і повертає число фіксованої довжини зі значенням від 0 до $2^{256}-1$. Кожне вихідне геш-значення h_{out} має однакову ймовірність:

$$P(h_{out}) = \frac{1}{2^{256}-1}$$

Щоб додати новий блок в блокчейн біткойна, в якості вхідних даних для геш-функції використовується пакет даних транзакцій, які стоять в черзі в мемпулі і очікують підтвердження. Задача майнера складається не тільки в створенні геша цих вхідних даних, а і в створенні геша, який починається з ряда провідних нулів, кількість яких власне і визначається складністю, і відповідає пакету вхідних даних. Незважаючи на те, що неможливо виконати обернену операцію і знайти дані, які згенерували геш-значення, завжди можна повторити операцію, щоб переконатися, що вона була виконана правильно. Таким чином, оскільки всі дані по транзакціям в блоці являються публічними, будь-хто може перевірити за допомогою $SHA-256$, що геш, обчислений майнером, є правильним. І тільки якщо геш був обчислений коректно, блок буде додано до блокчейну, і майнер(пул) отримає свою винагороду.

Задача 2.1. Як саме виконується розрахунок гешрейта майнера в пулі?

1. Пул відправляє роботу майнеру у вигляді майже повних шаблонів заголовків блоків. У заголовку відсутня невелика частина, яку необхідно заповнити «правильними» випадковими даними. По суті, майнінг — це процес багаторазового гешування заголовка блоку зі зміною цих випадкових даних до тих пір, поки результуюче геш-значення не буде задовольняти умову(таргет), що визначається складністю мережі і регулярно коригується, щоб час генерації блоку становив 10 хвилин.

2. Якщо майнер знаходить геш-значення менше, ніж таргет — він посилає знайдену шару в пул. За цією інформацією пул може перевірити, що майнер працює, і обчислити гешрейт такого майнера. Припустимо, що майнер відправив $s = shares(t)$ шар за проміжок часу t, тоді пул може легко оцінити гешрейт майнера за наступним співвідношенням [15]:

$$h = \frac{2^{32} \cdot s}{t} \tag{2.5}$$

Зауваження. У минулому значення 2^{32} було вибрано довільно, але воно є стабільним і дозволяє чітко визначити частку однієї шари в гешрейті майнера.

Означення 2.3. Гешрейт — це кількість геш-значень, що генеруються за секунду. З фізичної точки зору являється швидкістю гешування.

Існують різні типи гешрейта — номінальний та ефективний [16]. Між номінальною, вказаною в технічних характеристиках пристрою для майнінгу, та ефективною швидкістю гешування, яку вираховує пул, є різниця, через яку в багатьох майнерів-новачків виникають питання до пул-операторів, які «неправильно оцінюють гешрейт».

Означення 2.4. Номінальний гешрейт — це кількість геш-значень, яку пристрій здатний генерувати незалежно від зовнішних факторів.

Означення 2.5. Ефективний гешрейт — швидкість гешування, обчислена на основі шар, поданих пристроєм до пула, при цьому лише незначна частина згенерованих шар надсилається, так як вони повинні відповідати певним критеріям, що призначені пулом.

Здебільшого ефективний гешрейт буде дещо нижчим від номінального, тому що ефективний гешрейт залежить від «удачі» майнінгового пристрою та якості з'єднання з сервером пула. В разі виникнення проблеми з підключенням, ефективна швидкість гешування буде вираховуватись як середнє значення за певні інтервали, що встановлюються пулом, але всеж буде відчутно нижчою за номінальну в той період часу.

2.5 Огляд альтернативного методу визначення гешрейта

Під час написання кваліфікаційної роботи була знайдена стаття [17], в якій описано відмінний від «класичного» (2.5), метод оцінювання гешрейта. Розглянемо хід міркувань авторів, використовуючи оригінальні позначення.

Пропонується періодично передавати в пул інформацію про виконання роботи (звіт), що дозволяє оцінювати швидкість гешування.

Маємо вибірку випадкових величин, взяту з дискретного рівномірного розподілу, значення яких знаходяться на інтервалі $[0, 2^{256} - 1]$. Якщо майнер m періодично оголошує найменше значення, яке він отримав — можемо оцінити швидкість гешування, $\hat{h_m}$.

Нехай I — часовий інтервал, що містить σ секунд. Протягом цього інтервалу майнер виконує θ_m операцій гешування, що генерує послідовність геш-значень $Z = \{Z_1, \ldots, Z_{\theta_m}\}$. Хоча σ відома, проте ми не знаємо θ_m апріорі. В кінці інтервалу майнер відправляє звіт $V = Z_{(1)}$, що позначає мінімальне геш-значення, отримане за час σ.

Іншими словами, V — випадкова величина, що представляє першу порядкову статистику виборки, отриманої після гешування θ_m разів.

Нехай $S = 2^{256} - 1$ — розмір геш-простору. Ймовірність того, що будь-яке V більше за деяке $v \in [0, S]$, дорівнює $1 - \frac{v}{S}$. Таким чином, імовірність того, що V завжди більша за v при виконанні θ_m незалежних випробуваннь, зроблених протягом часу I, дорівнює $(1 - \frac{v}{S})^{\theta_m}$.

Тепер визначимо $F_V(v|S)$ — функцію розподілу V. Очевидно, що

$$F_V(v|S) = 1 - (1 - \frac{v}{S})^{\theta_m}$$

Параметр θ_m можна подати як функцію від S, а саме:

$$\theta_m = \frac{S}{\beta} \tag{2.6}$$

Потрібно зрозуміти, як змінюється $F_V(v|S)$ із збільшенням S. Використовуючи загальну границю, отримаємо:

$$\lim_{S \to \infty} F_V(v|S) = 1 - e^{\frac{-v}{\beta}} \tag{2.7}$$

Таким чином, $V \sim Exp(\beta)$, де β — коєфіцієнт масштабу. Оскільки середнім значенням розподілу є β, можна побачити, що математичне сподівання V дорівнює β, що в свою чергу дорівнює $\frac{S}{\theta_m}$. Далі буде показано як отримати оцінку θ_m і у кінцевому підсумку оцінку $\hat{h_m}$.

Зауваження. Існує альтернативна параметризація экспоненційного розподілу, яка широко використовується в закордонних наукових роботах. Параметр β, що називається коєфієнтом масштабу, дорівнює $\frac{1}{\lambda}$, де $\lambda > 0$ — інтенсивність. В альтернативній параметризації функція розподілу набуває вигляду:

$$F(x) = \begin{cases} \frac{1}{\beta} e^{\frac{-x}{\beta}}, x \geqslant 0 \\ 0, x < 0, \end{cases}$$

Тепер нехай $V = \{V1, \ldots, V_n\}$, де $V_i \sim Exp(\beta)$ — вибірка випадкових величин, що представляють n звітів, кожен з яких

відправлений майнером через σ секунд. Так як всі випадкові величини незалежні і однаково розподілені, оцінка максимальної правдоподібності для β дорівнює вибірковому середньому ($\hat{\beta} = \overline{V}$). Оскільки $\beta = \frac{S}{\theta_m}$ та враховуючи (2.6), конзистентна оцінка для θ_m задається формулою:

$$\hat{\theta_m} = \frac{S}{\overline{V}}$$

Нарешті, гешрейт майнера $\hat{h_m}$ можна оцінити наступним співвідношенням:

$$\hat{h_m} = \frac{\hat{\theta_m}}{\sigma} = \frac{S}{\overline{V}\sigma} \tag{2.8}$$

Основна ідея, а саме оцінювання гешрейта через мінімальне геш-значення, також була використана при виконанні роботи, та описана у наступному розділі, проте ця робота містить багато недоліків, серед яких:

– відсутність доведень наведених тверджень та майже інтуїтивний хід описання;

– необґрунтованість тверджень та наслідків, наприклад, визначення параметра θ_m згідно (2.6), перевизначення якого буде означати хибність кінцевої оцінки гешрейта за формулою (2.8);

– відсутність математичних викладок та пояснень в ключових місцях роботи, таких як визначення функції розподілу (2.7) (в загальному збірнику стандартних математичних формул та таблиць [18] немає границі, яку автори назвали «загальною»), тому твердження про експоненційний розподіл V може бути хибним і, як наслідок, оцінка θ_m також;

– часова та обчислювальна складність не є низькою, адже для реалізації методу потрібно відправляти звіти через фіксовані інтервали часу, кожен з яких є першою порядковою статистикою, тобто мінімальним геш-значенням, які потрібно зберігати для подальшого розрахунку середнього значення всіх мінімумів.

Висновки до розділу 2

У цьому розділі було описано види майнінгу, вказано їх переваги та недоліки. Зокрема, була освітлена проблема централізації майнінгових пулів в децентралізованій мережі, детально розглянуті поняття складності та гешрейта, оцінка якого є провідною темою кваліфікаційної роботи. Також наведено описи, в тому числі формальний, наявних методів визначення та оцінки гешрейта, які є неефективними, саме тому, в наступному розділі, буде запропоновано новий метод оцінки гешрейта та доведено його коректність.

3 НОВИЙ МЕТОД ДИСТАНЦІЙНОГО ВИЗНАЧЕННЯ ГЕШРЕЙТА

Станом на момент написання роботи, при під'єднанні до пула майнеру необхідно, переважно протягом години, відправляти шари в пул для визначення та підтвердження свого гешрейта відповідно до (2.5), що є дуже незручним і економічно невигідним, адже ніякої винагороди за відправлені шари в період встановлення гешрейта майнер не отримує, навіть якщо одна з них буде валідною. У тому числі, недобросовісні оператори пулів можуть навмисно використовувати потужності «непідтверджених» майнерів для збільшення винагороди учасників пула.

У даному розділі буде розглянуто, яким чином можна визначати потужності майнера більш ефективно, знижуючи часові та фінансові затрати, тим самим оптимізувати процес під'єднання до пула.

Зокрема, у цьому розділі будуть вирішені наступні задачі:

– формально описати новий спосіб дистанційного визначення гешрейта;

– оцінити похибки, що можуть виникати в процесі;

– підтвердити отримані результати практично;

– провести кількісний та якісний порівняльний аналіз.

3.1 Формалізація нового методу визначення гешрейта

Ідея запропонованого методу базується на встановленні часового інтервалу, який є значно меншим за прийнятий зараз, і розрахунку майнером мінімального, за вказаний час, геш-значення, яке він повинен надіслати в пул для подальшого встановлення його гешрейта.

Нехай m — деякий майнер, та $n = n(t)$ — кількість операцій гешування, яку він зробив за деякий інтервал часу t. Позначимо довжину геш-значень l, та $\mathcal{N} = 2^l$ — кількість всіх можливих геш-значень.

Геш-значення, що є бітовим вектором, будуть розглядатися як ціле число від 0 до $\mathcal{N} - 1$. Нехай M — мінімальне геш-значення, яке майнер m отримує протягом вказаного часового інтервалу. Також введемо наступні величини: ζ_i, $i = \overline{1,n}$ — послідовність незалежних випадкових величин, що приймають значення з множини $\{0, 1, \ldots, \mathcal{N} - 1\}$. Випадкові величини ζ_i — це всі геш-значення, які були отримані майнером за часовий інтервал. Позначимо $\xi_i = \frac{\zeta_i}{\mathcal{N}-1}$, $i = \overline{1,n}$ — послідовність нормалізовних незалежних випадкових величин, що приймають значення на інтервалі [0;1]. Оскільки $\mathcal{N}$ досить велике, можемо припустити, що ξ_i — випадкові величини, що мають неперервний рівномірний розподіл на проміжку [0;1] ($\xi_i \sim U([0;1])$). Тоді справедлива наступна теорема:

Теорема 3.1. *Про оцінку гешрейта майнера*

В наших позначеннях,

$$P\left(\left|\frac{M}{\mathcal{N}-1} - \frac{1}{n+1}\right| \geqslant c\right) \leqslant \frac{1}{(nc)^2} \tag{3.1}$$

Доведення. Позначимо

$$\eta = min_{i=\overline{1,n}}\{\xi_i\} \tag{3.2}$$

Тоді для функції розподілу $F_\eta(x)$ виконується наступна рівність:

$$1 - F_\eta(x) = P(\eta > x) = P(\bigcap_{i=1}^{n}\{\xi_i \geqslant x\}) = \prod_{i=1}^{n} P(\xi_i \geqslant x) = (1-x)^n$$

Остання рівність справедлива, оскільки ξ_i, $i = \overline{1,n}$ незалежні та однаково розподілені випадкові величини. Отже,

$$F_\eta(x) = 1 - (1-x)^n \tag{3.3}$$

$$f_\eta(x) = n(1-x)^{n-1} \tag{3.4}$$

Зауваження. Відповідно до функції розподілу, $\forall x : P(\eta > x)$ зменшується зі збільшенням кількості гешувань n, тому чим більшим є значення n — тим меншим є значення η, тобто мінімальне геш-значення.

$$M\eta = \int_0^1 x dF_\eta(x) = \int_0^1 x f_\eta(x) dx =$$
$$= n\int_0^1 x(1-x)^{n-1}dx = n\frac{(n-1)!}{(n+1)!} = \frac{1}{n+1} \tag{3.5}$$

$$M\eta^2 = \int_0^1 x^2 dF_\eta(x) = \int_0^1 x^2 f_\eta(x) dx =$$
$$= n\int_0^1 x^2(1-x)^{n-1}dx = n\frac{2!(n-1)!}{(n+2)!} = \frac{2}{(n+1)(n-2)}$$

$$D\eta = M\eta^2 - (M\eta)^2 = \frac{2}{(n+1)(n-2)} - (\frac{1}{n+1})^2 =$$
$$= \frac{2(n+1)-(n-2)}{(n+1)^2(n-2)} = \frac{n}{(n+1)^2(n-2)} \approx \frac{n}{n^3} = \frac{1}{n^2} \tag{3.6}$$

Так як має місце середнє значення і дисперсія, використаємо нерівність Чебишова [19]:

$$P(|\eta - M\eta| \geqslant c) \leqslant \frac{D\eta}{c^2}$$

або, використовуючи (3.5) та (3.6), отримаємо:

$$P\left(\left|\eta - \frac{1}{n+1}\right| \geqslant c\right) \leqslant \frac{1}{(nc)^2}$$

Відповідно до визначення (3.2), η — це нормалізоване мінімальне геш-значення, тобто $\eta = \frac{M}{\mathcal{N}-1}$, і нерівність набуває вигляду:

$$P\left(\left|\frac{M}{\mathcal{N}-1} - \frac{1}{n+1}\right| \geqslant c\right) \leqslant \frac{1}{(nc)^2}$$

□

Наслідок 3.1. *Кількість n операцій гешування можна апроксимувати наступним співвідношенням:*

$$\frac{1}{n} \approx \frac{M}{\mathcal{N}-1} \tag{3.7}$$

де M — мінімальне геш-значення, $\mathcal{N} = 2^l$, а l — довжина геш-значень. При чому, чим більше n, тим кращим є наближення, згідно (3.1).

Виходячи з примітивного означення гешрейта та наслідку з теореми (3.1), можна сформулювати наступне твердження:

Твердження 3.1. *Використовуючи (3.7) можна знайти кількість гешувань n як обернену величину. Так як $n = n(t)$, а часовий інтервал t відомий, остаточна формула для розрахунку гешрейта майнера m набуває вигляду:*

$$h = \frac{n}{t} \approx \frac{\mathcal{N}-1}{Mt} \tag{3.8}$$

3.2 Визначення можливих похибок оцінки

Наступним питанням роботи є оцінка похибок, що можуть виникати при апроксимації кількості операцій гешування відповідно до (3.7) та подальшому розрахунку гешрейта майнера за формулою (3.8). Це є важливим економічним аспектом, адже від точності визначення гешрейта безпосередньо залежить винагорода майнера, особливо в пропорційних пулах.

В цілому, гешрейт може змінюватися постійно, але в нормі не повинен виходити за межі певного діапазону, в рамках похибки, яку закладають самі виробники, тому абсолютне відхилення в 5% буде вважатися допустимим. Отже, з більш формальної точки зору задача цього підрозділу може бути сформульована наступним чином:

Задача 3.1. Яка ймовірність помилкової оцінки потужностей майнера? Нехай два майнери m_1, m_2, отримують мінімальні геш-значення M_1 і M_2, відповідно. Яка ймовірність того, що $M_1 > M_2$, якщо майнер m_1 зробив n_1 гешувань, а майнер m_2 зробив n_2 гешуваннь?

Зауваження. Якщо два майнери, гешрейти яких відрізняються суттєво, будуть виконувати гешування за один і той же проміжок часу, то очікується, що саме майнер з більшим гешрейтом отримає менше геш-значення з більшою ймовірністю. Тобто, якщо $n_1 > n_2$, то $P(M_1 < M_2) > P(M_1 > M_2)$, саме тому має місце теорема:

Теорема 3.2 (Про ймовірність помилкової оцінки гешрейта)**.**

В наших позначеннях,

$$P(M_1 < M_2) = \frac{n_1}{n_1 + n_2} \tag{3.9}$$

Доведення. Позначимо η_1 та η_2 відповідно до (3.2), тобто η_i — це нормалізоване мінімальне геш-значення i-го майнера:

$$\eta_i = \frac{M_i}{\mathcal{N} - 1}, i = \overline{1, 2} \tag{3.10}$$

Тоді, використовуючи (3.3) та (3.4), отримаємо:

$$F_{\eta_1}(x) = 1 - (1 - x)^{n_1};$$

$$f_{\eta_1}(x) = n_1(1 - x)^{n_1 - 1};$$

$$F_{\eta_2}(x) = 1 - (1 - x)^{n_2};$$

$$f_{\eta_2}(x) = n_2(1 - x)^{n_2 - 1};$$

Отже, враховуючи (3.10),

$$P(M_1 < M_2) = P(\eta_1 < \eta_2) = \int_{x,y:0 \leqslant x,y \leqslant 1} f_{\eta_1}(x) f_{\eta_2}(y) dxdy =$$

$$= \int_0^1 \left(\int_0^y f_{\eta_1}(x)dx \right) f_{\eta_2}(y)dy = \int_0^1 \left(\int_0^y n_1(1-x)^{n_1-1}dx \right) n_2(1-x)^{n_2-1}dy =$$

$$= \int_0^1 F_{\eta_1}(y) f_{\eta_2}(y)dy = \int_0^1 \Big(1-(1-y)^{n_1}\Big) n_2(1-y)^{n_2-1}dy =$$

$$= n_2 \int_0^1 \Big((1-y)^{n_2-1} - (1-y)^{n_1+n_2-1}\Big)dy = n_2\left(\frac{1}{n_2} - \frac{1}{n_1+n_2}\right) =$$

$$= n_2 \frac{n_1+n_2-n_2}{n_2(n_1+n_2)} = \frac{n_1}{n_1+n_2}$$

□

З (3.9) випливає, що $P(M_1 < M_2) > \frac{1}{2}$, якщо $n_1 > n_2$. Якщо ж гешрейти майнерів приблизно однакові, то ймовірність (3.9) приблизно дорівнює $\frac{1}{2}$. У випадку, коли гешрейти майнерів значно відрізняються, наприклад, $n_1 >> n_2$, і нехай $n_1 = kn_2$, для достатньо великого k, маємо:

$$P(M_1 < M_2) = \frac{n_1}{n_1+n_2} = \frac{kn_2}{kn_2+n_2} = \frac{k}{k+1} \to 1, k \to \infty \qquad (3.11)$$

Приклад 3.1. при $k = 9$, за формулою (3.11),

$$P(M_1 < M_2) = \frac{9}{10} = 0.9$$

Отже ймовірність неправильної оцінки мінімальних геш-значеннь майнерів є досить малою, коли різниця їх гешрейтів достатньо велика.

Зауваження. Якщо гешрейти майнерів майже однакові, то не має необхідності розрізняти їх з великою точністю, адже вони отримають майже однакову винагороду.

3.3 Практичне підтвердження отриманих результатів

Для практичного підтвердження результатів, отриманих в цьому розділі, була розроблена програма, текст якої наведено у додатку А.1. В наших позначеннях, n — кількість операцій гешування з одної сторони, або кількість геш-значень, з іншої. Основна ідея полягає в наступному:

Алгоритм 3.1.

1) Виконати заздалегідь відому кількість гешувань n (практичне n);

2) використовуючи (3.7) знайти теоретичну кількість гешувань $n_{jt}, j = \overline{1,2}$ (спостережуване n);

3) повторити кроки 1) — 2) достатньо велику кількість разів;

4) усереднити отримані значення n_{jt} (наближене n):

$$\overline{n_{jt}} = \frac{\sum\limits_{i=1}^{n} n_{jti}}{n};$$

5) оцінити похибку $\overline{n_{jt}}$ відносно n, яку ми будемо позначати δn_{jt}, що визначається наступним співвідношенням:

$$\delta n_{jt} = \frac{|\overline{n_{jt}} - n|}{n};$$

6) повторити кроки 1) — 5) для двох «майнерів» у 4 випадках:

а) $n_1 = n_2$;

б) $n_1 \approx n_2$;

в) $n_1 > n_2$;

г) $n_1 >> n_2$.

Зауваження. Практично, буде оцінюватися величина $\frac{1}{n}$, так як для невеликої кількості гешувань оцінка не є достатньо точною. Також слід враховувати, що за наслідком (3.7) збільшення кількості хешувань призводить до зменшення відносної похибки.

Всі розрахунки були проведені за допомогою **Google Colaboratory** — сервісу хмарних обчислень. Отримані результати представлені на рисунках 3.1 — 3.8:

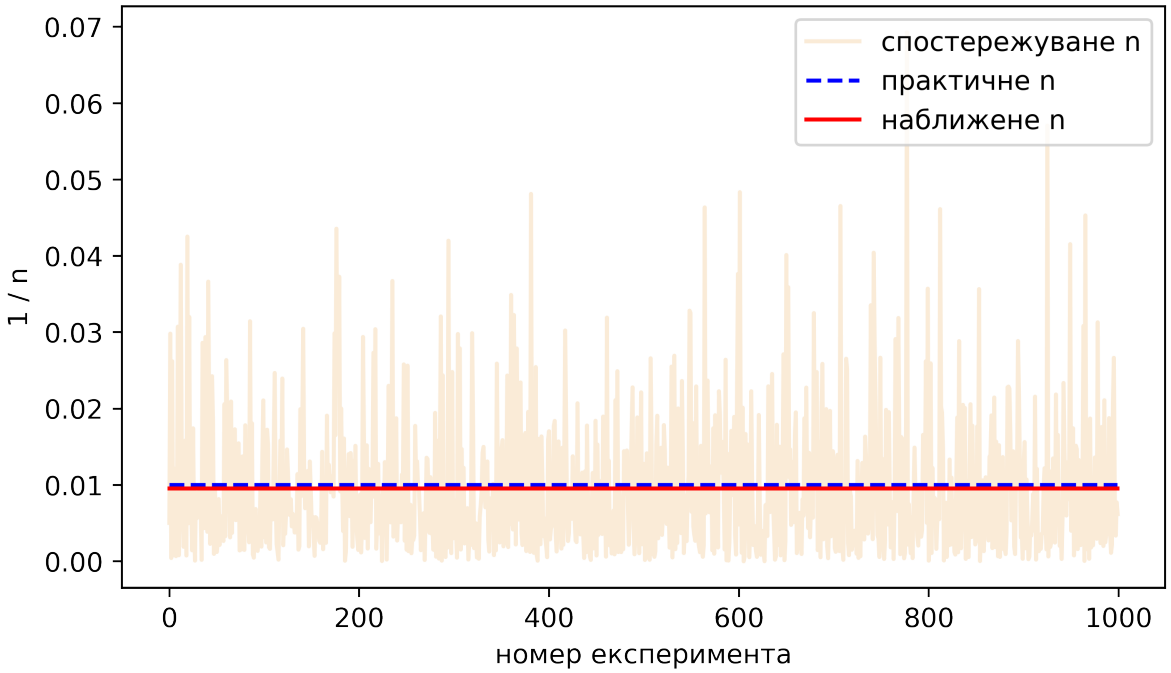

Рисунок 3.1 – Випадок а) $n_1 = 100$, апроксимація n_1, $\delta n_t = 3.68\%$

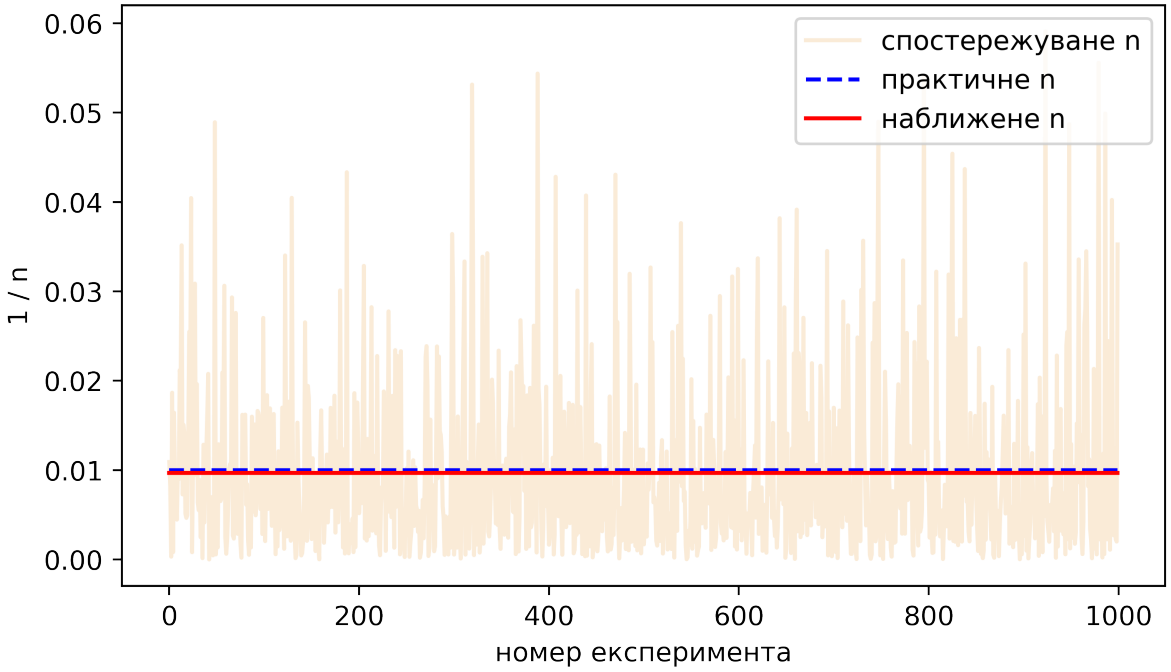

Рисунок 3.2 – Випадок а) $n_2 = 100$, апроксимація n_2, $\delta n_t = 3.508\%$

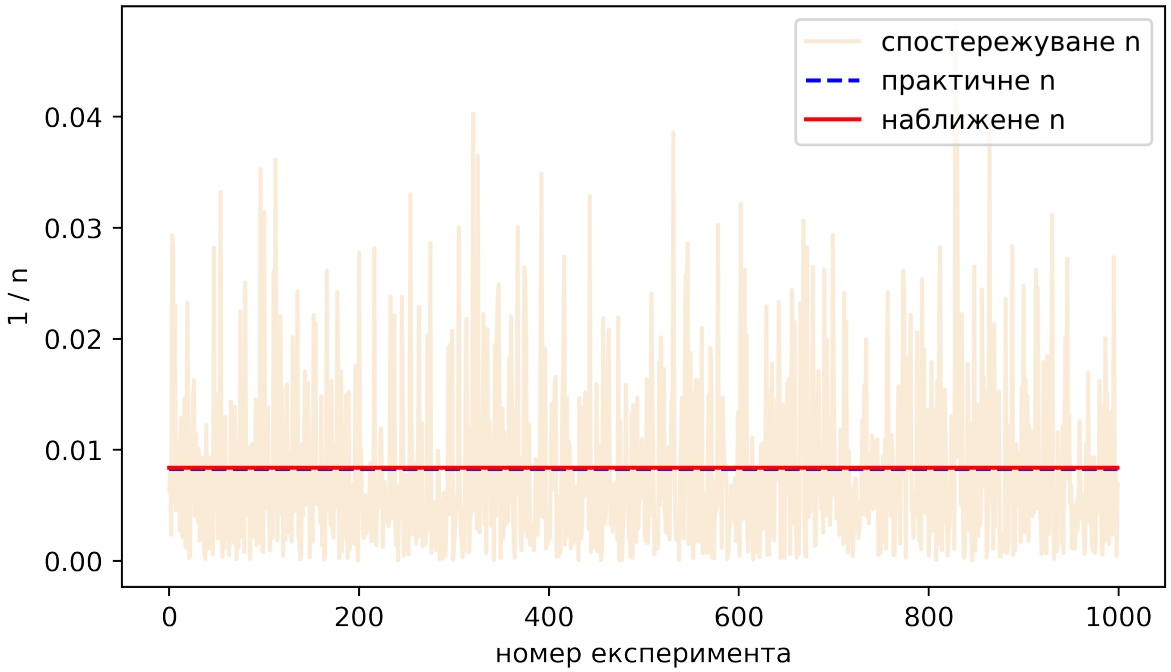

Рисунок 3.3 – Випадок б) $n_1 = 121$, апроксимація n_1, $\delta n_t = 3.376\%$

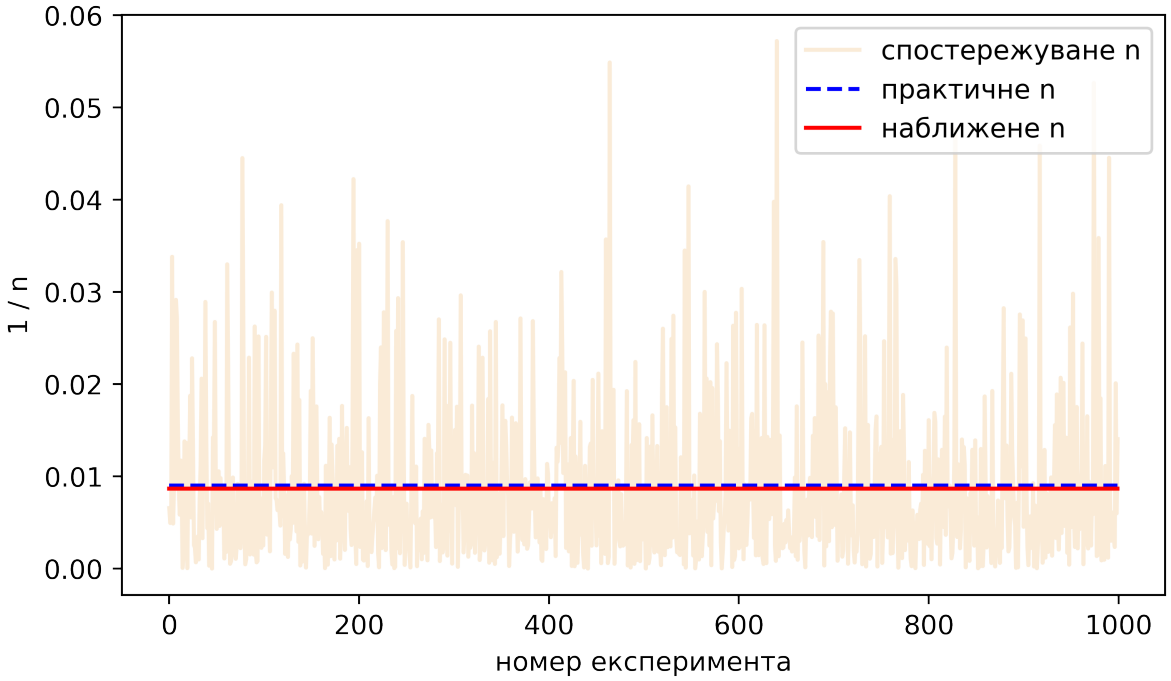

Рисунок 3.4 – Випадок б) $n_2 = 111$, апроксимація n_2, $\delta n_t = 3.413\%$

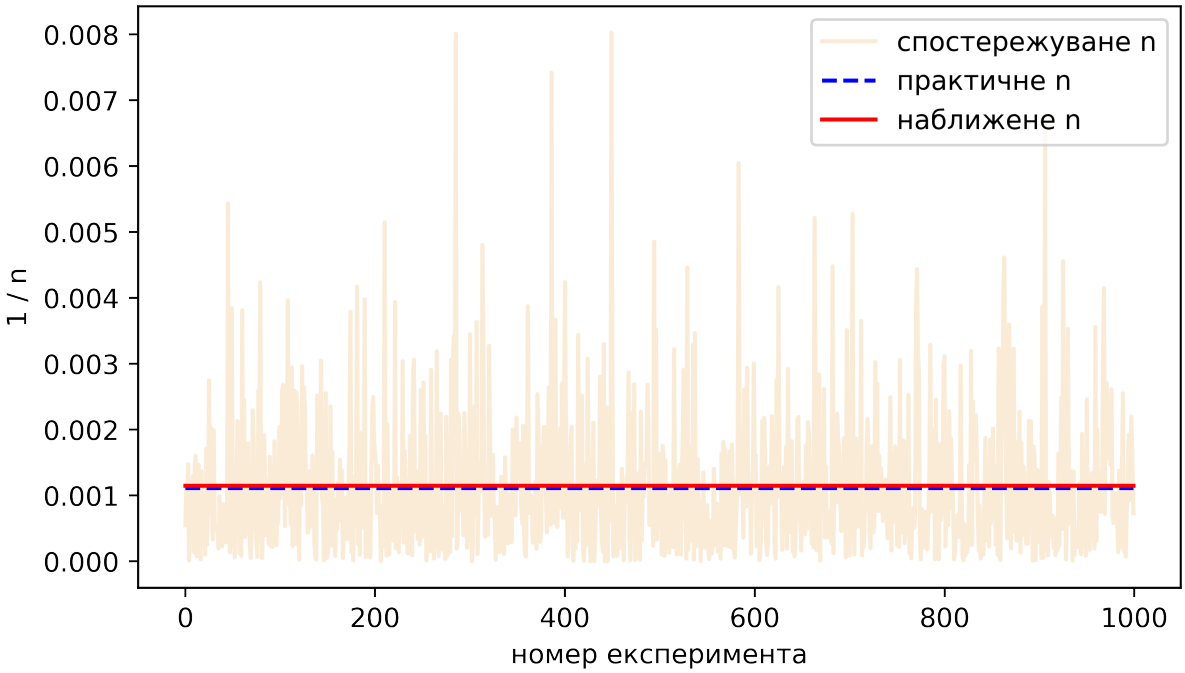

Рисунок 3.5 – Випадок в) $n_1 = 900$, апроксимація n_1, $\delta n_t = 2.478\%$

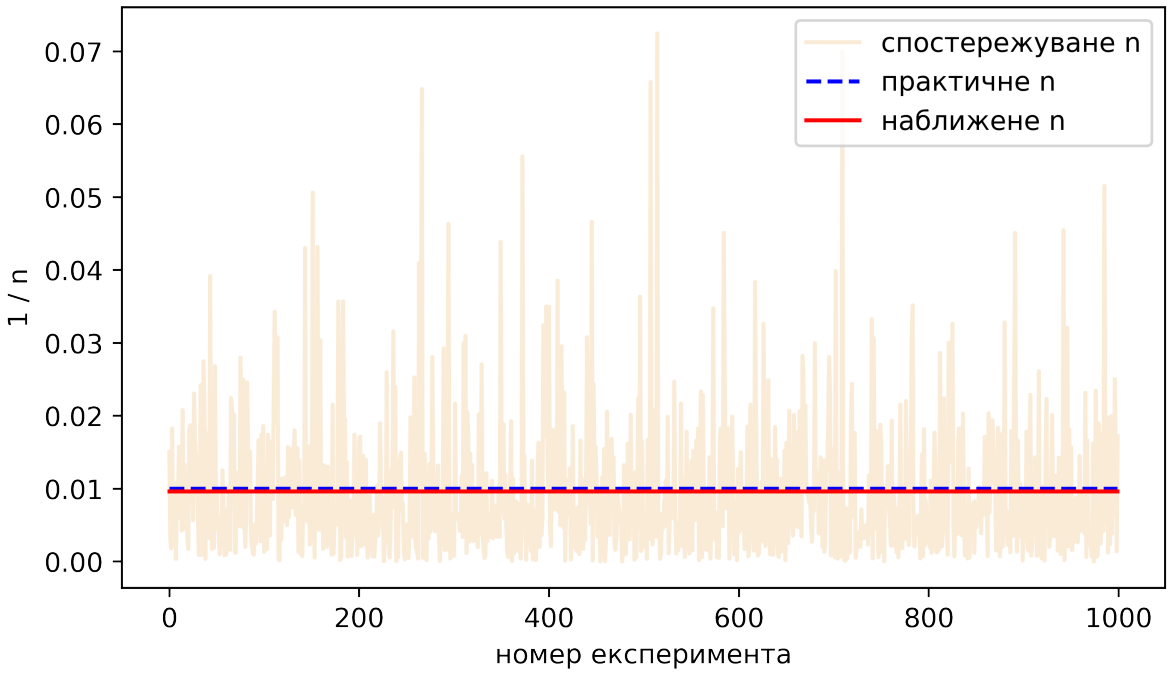

Рисунок 3.6 – Випадок в) $n_2 = 100$, апроксимація n_2, $\delta n_t = 3.721\%$

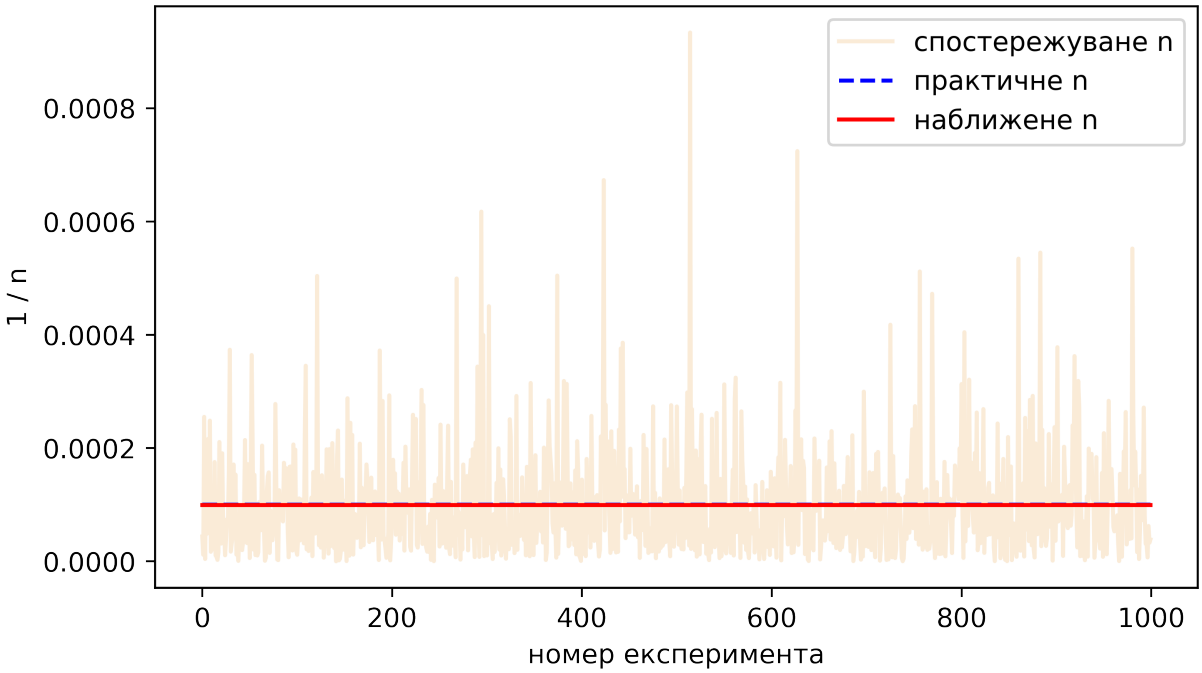

Рисунок 3.7 – Випадок г) $n_1 = 10000$, апроксимація n_1, $\delta n_t = 0.412\%$

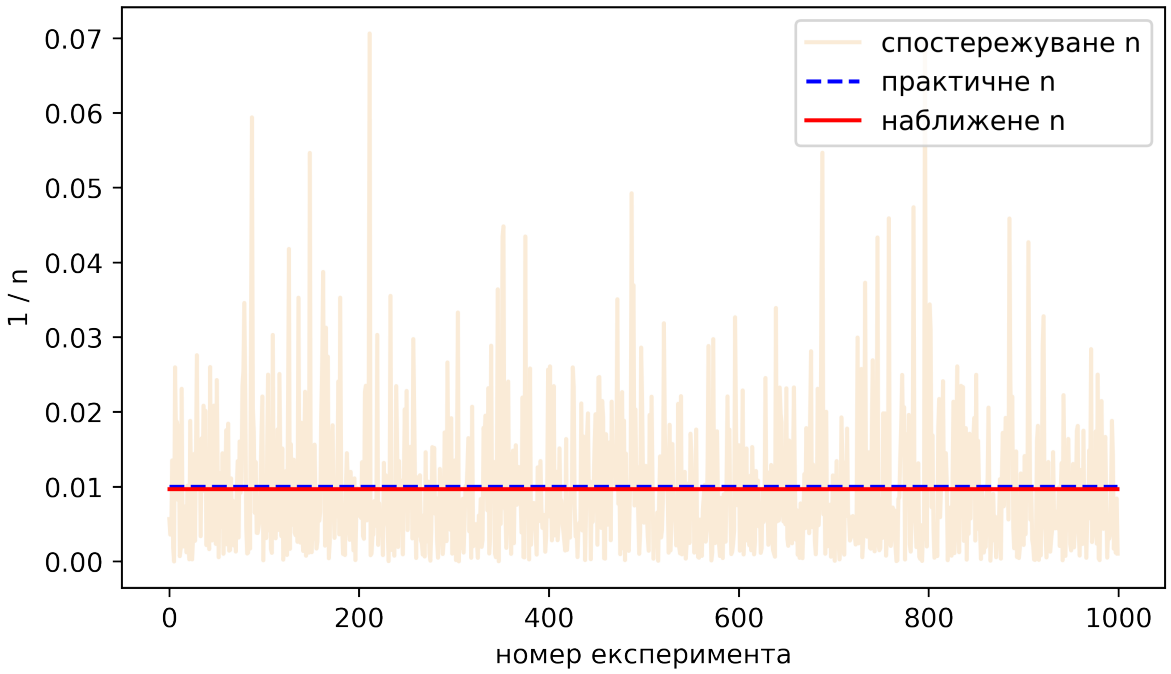

Рисунок 3.8 – Випадок г) $n_2 = 100$, апроксимація n_2, $\delta n_t = 3.387\%$

Зауваження. Доволі широкий спектр спостережуваних n пояснюється незначною кількістю операцій хешування, та, в деякому сенсі, нестабільністю хмарного процесора через неконстантну швидкість інтернет-з'єднання. При наближенні n до значень реального майнінгового обладнання спостережувані n також будуть наближатися до практичного значення.

Отримані результати наведені в таблиці для більш зручного сприйняття:

Таблиця 3.1 – Результати розрахунків

n_1	$\delta n_{1t}, \%$	n_2	$\delta n_{2t}, \%$
100	3.680	100	3.508
121	3.376	111	3.413
900	2.478	100	3.721
10000	0.412	100	3.387

Кількісний та якісний порівняльний аналіз. По-перше, при оцінці гешрейта за запропонованим методом значно скорочується необхідний для цього час. Наприклад, якщо обрати секундний інтервал — отримаємо виграш у швидкості в 3600 разів, тому у порівнянні з «класичним» методом (2.5), що потребує, в середньому, годину для оцінки гешрейта, новий метод є значно більш ефективним.

По-друге, для оцінки гешрейта за запропонованим методом не є обов'язким знаходження саме шари, тобто геш-значення, яке є меншим за таргет.

По-третє, через значне скорочення часових затрат, у разі використання нового методу, нечесні оператори майнінгових пулів не зможуть використовувати обчислювальну потужність майнера для збільшення частки винагороди всередині пула, куди майнер, на етапі встановлення його гешрейта, ще не входить.

Висновки до розділу 3

У цьому розділі запропоновано новий метод дистанційного визначення гешрейта майнера, наведено його обґрунтування та отримані практичні результати, аналізуючи які можна зробити наступні висновки:

– навіть для невеликої кількості гешувань похибка, що виникає внаслідок оцінки, відповідно до сформульованої теореми (3.1), не перевищує допустиме відхилення у 5%, що свідчить про коректність нового способу оцінки гешрейта;

– чим більша кількість гешувань — тим менша похибка, і тим точнішим є наближення, що повністю співпадає з наслідком (3.7);

– при майже однакових кількостях гешувань виникають майже однакові похибки, що також вказує на коректність запропонованого методу;

– запропонований метод є обґрунтовано більш ефективним за наявні.

ВИСНОВКИ

Під час виконання роботи, на основі огляду опублікованих джерел за тематикою дослідження, були описані базові поняття, які використовуються при побудові блокчейн-систем. Сформульовано та описано вимоги до протоколів консенсусу. Було описано види майнінгу, вказано їх переваги та недоліки. Також були детально розглянуті поняття складності та гешрейта, оцінка якого є провідною темою кваліфікаційної роботи.

Проаналізувавши наведені в літературі наявні методи за темою визначення та оцінки гешрейта, було наведено описи розглянутих методів, в тому числі формальний, подальший аналіз яких показав, що вони виявилися недостатньо ефективними.

В ході роботи було запропоновано новий метод дистанційного оцінювання гешрейта майнера, наведено його обґрунтування, за допомогою програмної реалізації були отримані практичні результати та проведено якісний порівняльний аналіз, який показав, що запропонований метод виявився більш ефективним за часом (виграш до 10000 разів), необхідним для оцінки, при цьому похибка оцінки не перевищує 5%, навіть для незначної кількості гешувань, і масштабується вниз з її збільшенням, а також новий метод є більш безпечним для майнера, так як через скорочення часу його обчислювальну потужність не зможуть використовувати шахраї.

Наведені у кваліфікаційній роботі результати можна також використовувати для визначення гешрейта неспеціалізованого обладнання, через відсутніть необхідності проводити надвелику, для такого класу обладнання, кількість розрахунків.

У подальшій роботі планується покращити оцінку гешрейта майнера, тим самим отримати меншу, незважаючи на допустимість отриманої практично, похибку для невеликої кількості гешувань.

ПЕРЕЛІК ПОСИЛАНЬ

1. Chaudhry Natalia, Yousaf Muhammad. *Consensus Algorithms in Blockchain: Comparative Analysis, Challenges and Opportunities.* 2018.

2. Nakamoto Satoshi. *Bitcoin: A Peer-to-Peer Electronic Cash System.* 2009.

3. Sadek Ferdous, Mohammad Jabed Morshed Chowdhury, Mohammad A. Hoque, Alan Colman. *Blockchain Consensus Algorithms: A Survey.* 2020.

4. Giuseppe Ateniese, Ilario Bonacina, Antonio Faonio, Nicola Galesi. *Proofs of Space: When Space is of the Essence.* 2014.

5. Iddo Bentov, Charles Lee, Alex Mizrahi, Meni Rosenfeld. *Proof of Activity: Extending Bitcoin's Proof of Work via Proof of Stake.* 2018.

6. Kostis Karantias, Aggelos Kiayias, Dionysis Zindros. *Proof-of-Burn.* 2019.

7. P. Stetsenko, G. Khalimov, Y. Kotukh. *Analysis of Attacks Surfaces on Blockchain Systems.* 2017.

8. Meni Rosenfeld. *Analysis of hashrate-based double-spending.* 2012.

9. Wang Canhui, Chu Xiaowen, Qin Yang. *Measurement and Analysis of the Bitcoin Networks: A View from Mining Pools.* 2020.

10. Lin William Cong, Zhiguo He, Jiasun Li. *Decentralized Mining in Centralized Pools.* 2018.

11. Meni Rosenfeld. *Analysis of Bitcoin Pooled Mining Reward Systems.* 2011.

12. Zhu Saide, Li Wei, Li Hong, Hu Chunqiang, Cai Zhipeng. *A survey: Reward distribution mechanisms and withholding attacks in Bitcoin pool mining.* 2018.

13. Difficulty [Електронний ресурс] / BitcoinWiki. — 2021. — Режим доступу: https://en.bitcoin.it/wiki/Difficulty.

14. D.Eastlake, T. Hansen. *US Secure Hash Algorithms (SHA and HMAC-SHA).* 2006.

15. Hashrate proof [Електронний ресурс] / Slushpool. — 2019. — Режим доступу: https://help.slushpool.com/en/support/solutions/articles/77000433900-hashrate-proof

16. Hashrate types [Електронний ресурс] / Slushpool. — 2019. — Режим доступу: https://help.slushpool.com/en/support/solutions/articles/77000422780

17. A. Pinar Ozisik, George Bissias, Brian N. Levine. *Estimation of Miner Hash Rates and Consensus on Blockchains.* 2017.

18. Catherine Roberts, Ray McLenaghan. *Continuous Mathematics in Standard Mathematical Tables and Formulae*. 1996.

19. Alsmeyer Gerold. *Chebyshev's Inequality.* 2011.

ДОДАТОК А ТЕКСТИ ПРОГРАМ

А.1 Програма симуляції результатів, отриманих у розділі 3

```
from hashlib import sha256
from random import getrandbits

import matplotlib.pyplot as plt


plt.rcParams['figure.figsize'] = (7, 4)
plt.rcParams['figure.dpi'] = 500

bit_length = 256
count = 1000


def generate_rnd_bits(bit_length):
    rand = hex(getrandbits(bit_length)).upper()[2:]
    if len(rand) != int(bit_length / 4):
        rand = '0' * (int(bit_length / 4) - len(rand)) + rand
    return rand


def get_minimal_hash(n, bit_length):
    hashes = []
    for i in range(n):
        hash = sha256(bytes.fromhex(generate_rnd_bits(bit_length)))
        hashes.append(hash.hexdigest())
    return min(hashes)
```

```
def calculate_hashing(min_hash, bit_length):
    return int('0x' + min_hash, 16) / (2**bit_length - 1)


def plot_results(n, n_calculated, n_observed, count):
    experiments = [i for i in range(count)]
    plt.plot(experiments, n_observed, label = 'спостережуване n',
    color = 'antiquewhite')
    plt.plot(experiments, [n] * count, label = 'практичне n',
    linestyle = '--', color = 'b')
    plt.plot(experiments, [n_calculated] * count,
    label = 'наближене n', color = 'r')
    plt.xlabel('номер експеримента')
    plt.ylabel('1 / n')
    plt.legend(loc = 'upper right')
    plt.savefig(f'result{getrandbits(10)}.png', dpi = 300)
    plt.show()


def run_experiment(bit_length, count, n1, n2):
    n1_observed = []
    n2_observed = []

    for i in range(count):
        M1 = get_minimal_hash(n1, bit_length)
        M2 = get_minimal_hash(n2, bit_length)

        n1_obs = calculate_hashing(M1, bit_length)
        n2_obs = calculate_hashing(M2, bit_length)
```

```
        n1_observed.append(n1_obs)
        n2_observed.append(n2_obs)

    n1_approx = sum(n1_observed) / count
    n2_approx = sum(n2_observed) / count

    error1 = abs(100 * n1 * (1 / n1 - n1_approx))
    error2 = abs(100 * n2 * (1 / n2 - n2_approx))

    plot_results(1 / n1, n1_approx, n1_observed, count)
    print(f'\nпрактичне n1: {1 / n1}')
    print(f'наближене n1: {n1_approx}')
    print(f'похибка наближення: {round(error1, 3)} %\n')

    plot_results(1 / n2, n2_approx, n2_observed, count)
    print(f'\nпрактичне n2: {1 / n2}')
    print(f'наближене n2: {n2_approx}')
    print(f'похибка наближення: {round(error2, 3)} %')


#run_experiment(bit_length, count, 100, 100)
```