Improvement of Information Propagation Delay in The Bitcoin Network

Semester Thesis submitted for master's degree in Computer Science and Engineering

Abdullah Ali Al Shahrani
College of Computers and Information Systems, Umm Al-Qura University

Supervisor
Dr. Khalid Trmissi

May 2020

Abstract

The features of Blockchain make it convenient for different applications. One of the characteristics of blockchain is transparency in which all information about transactions is shared and transferred to all nodes in the blockchain; each node knows what is going on in the database. This can confirm trust even in an environment of anonymity. Since the data is available to everyone in the system, this supports the immutability of data so no one can change or tamper with the transaction. A blockchain is a decentralized peer-to-peer system with no central authority figure. While this creates a system that is devoid of corruption from a single source, it still has some problems. Forking is one of these problems. In blockchain, forking refers to the branching of a blockchain path into two or more chains. Because of block propagation delay in blockchain, miners can find blocks at nearly the same time and that will bring a lot of risks to the blockchain network. In this thesis, a new method for block propagation will be proposed to reduce noncompulsory outgoing connections, and as a result, it will minimize the propagation delay.

نبذة مختصرة

تتميز تقنية البلوكتشين بأنها مناسبة للتطبيقات الحديثة المختلفة. واحدة من تلك الخصائص أنها مرنه وتتميز بالشفافية حيث يتم مشاركة جميع المعلومات حول المعاملات ونقلها إلى جميع المستخدمين في الشبكة؛ كل مستخدم يملك نسخة من جميع الإجراءات التي تحدث في الشبكة متزامنة مع بعض. هذا يكسب الثقة بين المستخدمين ويؤكدها حتى في بيئة عدم الكشف عن الهوية. نظرًا لأن البيانات متاحة للجميع في النظام، فإن هذا يدعم ثبات البيانات بحيث لا يمكن لأي شخص تغيير السجلات أو العبث بها. سلسلة الكتل هي نظام شبكي يستخدم النظير إلى النظير اللامركزي مع عدم وجود سلطة مركزية. في حين أن هذا يخلق نظامًا خالٍ من المشاكل الأمنية من مصدر واحد، إلا أنه لا يزال يعاني من بعض المشاكل الأخرى. الدفع بواسطة نفس العملة مرتين مثال على ذلك. وهذا يتم بسبب التأخير في وقت انتشار الاجراء القائم مما يسبب عدم تناسق في نسخ قواعد البيانات للمستخدمين ومن ثم استغلال هذه الثغرة من قبل بعض المخترقين لعمل إخلال في الشبة لمصالحهم. لذا في هذه الأطروحة، سيتم اقتراح طريقة جديدة لانتشار الاجراء لتقليل الاتصالات الصادرة غير الإلزامية، ونتيجة لذلك، ستقلل من تأخير الانتشار وتقليل مخاطر المشاكل الأمنية المذكورة.

Table of Contents

List of Figures

LIST OF TABLES

CHAPTER I

Introduction

BLOCKCHAIN

A Brief History

In the 1960s, the Internet started as a protocol that allowed computers to communicate with each other and by 1975, the US Department of Defense used cryptography for the first time in military history. They created the DES encryption algorithm for security purposes, which was monopolized and closed to others—other people were not allowed to analyze or to understand the structure of the algorithm. This was an incentive to develop another cryptography technique to be open and understood by all, and that happened in the 1980s with the development of Elliptic Curve Cryptography (ECC). In the late 1980s, an activist group of cypherpunks—people who are strongly interested in cryptography—started a cypherpunk forum where they could discuss and propose thoughts related to cryptography. Hashcash in 1997, B-money in 1998, and Bit Gold are solutions that proposed electronic cash systems. Bit Gold is a reusable form of Hashcash or proof-of-work system that is close to what we know as Blockchain today. In 2008 a user of the cypherpunk forum called Satoshi Nakamoto posted a white paper called “Bitcoin: A Peer-to-Peer Electronic Cash System”, which described the Bitcoin technology that is a combination of the previously mentioned solutions. In 2009, Satoshi released Bitcoin as open-source software for the community. Since then, developers have improved the Bitcoin network system which has become popular and has grown over time.[1]

The Concept of Blockchain

Blockchain is defined as several nodes or peers that communicate with each other in a decentralized environment with no single authority. Information is recorded in a ledger that is shared between the peers transparently, in which each node knows what is going on in the database. This can confirm trust between them even in an anonymous environment. Since the data is available to everyone in the system, this supports the immutability of data where no one can change or tamper with the transaction. [Figure1]

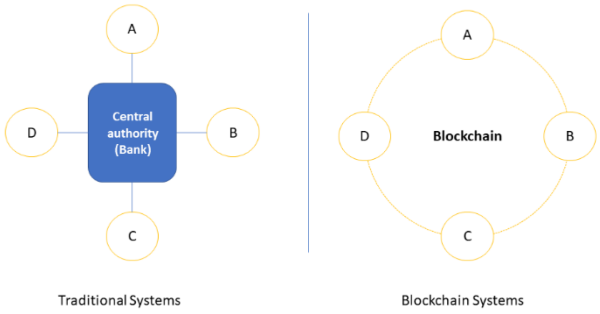


Figure 1:Blockchain Vs. Traditional systems

Blockchain has features which make it convenient for different applications. Since the information is seen by all the nodes, these transparent transactions increase trust and security between them. This has resulted in Blockchain technology being adopted for different applications. Money transfers or payment using Blockchain avoids the fees being paid to central authority banks. In addition, the Blockchain systems guarantee the charity gets to the intended recipient using the accountability of donations from benefactors. In political affairs, there are systems applying Blockchain technology to take votes and to ensure the best candidates win. Healthcare systems have started using Blockchain in patient medical record, in which the patient allows doctors and medical staffs to see his file anywhere and that can help to record a history of diseases and medical mistakes. Blockchain crowdfunding has the ability to change the industry. Because blockchain adds security to the funding process, makes it accessible from anywhere, and completely transparent, blockchain can help to maximize the success of a project in crowdfunding platforms. Cryptocurrency is the largest and most common system that is associated with the beginning of Blockchain. Since its inception, it has been popular and it has recently been traded by some governments and businesses, and people have started to transfer money using Bitcoin. The other application that uses Blockchain is the Smart contracts, in which traditional contracts are performed digitally using some computation predefined protocols that are verified by the system community without third parties.

Blockchain Architecture

Blockchain is a combination of three main components: Cryptography, a P2P network, and Game theory as shown in Figure2.

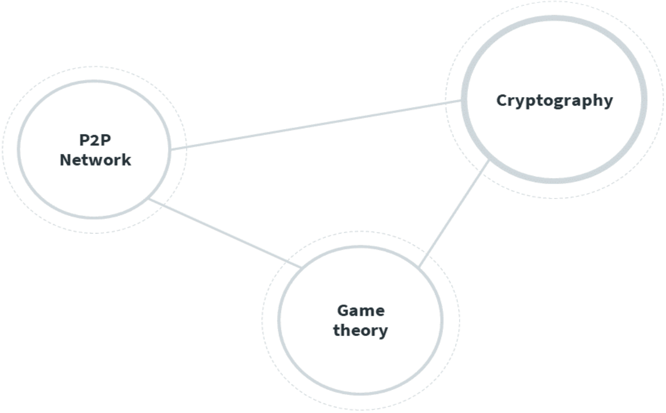


Figure 2:Components of the blockchain

Blockchain technology uses public key cryptography to ensure secure transfer transactions by performing the digital signature and hash function. Each node in the blockchain has public and private keys. The public key is used as an address for that node and other peers can use it to decrypt and verify the node transactions. Private keys are used for the digital signature where no one can tamper with or claim possession of the transaction. When a node wants to create a transaction in the blockchain, it possesses this transaction and before it propagates the transaction, it signs the transaction using the private key to ensure the identity, some encryption process happen to ensure the confidentiality, and the hash function is applied to ensure the integrity of the message.

The second component of Blockchain is the P2P network, in which the nodes are connected to each other directly without using an authority. In P2P, the parties have the same capabilities, and any party can initiate a communication session. Each node has at least 8 outgoing peers or neighbors that it can send messages. Although the nodes are connected in P2P, each node has to hold an identical copy of the data of the network or so-called distributed ledger [Fig3].

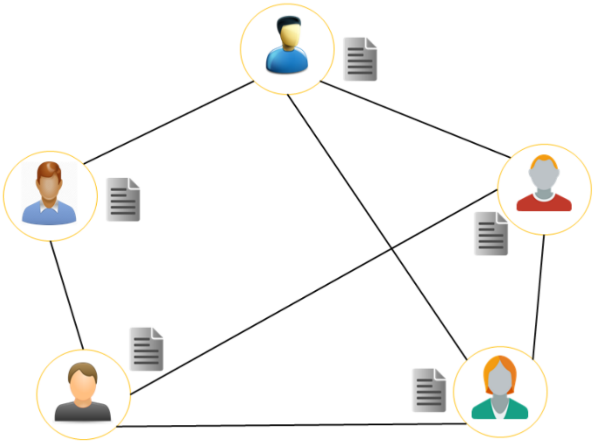


Figure 3:Nodes hold their identical copy of the ledger

The ledger contains all the information that the nodes create and transfer between them. It can be open for all and by using the cryptography hash function in a chain, no one can tamper or change the information recorded in the ledger.

The last component of the Blockchain architecture is Game theory. Game theory in Blockchain is defined as the consensus protocols that are used to make a decision by different participants and reach an agreement in a group. Since the participants are working on the P2P network, that means all have the same priority and no central authority can control the processes between them. Indeed, a consensus among those peers is required.

How Blockchain works

Blockchain has some terminologies that have to be highlighted before knowing how it works. A distributed ledger is an open database that shares information with all the nodes, which have all the data transferred between them. It consists of Blocks, and each block has a collection of data at any period of time. The Blocks are linked together using a cryptography hash function to ensure integrity and data manipulation. Any change in one block will affect all other linked blocks. Figure4 shows the structure of the blocks and how they are connected.

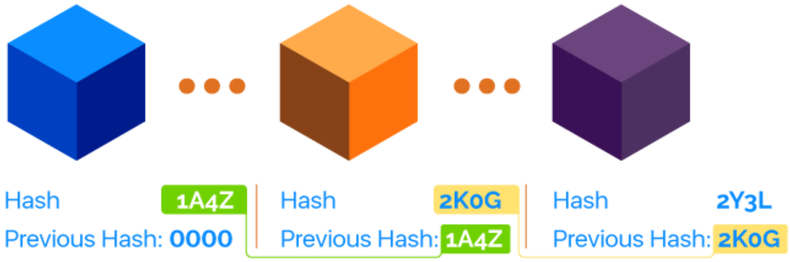


Figure 4:Structure of the blocks and how they are connected

The data is in the transactions that are created by nodes to process any intended goal, for example buying something or paying using cryptocurrency, or transferring an asset using a smart contract. Each transaction has to be broadcast to the peers, and to the transaction has to be approved and verified. The data block is structured using a Merkle tree, in which transactions represent the leaves of the tree and every two consecutive transactions are hashed together and the resulting hash is again hashed to the result of the next hash up to the root of the tree, which is considered the Block hash.

When node A wants to transfer some coins to node B, a transaction is created containing the details of this process like the amount coins that have to be paid and the sender and receiver addresses. This transaction is propagated to node A's peers using an advertisement technique. Once a peer receives a message from node A, it checks whether it has received this message before or not. If the node did not receive the message before, it will send a request message to node A to get the transaction. Node A subsequently sends the transaction to node B, and the process is repeated to the remaining nodes. For a period of time, the transactions aggregated to form a so-called Block are linked to the head of the chain using a mechanism called a consensus. The consensus mechanism is the process to reach an agreement among all the nodes. When the block is ready to be chained, the nodes compete to solve a mathematical puzzle that is related to building the block compatible with the head chain. The complexity of the puzzle depends on the network size and the number of blocks; when the size of the Blockchain increases, the complexity increases and the required time to solve and computation power as well.

Security in Blockchain

Blockchain uses Asymmetric Cryptography or so-called Public Key Cryptography to ensure security in communications between the nodes. Specifically, Blockchain uses the ECDSA algorithm which stands for Elliptic Curve Digital Signature Algorithm, that has been applied especially for digital signatures. There are four basics to consider when transferring a message on a Blockchain network: confidentiality, integrity, non-repudiation, and authentication. When the transaction is sent to a peer, it must be kept hidden from other unauthorized parties by applying encryption and decryption. The integrity of the data means that no one has altered or tamper with the data, and that is achieved by using the hash function. As mentioned, a digital signature is applied using ECDSA to ensure the non-repudiation which means the node which is responsible for sending a message cannot claim the unlike that. The authentication property is fulfilled by public and private keys. When the message is received and decrypted the receiver is able to verify that the other party is really who sent this message. Figure 5 below describes the message transfer between the nodes.

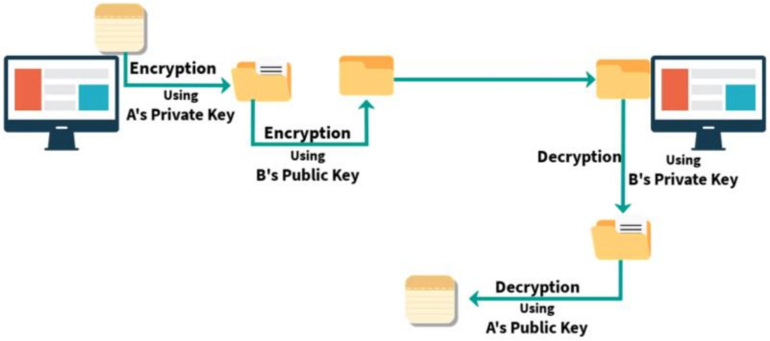


Figure 5:Transaction transfer between the nodes

CHAPTER II

Bitcoin Network

Introduction

Bitcoin is the first digital currency that depends on Blockchain technology. The Bitcoin network is a collection of nodes which follow some protocols to communicate peer to peer to transfer and store values between them. The Bitcoin software is flexible and simple to run in different areas of computing, like smartphones, which make it accessible and easy to use. Bitcoin users can buy goods and exchange money using Bitcoin.

Structure of Bitcoin

The users of Bitcoin are connected to each other on a P2P network, using pre-defined protocols. There are two types of nodes: full nodes and light nodes. Full nodes are the nodes that hold the entire ledger of bitcoin history which can verify and validate the transactions. Light nodes are the nodes that are connected to full nodes to process simple payments and can hold only the block header of the Bitcoin network.

How Bitcoin works

Transaction creation

There are some basic elements for doing a Bitcoin process. The nodes can create transactions that aggregate to one block every 10 minutes. This block must be linked to the chain by miners who can do the so-called process of mining using a Proof of Work technique. After linking the block all nodes have to verify this work and update their blockchain. In detail, transactions are comprised of inputs, outputs, and fees. Input references the previous transaction and output references the address of the new owner that the value will be sent to. An input is considered as the unspent amount from previous output minus the coinbase transaction and it must be signed by the sender of the transaction.

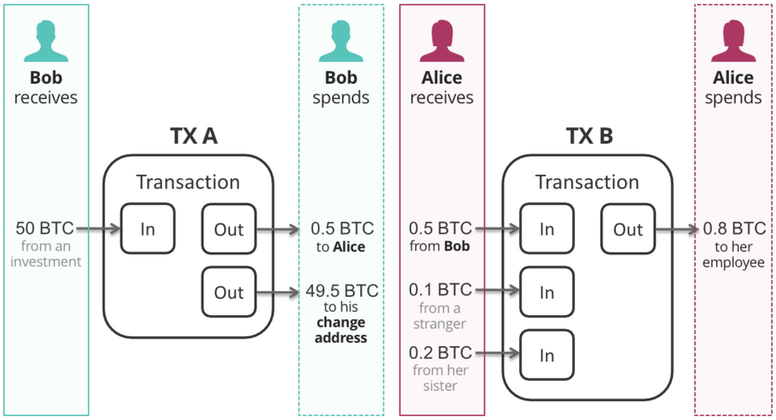


Figure 6:Transaction Inputs and Outputs Representation

It cannot be divided like real currency; when someone purchases an item costing 10 SAR and he uses 50 SAR, he expects to receive a 40 SAR as change. Similarly, the same idea is represented in Bitcoin transactions. When node A creates a transaction with a payment of 5 BTC and it uses the input of 50 BTC, it can create two outputs, one of for the new owner's address and another output for itself as a change. Outputs represent two things: the address of the new owner or change returned to the creator. Regarding the fees, it can be a small payment for the network to validate and bundle the transaction to the next block with other transactions to be mined. It can be calculated based on the size of the transaction and it gives the transaction strength and the priority to be validated and included in the next block mining. Even though transactions with fewer fees or no fees can be delayed or might be dropped from the network, miners can save the network security by rejecting invalid transactions to encourage them to pay the transaction fee.

Moreover, the transaction has to be signed by the owner using a private key and transferred to the new owner using its public key included in the transaction data. This makes the validation of the transaction easy, and fast to disseminate over the network. Because of this, an invalid transaction can no longer be forwarded.

Propagation mechanism

Once the transaction is ready to be broadcast, the node propagates the transaction over the network to its peers using a mechanism called Gossip protocol. Figure 7 shows this protocol, where the node sends an ‘inv’ message with the transaction information to the peer. Once the peer receives this message, it checks its list of known transactions, if it is already known, it replies with ‘getdata’ message and the node will respond by sending the transaction.

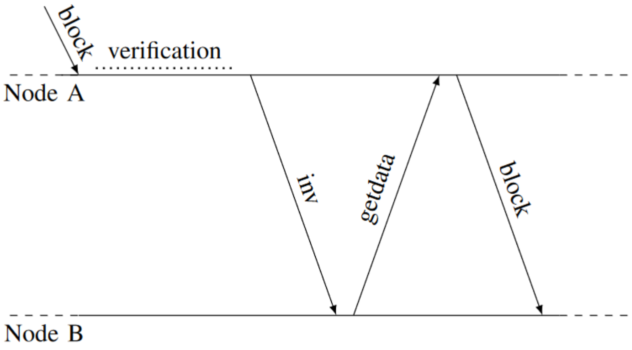


Figure 7:Exchange the message between the nodes in Bitcoin network

With this technique, the valid transaction will immediately be disseminated if it has not been seen before over the peer to peer network.

Transaction Confirmation

Once the valid transaction is verified, it is included in a ‘mempool’ and broadcasted to the peers. A ‘mempool’ or memory pool is a system of memory in which each node maintains all the transactions that have not yet been confirmed. Regarding some rules like the amount of fees, miners pick the transaction to be included with other transactions in the block they will mine. When the block is approved and linked to the head of the chain, that means the transaction has been accepted by the bitcoin nodes. In that manner, a transaction has one confirmation and when the next block is approved and linked to the chain, it is considered as two confirmations and so on. When a transaction has more than six confirmations, it can be impossible to revoke because a massive amount of computing time is required to recalculate six blocks.

Mining the block

Bitcoin used a Proof of Work algorithm to mine the next header block on the chain. It is the mechanism to reach a consensus among all the Bitcoin nodes. Once transactions are aggregated in the 'mempool' for a specific amount of time (every 10 minutes), miners pick up transactions based on some rules to be included into the next block. The miners compete to find a numeric value using SHA256 that meet a prerequisite network target. Hashing the block must give a fixed value which fulfills the network target, thereby a nonce 32-bit value is added to the block incrementally in each hash process until they find the network target. This target changes its difficulty dynamically every 2016 blocks to control the computation power used for producing PoW. The first miner who gets to the right target value broadcasts it to the network and he gets a block reward and keeps all transaction fees included in that block.

Block verification

After the block is mined and disseminated to all bitcoin nodes, they verify the block using the same hashing process by checking whether the solution matches the network target or not. This process is easy to do but hard to find.

To understand Bitcoin and how it works, the following glossary table describes the most common terminologies:

Table 1: Most Common Terms in Bitcoin

#	TERM	DESCRIPTION
1	Digital currency	Unlike fiat currency, it's fully digital and does not have a physical existence
2	Decentralized	No central authority, and the nodes use P2P network communications
3	Address	A Bitcoin address is a public key which is known by all peers
4	Block	Group of transactions that have a unique id number as a hash number
5	Hash	Hash is the fingerprint representing the identity of the block performed by a hash algorithm like SHA 256
6	ECDSA	Cryptographic algorithm used in bitcoin for digital signatures
7	Merkle tree	Structure of the transactions in each block
8	Merkle root	Represents the hash number of the block
9	Genesis block	The first block in the blockchain is the genesis block
10	Coinbase transaction	It is the first transaction in the block created by the miner
11	Miner	A node that can find the validity of the next block in the blockchain
12	Full node	A node containing all the data of all the users' transactions from the beginning to the current block
13	Consensus	The consensus mechanism is the process to reach an agreement among all the nodes
14	PoW	Proof of the Work process done by miners to proof their work finding the next block
15	Nonce	It is a changeable number included in the Block, it sets the hashing process to reach some target value
16	Network target	The block hash value starting with zeros
17	Difficulty	Determines how much computation power is required for producing a PoW
18	Reward	An amount rewarded for the node that proofs its work for finding the next block
19	Fees	A small payment for the network to process and validate the transaction
20	Mempool	A system memory that contains all transactions which are verified by a node and still not confirmed
21	Confirmation	Acceptance of transaction from the network once it is recorded in a block, and that means one confirmation
22	Wallet	Software contains the public and private keys for the bitcoin node able to send, receive and store the Bitcoin

Technical Challenges and Security issues

Despite the Bitcoin being devoid of corruption from a single failure point and solving some obstacles in other traditional payment systems like Visa, PayPal, Bitcoin still suffers from some technical challenges and security issues that need to be researched and addressed. In the following section, some of that will be shown and discussed:

Technical and ethical issues:

- **Throughput**

  While Bitcoin processes one transaction per 10 minutes which need to be disseminated and verified, VISA treats 2000 transactions per second, meaning Bitcoin has less productivity than VISA.

- **Latency/performance (10 Mins to process)**

  As stated previously, Bitcoin needs to be propagated and verified and processed, so to make a chain into the block must take time at least 10 minutes to finish one transaction. This latency makes the performance of the Bitcoin less than others.

- **Size and Scalability**

  As time proceeds, the Bitcoin network is growing quickly and the scalability is also getting bigger, thereby transactions broadcasting and processing take a lot of time.

- **Wasted Resources (mining)**

  When miners compete to solve the mathematical puzzle to produce the next block on the chain, they use special computers and tools like GPUs and different servers to find a suitable solution which requires high energy consumption and wastes resources.

- **Versioning**

  Upgrading versions of the Bitcoin network is very hard to apply. Because the nodes have the same features and no one can get authority over others, updating the version is a decision that must be unanimous, or at least 51% of the nodes must agree to the change.

- **Criminal activities**

  One of the reasons behind Bitcoin is to be hidden from governments and prosecution, and this facilitates criminality and suspicious activities that can be done easily [3].

Security risks:

- **Majority attack**

  A majority attack occurs when some adversary has control of 51% of the network, in which he can pass his malicious activities over the network forcing the nodes to approve his work thinking that they are doing the right thing.

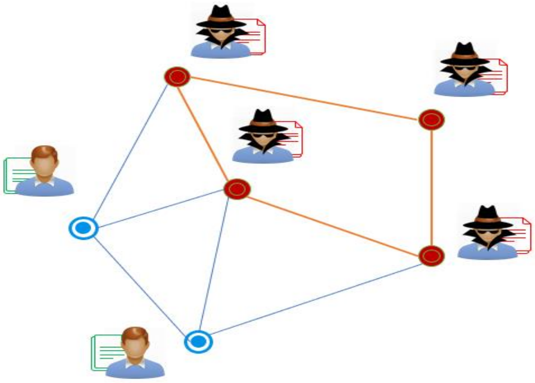


Figure 8:51% attack where the most of nodes are adversary

- **Inconsistency**

  When the message has become slow to disseminate, the synchronization of the ledger will be a challenge. This creates many potential risks like double spending, partitions, and eclipse attacks.

- **Spending coin twice**

  A double spending attack is conflicting transactions that attempt to spend the same coin in order to defraud a third node. Because of the inconsistency in the replicas, the double spending has the opportunity to occur and abuse the public ledger. The adversary may spend the same coin twice.

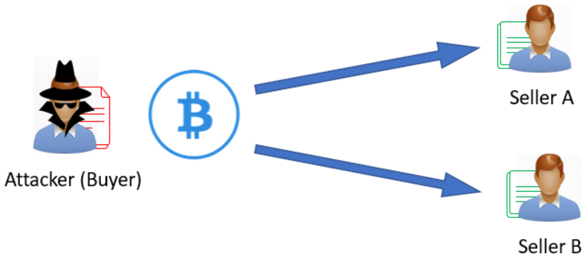


Figure 9:Describe how double spending attack occur

- **Forking**

One of the biggest issues of Bitcoin blockchain is the vulnerability of forks; fork refers to the splitting of blockchain into two paths growing forward. Figure 8 shows the forking in a blockchain. It is due to two blocks being mined and found by two miners nearly at the same time. When the next block is generated, it will chain to one of the two stated paths, and that is unlikely to be simultaneous. The chain which will be longer is considered as the authentic one and the short one will be called stale blocks or orphan blocks. The stale blocks bring risks to the blockchain which will be as a stage for an adversary to do his malicious activities with the aim of earning more rewards or to get benefits of nodes computational power to support his bad behavior.

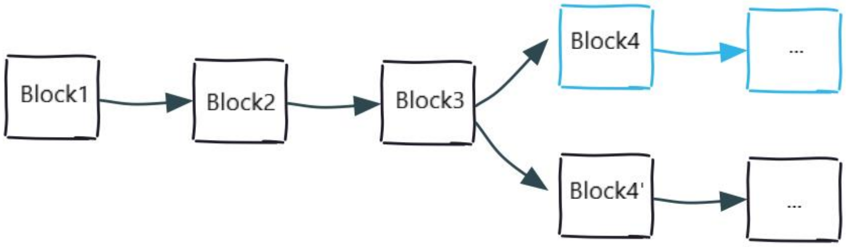


Figure 10:Forking in Bitcoin Blockchain

- **Withholding attack**

Called an eclipse attack, the attacker can isolate the victim from in- or out-going connections in order to perform some nefarious purposes.

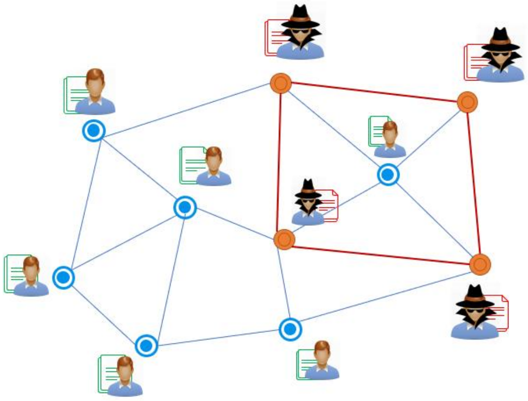


Figure 11:Eclipse attack

PROBLEM STATEMENT

Since Bitcoin's inception, researchers have studied most issues and proposed solutions to the adversarial strategies and security vulnerabilities. Since its responsible for most of the security issues in an anonymous and decentralized network such as Bitcoin, it has been gaining more attention in the field of Blockchain research. They found that the delay of information propagation, which is a combination of transmission time and verification time, is responsible for risks and security attacks on the Bitcoin network. From double spending attacks to inconsistencies in the replicas of ledgers and other attacks like partition attacks and eclipse attacks, which occur as a result of propagation delay.

CONTRIBUTION

State of the art research on information propagation delay and its impacts on the Bitcoin network will be introduced in this thesis and how to countermeasure it in different ways. Based on our studies, we have classified the enhancement propagation delay solutions into four categories:

1. Change consensus protocol
2. Minimize verification time
3. Propagation protocol
4. Network topology

After reviewing the previous works, a new method for information propagation is proposed to reduce noncompulsory outgoing connections, and as a result, it will minimize the propagation delay.

ORGANIZATION

This thesis is organized as follows: Chapter I is an introduction to Blockchain. Chapter II reviews the relevant literature and analyzes the proposed methods and mechanisms. Chapter III will introduce the proposed method and analyze the results.

CHAPTER III

Literature Review

OVERVIEW OF PROPAGATION MECHANISMS

The propagation mechanism describes how the transaction or block spreads through the network to all the nodes, in which all the information is expected to be received completely. Unfortunately, that does not always happen due to the impacts of network scalability and security issues like partitioning of the network. The propagation mechanisms used in Blockchain is described below [7], [8]:

Advertisement-based information dissemination:

This protocol is known as *Gossip-like* protocol, which used in the Bitcoin network. When node A receives a message, it announces it to its peer using *inv-message*. Node B responds to that message by using the *getdata-message* if it has not already received it. Otherwise, no action will be taken.

Send headers:

This is an updated form of the previous one, in which peers can send a block header directly without sending an *inv-message* to reduce the latency and decreases the bandwidth overhead.

Unsolicited block push:

When the miner mines a block, there is no need for the block to be advertised because it is not yet known by the other nodes. This reduces the overhead bandwidth and time latency.

Relay networks:

This mechanism allows miners to share a mining pool using transaction IDs. Since it has less size than a transaction, the transaction is replaced by IDs in the block when broadcasting to minimize the delay.

Hybrid push/advertisement systems:

In this mechanism, when the node A has n peers, it will announce the block to the $\sqrt{n}$ and push the block to $n - \sqrt{n}$. This protocol is used in Ethereum.

ANALYTICAL STUDIES ON PROPAGATION DELAY

According to the decentralization of the Bitcoin network, the information of either transactions or blocks has to reach a consensus for verification and validation. There are several common factors that have a direct impact on the propagation and cause inconsistencies in the replica. We will describe them in the following points:

Negative Factors on Propagation Delay:

Analysis of information propagation in the Bitcoin network is presented by [9],[10],[11],[12] and [15], in which they concentrated on the following factors:

Network scalability:

While the Bitcoin network relies on participating nodes with no central authority, a transaction has to be transmitted through all of them. When the number of nodes is scalable, the transmission speed will be slower.

Bandwidth overhead:

When the number of exchanged messages increases, the bandwidth overhead also increases, thereby delaying the propagation.

Block size:

Since the inception of the Bitcoin, the number of transactions was limited and still increasing gradually. Up to now the average number of transactions is 2000 t per block (source: Blockchain.com), when the number of transactions increases, the size of the block becomes larger, and that will affect the speed of the block propagation over the network.

Link latency:

When the node creates a transaction and broadcasts it to the peers, the transmission time processing is the link latency between the origin node and its peer. When the origin node is far away from the peer the link latency will be lower.

Client behavior:

Node session length refers to client behavior, and how long the client has been connected to the network. When the node is connected for a while and then disconnected that will affect the links between the peers and thereby the network topology which is in charge of effecting the propagation processing.

Network topology:

The nodes connect randomly to each other based on network protocols, in which every node maintains a list of DNS servers returning IP addresses of peers. Such a random connection provides non-compulsory hops that will affect the propagation by wasting time to disseminate them.

Threats and obstacles due to propagation delay:

As we mentioned, the propagation delay causes some security issues and affects performance. Below, we describe them briefly.

Replica inconsistency:

When the message has become slow while disseminating, the synchronizing of the ledger will be a challenge. This, therefore, creates many potential risks like double spending, partitions, and eclipse attacks.

Double spending attack:

From an information propagation perspective, the papers in [11],[14] studied and analyzed this problem. Because of the inconsistency in the replicas, double spending has the opportunity to occur and abuse the public ledger. The adversary may spend the same coin twice. When the attacker creates two transactions (*ta, tm)* with the same input and different recipients, *ta* will be sent to the majority and *tm* will be sent to the merchant. If the *ta was* accepted by the majority, then *tm* will not be valid and will be rejected by them. In this situation, we consider double spending a successful attack. The proportion of double spending will potentially be higher when the speed of transaction propagation is slower [13].

Partition attack:

Blockchain forks are addressed by [9] and [10]. In this case, we can say partition occurs when there is more than one head in the Blockchain, and the nodes do not agree on which the block is the head of the chain. In time the longer chain will become adopted as the main chain and the shorter one will be removed by the nodes. That forms so-called stale blocks or orphan blocks, in which those blocks increase the advantage of double spending and eclipse attacks, and most adversaries exploit that blocks to do their malicious activities. The propagation delay pertained to the occurrence of partitions on the network by delaying the ledger synchronization.

COUNTERMEASURES

In this section, we are going to introduce most of the countermeasures that are proposed to improve propagation delay in the Bitcoin network.

Karame, et al. [16], proposed a set of countermeasures that enables the detection of the double spending attack on Bitcoin's fast payments. The first method is by waiting for a period of time after receiving a transaction from a node and before sending a service to him to check whether there is a conflict with another transaction with the same input. Another solution to detect double spending is to set an observer as a node that directly relays all transactions to the vendor make them aware of double spending. The third solution is to adopt Bitcoin peers to create alerts about conflict transactions.

Bamert, et al. [18] minimized the chance of the double spending problem in fast payment scenarios by proposing some strategies in which they claim to improve the payment processing time. They suggested that the merchant should not accept a direct transaction from the sender itself. Additionally, the merchant has to be connected to as many random nodes as possible to avoid any possibility of fault transaction injection.

Decker, et al. [9] proposed three methods to speed up and improve the propagation information in the network. The time it takes to verify the block is a major contributor to the propagation delay, and there is a correlation between block size and time to verify it. The first method proposed here is to minimize verification by dividing the process into two phases: an *initial difficulty check* which consists of validating the proof-of-work, and a *transaction validation* that checks the validity of each transaction. In this case, the block is relayed to its neighbors, as soon as the difficulty is checked but before the transactions' verification, instead of waiting for longer validation of the transactions to be finished. One thing to be considered is that relaying information that has not been validated might allow an attacker to send unpredictable data arbitrarily that is then relaying over the nodes and resulting in a distributed denial of the service attack.

The second method is pipelining block propagation. Fig. 12 shows this technique, which can be done immediately by forwarding ‘inv’ messages to neighbor nodes utilizing the round-trip times between nodes and its neighbors by announcing block availability before getting it. One of the limitations is the advisory may announce a number of fake blocks that he cannot provide when asked for them.

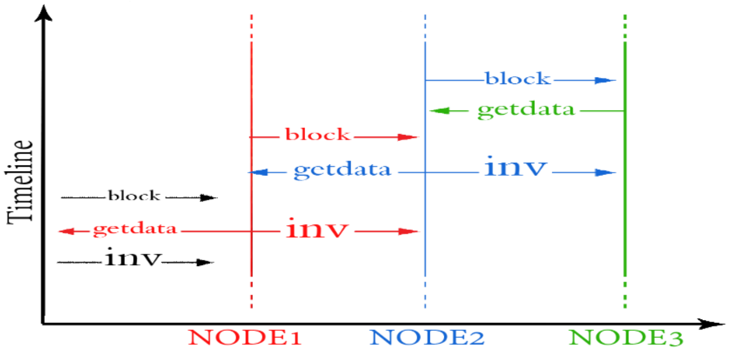


Figure 12:Message exchange using pipelining technique

The third method works by shortening the distance between the nodes by using a star-sub graph network, the hub between every two nodes becomes near to two hubs. While shortening the distance can work efficiently on a small network, in a larger network it can cost a vast amount of bandwidth.

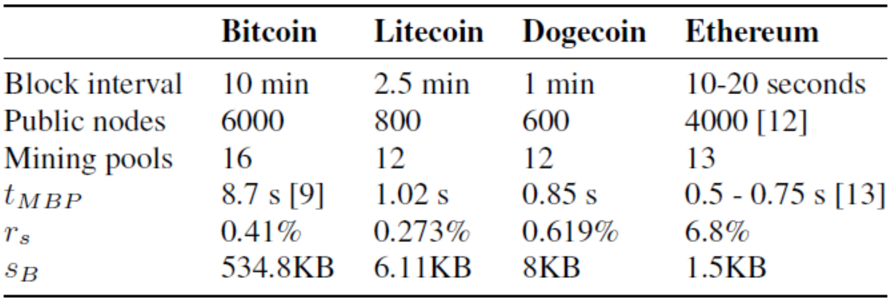
Analysis of the feasibility of a partitioning attack on the Bitcoin network is presented by Neudecker, et al. [10]. They proposed a simulation model that studies a feasible attack on the Bitcoin network topology[1]. The model was parameterized based on some measurements like peer’s session length and link latencies between them which are performed using the *bitnodes.io* project that provides a crawler for reachable nodes on the Bitcoin network[2]. As a result, validating the model showed that

[1] https://github.com/bitcoin/bitcoin
[2] https://bitnodes.earn.com/

correspondence with information propagation observed on the real Bitcoin network. The analysis revealed that having 6000 peers in control reduced the chance of attackers exploiting the partitions on the network.

Gervais, et al. [11] studied and devised various optimal strategies for double spending and selfish mining PoW blockchain. They presented a novel quantitative model that analyzes different implications of PoW blockchain[3]. They simulated a model of PoW-Blockchain and network layer, which mimics aspects of a real-world network and Blockchain parameters, and is modeled on the Markove Decision Process (MDP). They presented crawler nodes for different PoW Blockchain based instances, which in turn measured the stale blocks rate (Table 2), that thereby fed their model as input to quantify the optimal attacker strategies for double spending and selfish mining. The result showed the impacts of network parameters on the security of PoW Blockchain for stale blocks on double spending and selfish mining.

Table 2:Comparison of different Bitcoin forks



	Bitcoin	**Litecoin**	**Dogecoin**	**Ethereum**
Block interval	10 min	2.5 min	1 min	10-20 seconds
Public nodes	6000	800	600	4000 [12]
Mining pools	16	12	12	13
t_{MBP}	8.7 s [9]	1.02 s	0.85 s	0.5 - 0.75 s [13]
r_s	0.41%	0.273%	0.619%	6.8%
s_B	534.8KB	6.11KB	8KB	1.5KB

Bitcoin network measurement was presented by [12] for simulating and validating transaction propagation. They discussed the effect of delay on security due to inconsistencies in the replicas that leads to opportunities for double spending and then abuse of the public ledger. They run a real Bitcoin client that works as a crawler for learning the number of reachable connected nodes and their session lengths precisely. In addition, they implemented a measuring node that has the same behavior of the real Bitcoin node, such as a node connected to peers and can create and propagate transactions. The measuring node was able to track the

[3] https://github.com/arthurgervais/Bitcoin-Simulator

dissemination of that transaction over the network, and thereby calculating the propagation delay differences between sending time and the received time by each node. Figure 13 shows the distribution of propagation in a real network compared to the simulation.

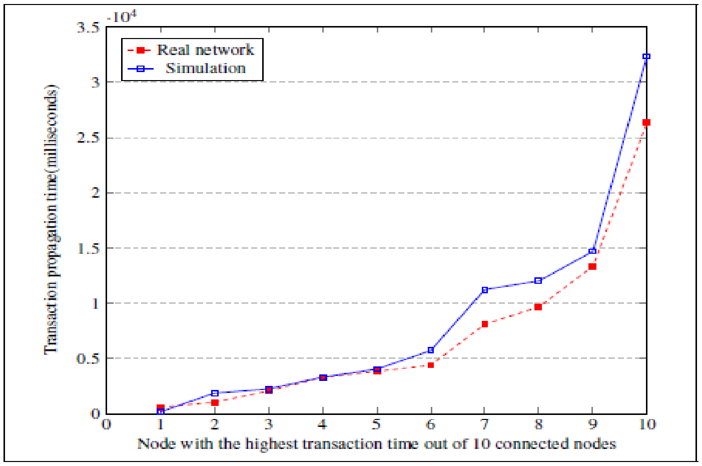


Figure 13:Distribution of propagation in real network compared to the simulation

The result revealed that increasing the number of nodes has a direct impact on propagation delay and not all the nodes, except rare cases, receive the transaction during dissemination.

Three different Bitcoin models are presented by Fadel, et al. [14] to enhance the propagation delay on the Bitcoin network. The first method is called Bitcoin Clustering Based Super Node (BCBSN) that is parametrized based on the real Bitcoin network, which they measured by creating a Bitcoin client for crawling the network and gathering a required data i.e. the number of reachable nodes, peers' session length, and link latencies between them.

The main idea of the model is to reduce the non-compulsory hops and thereby enhance the propagation delay by building a Bitcoin network using clustering peers in which each cluster is maintained by a node called a super node which is known by other super nodes and other nodes connected to super nodes, both super nodes and other nodes are based on some features like higher weighted, node reputation and geographical algorithms.

The result displayed in fig.14 notes the decrease in propagation delay compared with the existing Bitcoin protocol with a high proportion.

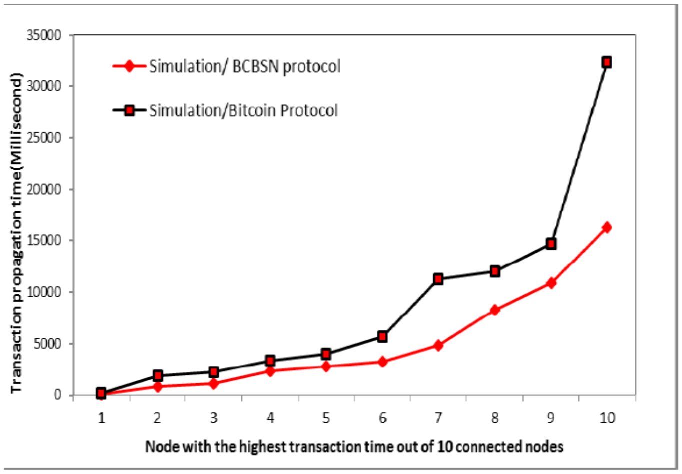


Figure 14:Comparison of the distribution of Δtcn as measured in the simulated Bitcoin protocol with BCBSN protocol simulation results

The second presented protocol is a Locality-Based Clustering (LBC) that forms peer connections with the aim of reducing the non-compulsory hops and improving propagation delay. Based on a threshold distance, the node measures the distance to the discovered node and sends a JOIN request to it. Once it receives a connection to it, it learns the IPs of the nodes that belongs to the same cluster. By evaluating this method, the result showed a decrease in the propagation delay compared to real protocol and previous methods (see fig.15).

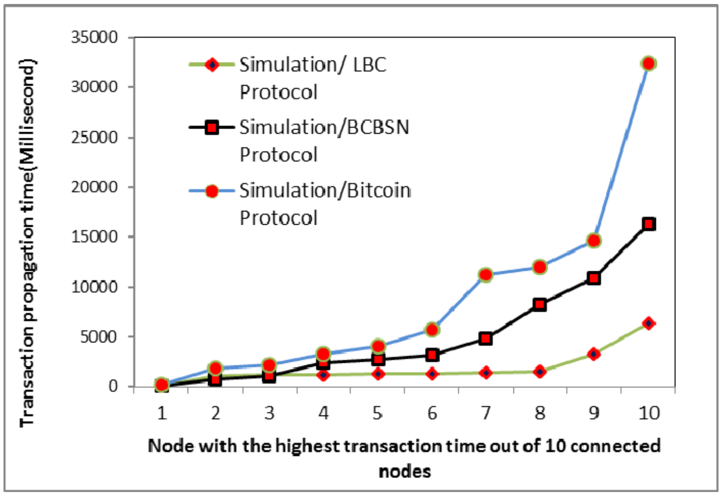


Figure 15 :Propagation delay distribution as measured in real Bitcoin, BCBSN and LBC

The third method is a proximity-based clustering approach (BCBPT) using time latency to structuring peer nodes of the Bitcoin network. The key reason behind this is to decrease the links between nodes and therefore latencies between them. The results of all the three proposed methods are displayed in fig.16 which demonstrates the decreases in the propagation delay distribution.

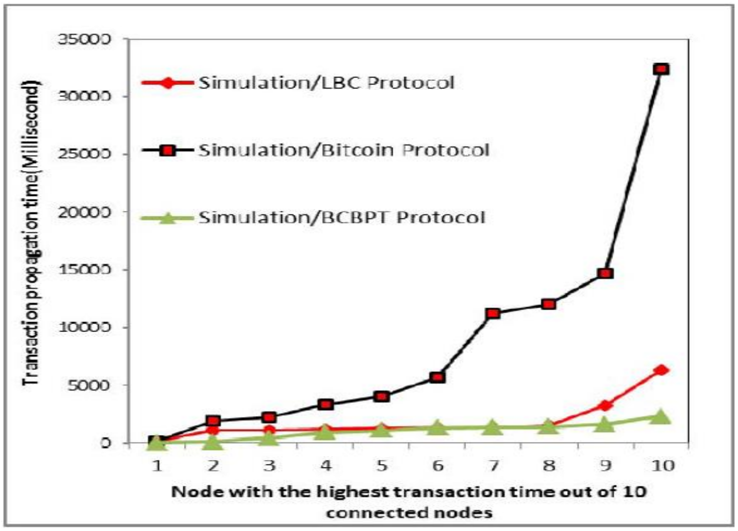


Figure 16:Propagation delay distribution as measured in real Bitcoin, LBC and BCBPT

However, this method is susceptible to adversarial activities like partitioning and eclipse attacks which reduces the randomness for peer selection and thereby decreases the security of the network. As the node's behavior is unstable, all the proposed methods will suffer from clients joining and disjoining while looking for an optimal peer every time.

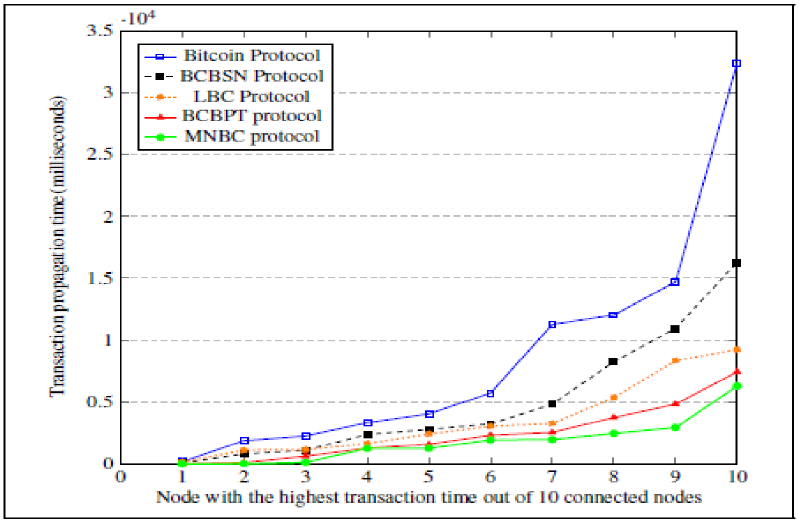


Figure 17:Comparison of different proposed method with the real protocol

Furthermore, the researchers above conducted an improvement of their previous protocol (BCBSN) called Master Node Based Clustering (MNBC), in which the master nodes are fully connected based on proximity and information can be transferred between master nodes as well as the normal nodes. The main idea for this is to reduce the occurrence of partition attacks. Fig.17 shows the result of evaluating different presented methods.

Stathakopoulou, et al. [19] conducted a dissertation that tries to address the problem of consensus on transaction history by minimizing propagation delay. By using pipelining messages, i.e. ‘inv’ messages sent directly to peers as soon as it arrives and while the node is waiting to get the data without verifying that message, to be spread rapidly over the network. In addition, they tried to increase the connectivity of the geographically closest nodes to speed up information propagation. Implementing the method showed that when increasing connectivity to the closest peers, the average time of requesting data of ‘inv’ message decreased from 0.86 to 1.14. And when a pipelining mechanism was applied to the broadcasting transactions, the average time a transaction has to be propagated decreased to 0.2943 seconds whereas without pipelining it was 0.7474 seconds and by combining the two proposed solutions, the average percentage of announced transactions was 71%. Despite the effectiveness of that solution, their suggestions signify compromises on security, which means the adversary will flood the network with fake transactions. Additionally, connecting to the closest peers using the selection method is vulnerable to an eclipse attack.

Analysis of the Bitcoin network and observation of the transaction and block dissemination is presented by Pappalardo, et al. [20]. They used a Bitcoin client that can establish connections with peers and to able to monitor the network activities for a period of time, to identify the appearance of transactions in the Blockchain network. In addition, they measured the propagation mechanism and the time of including the transaction or block on the blockchain. By observing the network, the results reveal that 42% of the low-value transactions were not included in the ledger of the chain one hour from their appearance and 20% were not included after 1 month. This was not because of block size but because of the low fees that did not induce the miners to mine those transactions.

Fig.18 shows the comparison of observed transactions in a period of time and the transactions included in blocks at the same time.

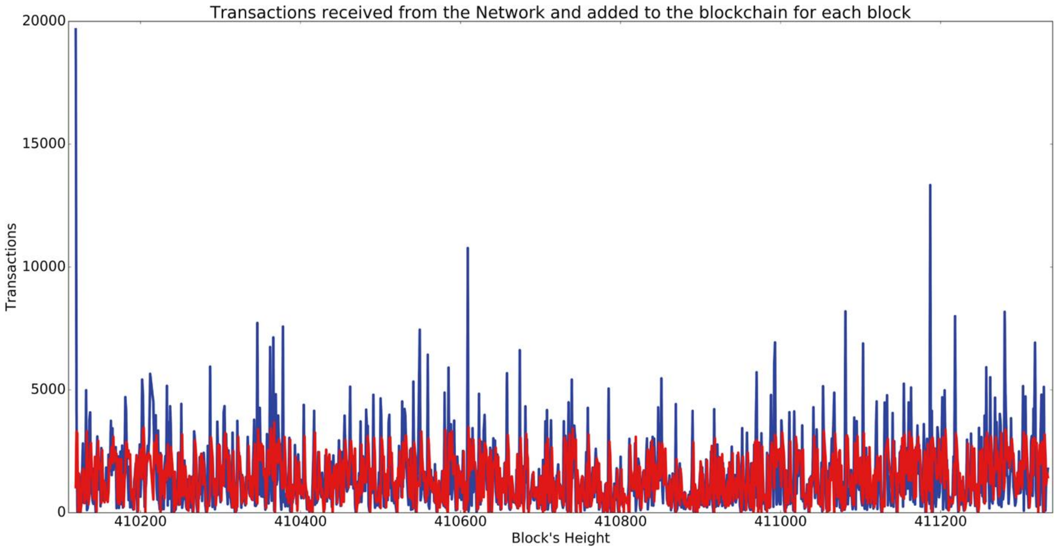


Figure 18:Comparison of observed transactions in a period of time and the transactions included in blocks in the same time

They recommended a level of transaction fees as a countermeasure to incentivize miners to process in a timely manner and guarantee their inclusion on the blockchain.

Marçal in [23] studied the problem of minimizing exchange messages between nodes which saves bandwidth without affecting the current approach. The main goal is to decrease the number of duplicated advertisements over the network and ensure that the transactions get to the miners. Some algorithms are applied to predict miners or peer nodes connected to miners based on some priorities. Every node maintains a list of transactions sent by the peers and the time taken to be bunched in a block. By implementing the method and analyzing the result, they saw a bandwidth reduction by 10.2% and the number of exchanged messages reduced by 41.5%. However, they implemented the method in a stable network. If we take client behavior into consideration, since it changes every time because of the session's length, this method might be useless.

Sudhan, et al. [17] studied the ledger inconsistencies caused by transaction propagation delay in the network, which thereby help to double spending twice. They proposed a peer selection technique to find the best combination of the number of outgoing connections either randomly or based on proximity to reduce propagation delay. This method has two aspects: changing the number of outgoing connections and selection technique based on both proximity and randomness.

Table 3:Propagation delay varying proximity (Np) and randomly (Nr) order of time and based on threshold distance (DT)

NP = Nodes selected based on threshold (proximity)	NR = Nodes selected randomly outside proximity parameter	Threshold Distance (DT)
6	2	1500
6	2	3000
4	4	1500
2	6	1500
4	4	3000
6	2	5000
Random Selection. (default in Bitcoin protocol)		
4	4	5000
2	6	3000
2	6	5000

The evaluation of the results shown in Table 3 reveals that the optimal number of outgoing connections is outgoing connections based on proximity and 2 outgoing connections randomly selected at the distance of 1500. By applying the peer selection algorithm, the propagation delay decreased when the outgoing connection has a high number of connections. However, this proposed method has the potential for eclipse attacks.

Bitcoin NG is a new Blockchain protocol proposed by Eyal, et al. [21] to tackle the problem of scalability, which is one of the issues that causes propagation delay in the Bitcoin network. It decouples the block into two types: one for electing a leader called *block key* and the other for recording the transactions called a *microblock*. Miners are competing to become a leader, in which the winner will be responsible for serializing the transactions until a new leader appears. Time is divided into epochs, in which each epoch has a single leader. By applying this method, the leaders will be in charge of recording transactions and generating the block in that epoch, and other nodes responsible for exchanging the messages between peers.

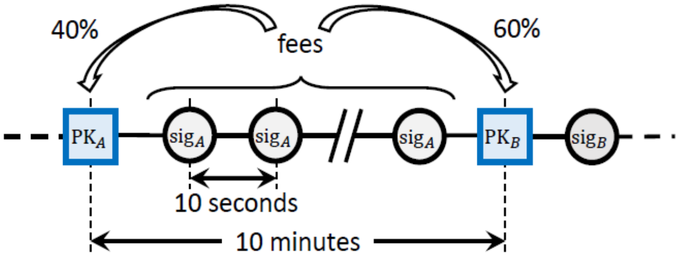


Figure 19:Design of the Bitcoin-NG protocol. Microblocks are denoted by circles and Key blocks by squres. in which microblocks signed by the current leder and the fees shared between the current leder (40%) and the next one(60%)

Fig.19 shows the structure of this protocol, that accelerates transaction confirmation and improves the latency. However, this method is vulnerable to selfish mining attacks, and to introducing a tradeoff between security and a Bitcoin network where the leaders pose a threat to the community and the protocol, in which they control most of the processes in the network.

Bi, et al. [22] proposed a method called Closest Neighbor Selecting (CNS) for selecting closest peers based on Round Trip Time (RTT). RTT is used to measure the distance between connected peers. The smaller the distance the closer the node. They claim the method accelerates the propagation process and gives a better performance, as seen in Fig.20.

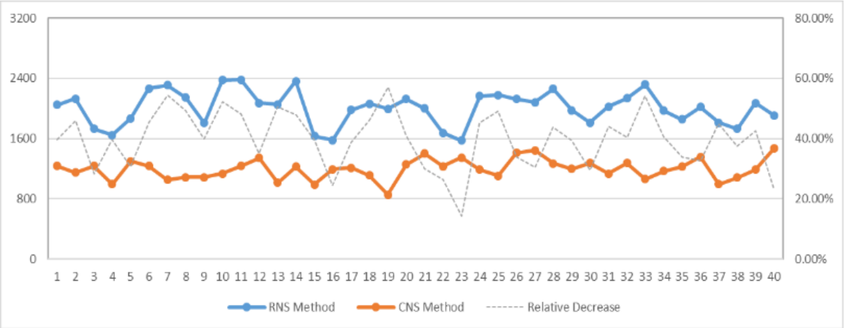


Figure 20:Comparison of Random Neighbor Selection (RNS) and CNS with average latency

However, the method has some limitations. They implemented the method on the small number of nodes (max experiment nodes were 40), which did not produce an accurate result when the number of nodes increases. Furthermore, the ability to select the peers increases the advantages of eclipse attacks and decreases the randomness thereby the security of the Blockchain network.

Compact block was presented by Corallo [23], [24] using the idea of a Bloom filter to reduce the bandwidth overhead of a new block propagation to full nodes. Rather than sending a whole block, the node sent a sketch to the peer which contains only a block header, transaction IDs and full transactions that were not expected received by the peer before. Once the peer receives that sketch, it tries to reconstruct the block based on the information in the header and the transactions which are in its memory pool. If there is a need for some transaction it will send a request for that missing transaction from the block sender. This approach has the advantage that in the best case it only sends the transaction once and this reduces the amount of bandwidth thereby improving the propagation delay. Fig.21 shows the standard block relaying compared to a compact block with high and low bandwidth relaying.

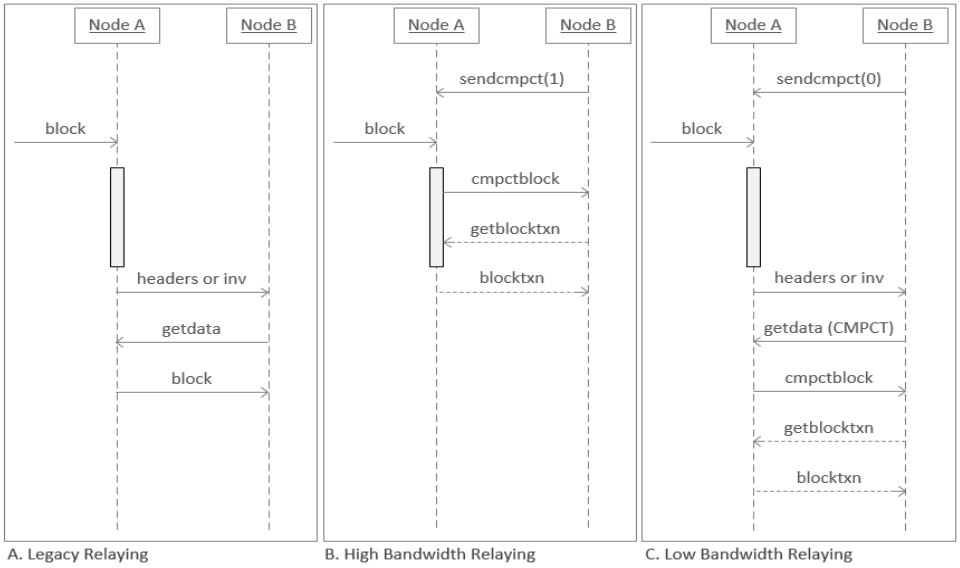


Figure 21:Classic block relaying compared to Compact Block with high and low bandwidths relaying

Tschipper [25] presented an updated model of the compact block protocol called Xtreme Thinblock by adding Bloom Filter to the compact block Fig.22. Precisely when an ‘inv’ message is sent to get a missing block, it sends a bloom filter of its transactions along with the request. This method will reduce the message exchange into two but with a big size compared to a compact block. However, taking the bloom filter into account which produces positive false values affecting the missing transactions.

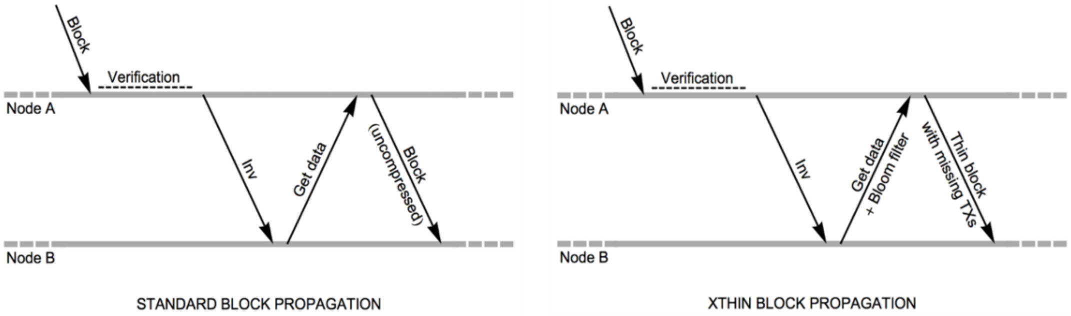


Figure 22:Standard block propagation vs. xThin block propagation

Researchers in [26],[27] discussed using Invertible Bloom Lookup Table (IBLT) to reduce block propagation. The block to be sent computes the IBLT and sends to the peer to compare with the IBLT mempool, and the symmetric difference between them is the missing transactions where the largest IBLT will be returned. However, this method has to be addressed and evaluated formally.

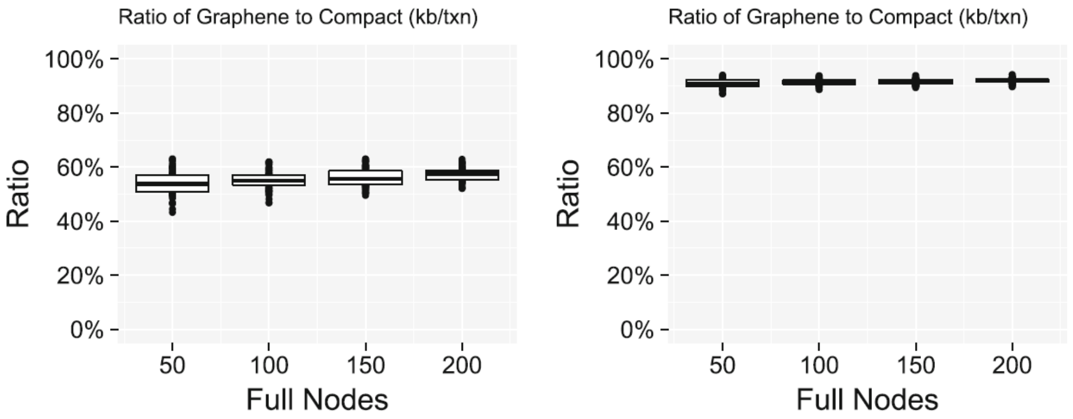


Figure 23:"Graphene reduces traffic to 60% of the cost of Compact Blocks (or to 10% for total traffic, which includes transaction data)"

Ozisik, et al. [28] proposed a protocol called Graphene, that merges the two previous methods: Bloom filter and IBLT to efficient block propagation. The solution used the Bloom filter to compute the symmetric difference between the mempool and the block and then applied ILBT to recover from Bloom filter errors. In detail, the sender sent the header, ILBT I and Bloom filter S of the block transaction IDs. The receiver uses S to find out m' which is the transactions found in S, then recovers from error by computing I'=ILBT(m') to decode it with I. If I-I' is decoded, then the transaction IDs are included on the block. They claim that their solution reduces bandwidth overhead to about 60% compared to Compact Blocks as shown in Fig.23.

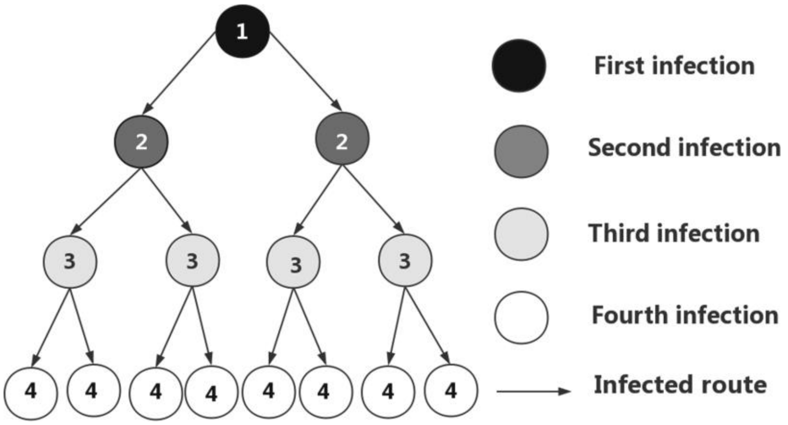


Figure 24:Structure of network broadcasting

A Tree Based Network routing protocol is presented by Kan, Jia, et al. [29]. The concept of this protocol is to disseminate the messages based on the tree structure. As a result, they claimed that it can speed up the broadcasting process and minimize path duplication. When a node joins the network, it is connected as a leaf node. In the case of exchanging a message, the originator node propagates the message to its parent and two children. The message will then be forwarded to all others except the sender node. Fig. 24 shows the structure of network broadcasting.

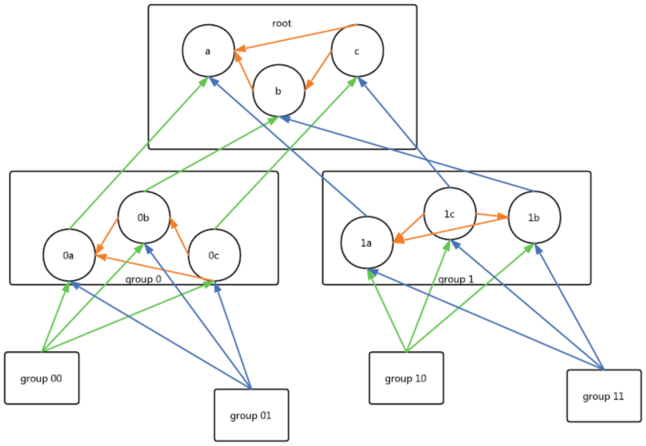


Figure 25:Broadcasting tree based with clusters

However, because the message is sent in one direction, the protocol is vulnerable to single point failure. They overcome this problem by using cluster groups, in which each group has three nodes connected to each other, each node connected to a parent and children on other cluster groups. In the case of node failure, the cluster is still connected by buddies or children in another cluster group (see Fig.25).

In Table 4, a comparison of different countermeasures stated earlier is presented:

Table 4:Comparison of different countermeasures (1)

RESEARCH	METHOD	FEATURES	LIMITATIONS
Decker, et al. [9]	1. Minimize verification 2. Pipelining block propagation 3. Connectivity increases	1. Speeds up transaction propagation 2. Resistant to partitioning attack	Allows fake transactions to flood the network. Bandwidth overhead
Neudecker, et al. [10]	Bitcoin Simulation Model to find partitioning attack on the Bitcoin network	Control of 6000 of the peers gives less chance to attackers to exploit the partitions on the network	-
Karame, et al. [16]	1. Waiting for a period of time after receiving transaction to detect conflicting transaction 2. Set observer node to relay all transitions to vender 3. Adopt node to alert about conflicting transactions	Detect double spending attack on Bitcoin fast payment	Double spending can still occur so the basic problem is not solved
Bamert, et al. [18]	1. The merchant should not accept a direct incoming connection from the sender 2. Merchant has to be connected to large random nodes	Minimized chance of double spending problem in fast payment	Attacker could still be propagated to the majority
Gervais, et al [11]	Studying and devising various optimal strategies for double spending and selfish mining PoW blockchain	Presented a novel quantitative model that analyzes different implications on PoW blockchain	-
Fadhil, et al. [12]	Measurements for simulating Bitcoin network	Transaction propagation measurements Bitcoin network measurements	-
Fadhil, et al. [14]	1. Clustering based on super nodes 2. Clustering based on locality 3. Clustering based on ping time protocol 4. Master node-based clustering	Improves the propagation delay significantly	Vulnerable to partioning and eclipse attacks
Stathakopoulou, et al [19]	1. Pipelining messages 2. Increases connectivity of the geographically closest nodes	Enhance information propagation delay	Allow non-existent transactions to be exchanged
Pappalardo, et al. [20]	Analysis of the Bitcoin network and observation of the transaction and block dissemination	Incentive miners with high enough fees	-

Table 5:Comparison of different countermeasures (2)

Marçal [15]	Algorithm to predict miners or peer nodes connected to them	Decrease the number of duplicated advertisements over the network	Implemented the method in a stable network
Sudhan, et al. [17]	Peer selection technique to find the best combination of the number of outgoing connections either randomly or based on proximity	To reduce propagation delay	Potential for eclipse attack.
Eyal, et al. [21]	Decouples the block into two types: block key for leader and microblock for recording the transactions.	Accelerates transaction confirmation and improves the latency	Is vulnerable to selfish mining attack
Bi, et al. [22]	Selecting closest peers based on Round Trip Time	Accelerated the propagation process and gives a better performance	1. Implemented the method on a small number of nodes 2. Decreases the randomness of connecting to peers
Corallo [23], [24]	Sends compressed block rather than the whole block	1. Sending the transaction once in the best case 2. Reducing the amount of bandwidth	Node has to receive the transaction initially before block exchanged
Tschipper [25]	Adding Bloom Filter to the compact block	Reduced the message exchange into 2 but with a big size compared to compact block	Positive false of bloom filter values
Researchers in [26],[27]	Using Invertible Bloom Lookup Table (IBLT)	To reduce block propagation	Needs to be evaluated formally
Ozisik, et al. [28]	Bloom filter and IBLT (Graphene protocol)	Efficient block propagation	A node must have 15% or more in mempool of the propagated block
by Kan, Jia, et al. [29]	Tree Based Network routing protocol	Speeds up broadcast process and minimizes duplication	Vulnerable to single point failure

CONCLUSION

Bitcoin networks are vulnerable to security risks due to delays in information propagation. In this section, we initially highlighted different studies and countermeasures for the propagation delay in Bitcoin networks. In the next Chapter, a propagation mechanism proposed to improve time delay in Bitcoin network.

CHAPTER IV

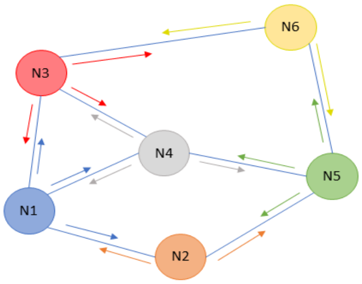
IMPROVEMENT OF INFORMATION PROPAGATION DELAY

INTRODUCTION

As mentioned above, that propagation delay is responsible for inconsistencies in replication and slowing in transaction verification. Most security issues occur as a result of this, and attackers take advantage of nodes conflict to do their malicious activities like double spending attack. One of the reasons which delay the propagation is the number of connections between the nodes. Each node maintains 8 to 10 outcome connections and more than 100 income connections, which makes the process of delivering the transaction more complicated. In this chapter, a method of minimizing the dissemination of information between nodes is explained to reduce noncompulsory connections that are not important to propagate the information.

PROPOSED METHOD

The default propagation mechanism in Bitcoin depends on gossip like protocol. Where each node sends an 'inv' message to all its connected peers even if the message comes from one of them because it applies the same protocol for each node. Figure 26 describes how the protocol works and messages transfer.

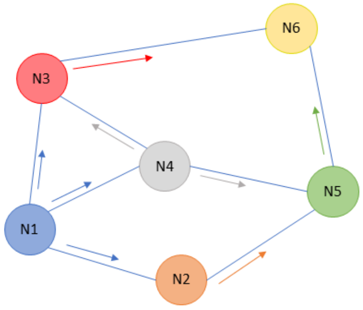


N1 → N2, N4, and N3
N2 → N1 and N5
N3 → N1, N4, and N6
N4 → N1, N3 and N5
N5 → N2, N4 and N6
N6 → N5 and N3

Total number of messages are 16

Figure 26 :Messages transfer between nodes for the standard protocol

The proposed method depends on reducing messages to the nodes that already have the transaction before, by knowing those nodes when the node receives a message. By applying this method, the number of messages will reduce approximately by half, regardless of the nodes that do not have a direct connection to the node, or it is not clear to the node whether the peer has received the message from another peer not connected to it. The following figure shows how messages decrease to half.



N1 → N2, N4, and N3
N2 → N5
N3 → N6
N4 → N3 and N5
N5 → N6
N6 → nothing

Total number of messages are 8

Figure 27:Messages transfer of Proposed method

It is worth noting that message transfer between nodes represents a semi directed graph which guarantees transaction delivery to most of the nodes. As we mentioned in Chapter 3, the number of nodes and the network topology has a direct impact on the dissemination and time delay, therefore an experiment of the proposed method will be implemented in the next section.

THE MODEL EXPLANATION

A survey has been conducted to find out the most appropriate model for implementing the proposed method and we discovered that the method can be implemented using network simulator NS3 which has the feature of simulating different networks and internet environment systems in discrete events (more details about the code will be explained in the appendix).

To verify our proposed method, we used a P2P network build with the following specification:

1. There is a predefined number of m nodes connected randomly to each other using NS3.
2. There is a mechanism for message propagation between them.
3. Each node has the feature of sending a message to its peers randomly.
4. There are two protocols for propagation:

Standard Protocol:

This protocol simulates the Bitcoin propagation protocol, where node N can send an 'inv' message to peers connected to N directly. Once the peer has received it, it checks if it had the message before. If it has not received the message before, it will reply with 'getdata' and N will send the data to the peer. If the peer had received the message before, no action will be taken. Once the peer receives the data, it applies the same protocol (send 'inv' message to every node connected to it even if some nodes had it before).

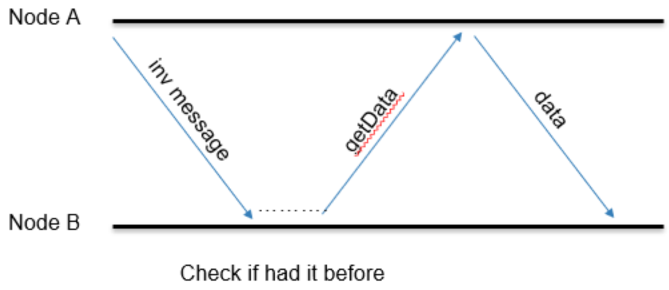


Figure 28:Standard propagation protocol

Proposed Protocol:

Node N sends data directly to any peer and appends its ID in an array with a size of 8. Once the peer receives it, it checks the array. If the peer is there it ignores that node and sends the data to other peers which they are not included in the array list after it adds its ID to the array. Once the array is full, old ID will be deleted and new ID will be added.

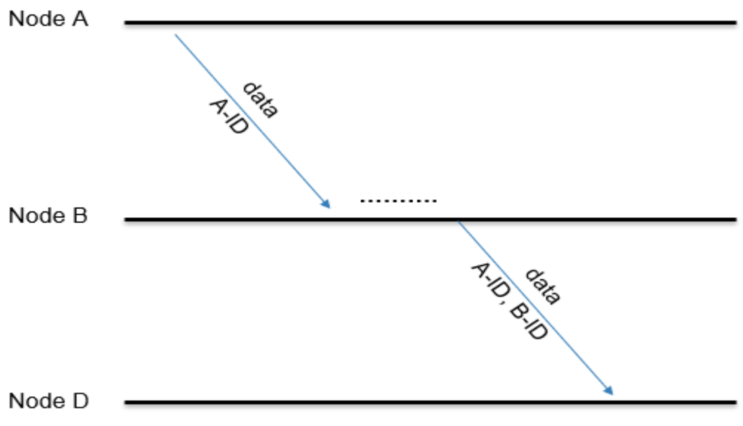


Figure 29:Proposed propagation protocol

5. At the same time, propagation delay is computed for both protocols, such as:
   t = timestamp and N propagates a message at time Ts, it is received by its connected nodes at different times (T1, T2, T3,…,Tn). The time differences between the first transaction propagation and subsequent receptions of the transaction by connected nodes were calculated according to Eq:

   $$\text{Delay Time} = \sum\left(\frac{Tr - Ts}{m}\right) \qquad \text{(Eq.1)}$$

   Where:

   Tr : PacketReceiveTime.

   Ts: PacketSentTime

   m: total number of messages

After computing the delay time for both protocols, we can evaluate our proposed method.

IMPLEMENTATION DETAILS

The proposed method is represented by applying a dynamic array with the size of 8 send with the transaction which contains the IDs of the nodes that have already received the associated transaction. We specify the size of the array depending on the number of outgoing connections for each node that is identified by 8 to 10 connections. Considering the node is known to all its peers, each node registers its ID in the array before sending the message which contains the transaction and the array of nodes IDs. When the peer receives the message for the first time, it includes its ID to the array and forwards the message again to its connected peers. The array will be updated with new IDs when it reached the size limit by deleting the old ones.

To ensure the result, we applied the model four times using different numbers of nodes: 10, 100, 1000, 10000 nodes respectively. According to Bitnodes, a website developed to estimate the current size of reachable nodes, the number of reachable nodes is approximately 9300 for this research. For this reason, the maximum size of the model was set to 10000 nodes to get results close to the real system.

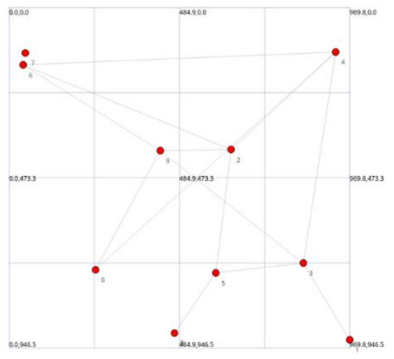


Figure 30:The network topology when number of nodes is 10

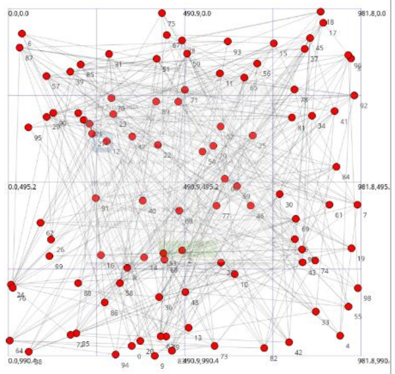


Figure 31:Network topology when the number of nodes is 100

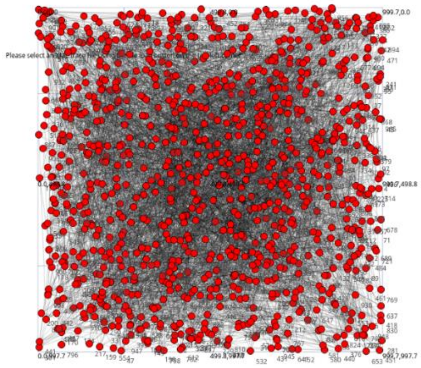


Figure 32:Network topology when the number of nodes is 1000

The figures 30, 31 and 32 represent the network topology when running the model with 10, 100, and 1000 nodes respectively. The observation of the network topology with the number of nodes at 10000 is not clear because of the large number of nodes.

In the next section, the evaluation of the results and performance will be discussed as well as how the throughput and propagation time changed when applying the proposed model in comparison with the real system protocol.

EVALUATION AND PERFORMANCE

By applying the method, there was a decrease in the number of outgoing connections observed and an enhancement of propagation delay from *0.0025t* to *0.0014t* when the number of nodes was 10, whereas the propagation delay enhanced from *0.0112t* to *0.0075t* when the number of nodes was 10000.

Throughput

The throughput can be measured in bits per second (bps) as follows:

$$\text{Throughput} = \frac{Packet\ Size}{Delay\ Time} \qquad \text{(Eq.2)}$$

Table 5 below describes the throughput of the proposed method with different numbers of nodes. There is a noticeable change in performance when the number of nodes increases.

Table 6:Throughput of the proposed method with different number of nodes

Throughput				
No. of nodes	10	100	1000	10000
Standard	0.38349	0.32012	0.2392	0.0845
Proposed	0.6579	0.5381	0.3087	0.1257
% change	72%	68%	29%	49%

In figure 33, the chart shows the improvement in the performance of the proposed method and reveals how increasing the nodes affects the performance.

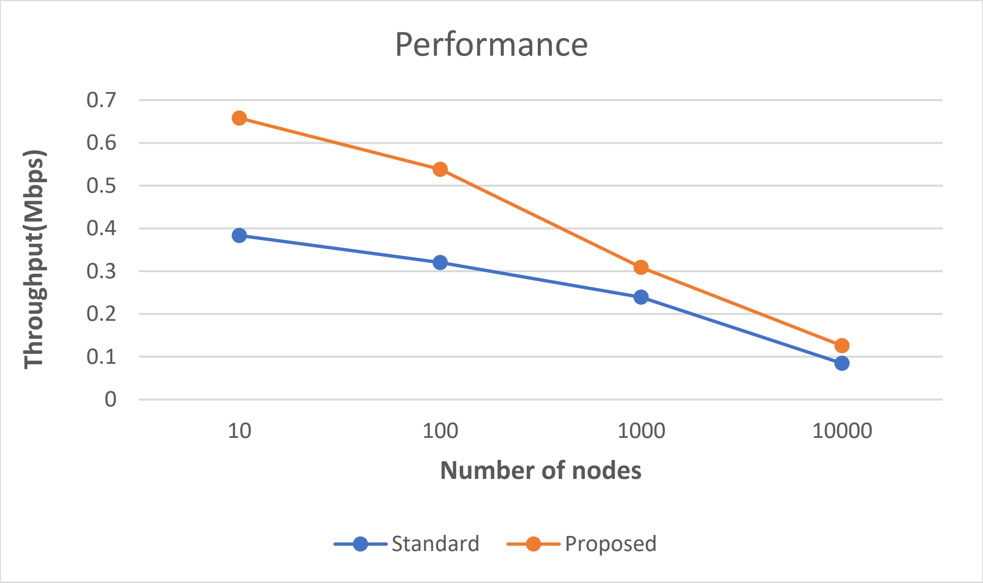


Figure 33:Graphical representation of throughput when applying the proposed model

Delay of Time

The delay of time specifies how long it takes for a bit of data to travel across the network from one communication endpoint to another, which we can find through the following relationship as mentioned in equation1:

$$\text{Delay Time} = \sum\left(\frac{Tr-Ts}{m}\right)$$

The main idea of the research is to minimize the propagation delay. In the results, the proposed method provides a respectable result especially when the number of nodes increases. Table 6 shows the results of the time delay for both the standard method and the proposed method.

Table 7:The time delay with both standard and proposed method

Delay Time				
No. of nodes	10	100	1000	10000
Standard	0.00248683	0.002979	0.003987	0.011286
Proposed	0.001449573	0.001772	0.003089	0.007587
Improvement rate	42%	41%	37%	33%

The result shows improvement in the delay time with different numbers of nodes, mostly when the number of nodes increases. The method is effective with a large number of nodes because it depends on knowing a large number of peers. The test of 10000 nodes demonstrates an enhancement by 33% whereas 10 nodes provide 42%. Regarding 1000 nodes, the result displays a decrease in rate compared to the text on 100 nodes, and that's due to different connecting peers.

The following chart graph represents the improvement in the propagation delay with the proposed method.

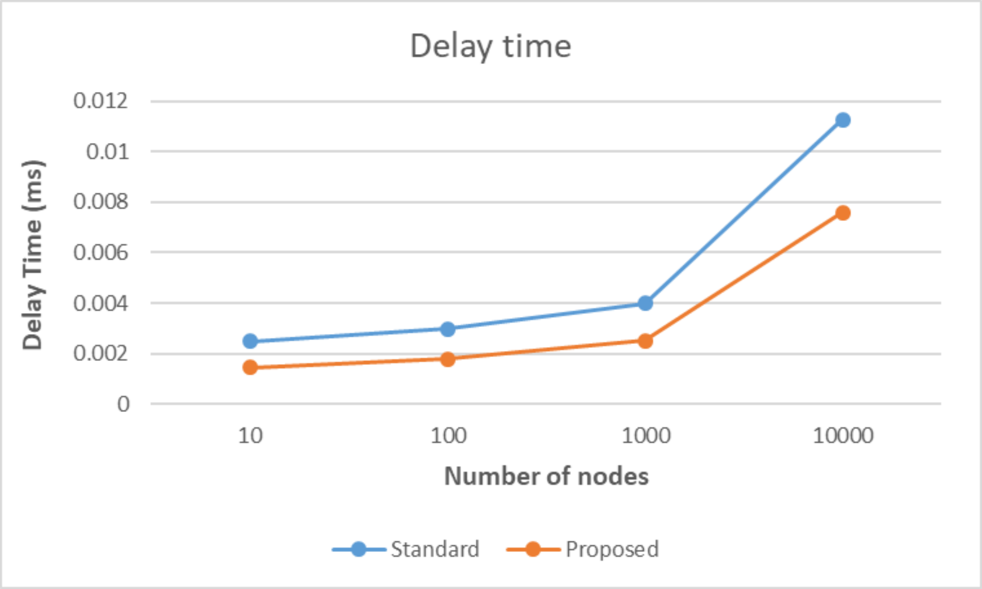


Figure 34:Time delay improvement using the proposed method

Transaction delivery ratio (TDR)

As mentioned before in Chapter 3, not all the nodes, except in rare cases, receive the transaction during dissemination. By applying this method, every node concentrate on delivering the transaction to the peer that is not registered in the list of peers that are aware of the transaction. We can find the TDR from the following equation:

$$\mathrm{TDR} = \frac{Number\ of\ Received\ Packet}{Number\ of\ Sent\ Packet} \qquad \text{(Eq.3)}$$

Table 8:Transaction Delivery Ratio of the proposed method

Transaction Delivery Ratio				
No. of nodes	10000	1000	100	10
Standard	11	35	91	100
Proposed	25	78	100	100
% change	127%	123%	10%	0%

The chart graph in Fig.35 demonstrates the transaction delivery ratio which proves the effectiveness of the proposed method.

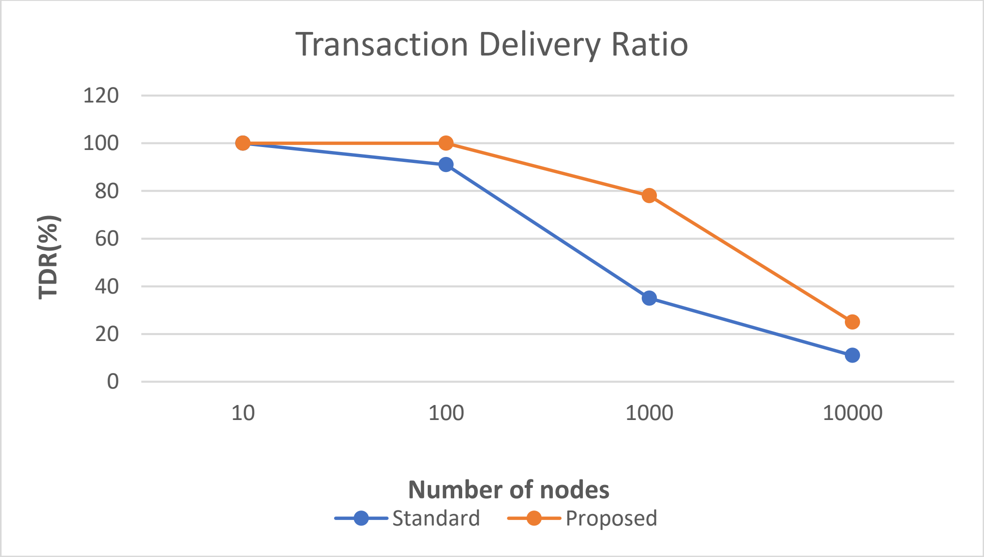


Figure 35:Transaction Delivery Ratio of the proposed method

CONCLUSION AND FUTURE WORK

Through our study of various proposed solutions to the propagation delay problem, we found some important findings have to be consider on this area research:

- Most of the research that addressed the propagation delay and evaluated different proposed countermeasures are concentrated on the four categories we mentioned above: work with consensus protocol, minimize verification time, change propagation protocol, and work with the network topology.

- Because of decentralization of the Bitcoin network and the information having to be transmitted between the nodes, we found that the propagation delay is the fundamental originator for most security issues in the Bitcoin network like replica inconsistencies, double spending attack, partitioning attack, Blockchain forks, eclipse attack, etc.

- The choice of selecting a peer, either using clustering or organizing the network based on some graph, is reducing randomness, and thereby exposing it to various security threats, for instance, selfish mining attack and eclipse attack.

- Bloom filter and Invertible Bloom Lookup Table (IBLT) data structures are effective mechanisms to minimize block size during propagation that have to be addressed and evaluated more.

- There are many tools that might help to reduce information dissemination. Minimum Spanning Tree (MST) is one of them that may facilitate block broadcasting by selecting the best weighted hop among miners.

In fact, we found that there is a potential of eclipse attack occurrence in some cases and we will leave that as a future work by combining of others way of enhancement that mentioned before.

REFERENCES

1. BLOCKCHAIN-DECENTRALIZED TRUST (book).
2. https://freedomnode.com/guides/17/how-bitcoin-works
3. Conti, Mauro, et al. "A survey on security and privacy issues of bitcoin." IEEE Communications Surveys & Tutorials 20.4 (2018): 3416-3452.
4. Antonopoulos, Andreas M. Mastering Bitcoin: Programming the open blockchain. " O'Reilly Media, Inc.", 2017.
5. Nakamoto, Satoshi. *"Bitcoin: A peer-to-peer electronic cash system."* (2008).
6. Yli-Huumo, Jesse, Deokyoon Ko, Sujin Choi, Sooyong Park, and Kari Smolander. *"Where Is Current Research on Blockchain Technology?—A Systematic Review."* PloS one 11, no. 10 (2016): e0163477.
7. Gervais, Arthur, et al. "On the security and performance of proof of work blockchains." *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. ACM, 2016.
8. Wüst, Karl. "Security of Blockchain technologies." Master's thesis, ETH Zürich, 2016.
9. Decker, Christian, and Roger Wattenhofer. "Information propagation in the Bitcoin network." *IEEE P2P 2013 Proceedings*. IEEE, 2013.
10. Neudecker, Till, Philipp Andelfinger, and Hannes Hartenstein. "A simulation model for analysis of attacks on the Bitcoin peer-to-peer network." *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. IEEE, 2015.
11. Gervais, Arthur, et al. "On the security and performance of proof of work blockchains." *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. ACM, 2016.
12. Fadhil, Muntadher, Gareth Owen, and Mo Adda. "Bitcoin network measurements for simulation validation and parameterisation." *11th International Network Conference, INC 2016*. University of Plymouth, 2016.
13. Karame, Ghassan, Elli Androulaki, and Srdjan Capkun. "Two Bitcoins at the Price of One? Double-Spending Attacks on Fast Payments in Bitcoin." *IACR Cryptology ePrint Archive* 2012.248 (2012).
14. Sallal, Muntadher Fadhil. *Evaluation of Security and Performance of Clustering in the Bitcoin Network, with the Aim of Improving the Consistency of the Blockchain*. Diss. University of Portsmouth, 2018.
15. Marçal, Joao Esteves. "Adaptive Information Dissemination in the Bitcoin Network."
16. Karame, Ghassan, Elli Androulaki, and Srdjan Capkun. "Two Bitcoins at the Price of One? Double-Spending Attacks on Fast Payments in Bitcoin." *IACR Cryptology ePrint Archive* 2012.248 (2012).
17. Sudhan, Amool, and Manisha J. Nene. "Peer Selection Techniques for Enhanced Transaction Propagation in Bitcoin Peer-to-Peer Network." *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*. IEEE, 2018.

18. Bamert, Tobias, et al. "Have a snack, pay with Bitcoins." *IEEE P2P 2013 Proceedings*. IEEE, 2013.
19. Stathakopoulou, Chrysoula, C. Decker, and R. Wattenhofer. "A faster Bitcoin network." *Tech. rep., ETH, Zurich,. Semester Thesis* (2015).
20. Pappalardo, Giuseppe, Tiziana Di Matteo, Guido Caldarelli, and Tomaso Aste. "Blockchain inefficiency in the Bitcoin peers network." *EPJ Data Science* 7, no. 1 (2018): 30.
21. Eyal, Ittay, et al. "Bitcoin-ng: A scalable Blockchain protocol." *13th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 16)*. 2016.
22. Bi, Wei, Huawei Yang, and Maolin Zheng. "An Accelerated Method for Message Propagation in Blockchain Networks." *arXiv preprint arXiv:1809.00455* (2018).
23. Corallo, M.: Bip152: compact block relay, April 2016. https://github.com/Bitcoin/bips/blob/master/bip-0152.mediawiki
24. https://bitcoincore.org/en/2016/06/07/compact-blocks-faq/
25. Tschipper, P.: BUIP010 Xtreme Thinblocks, January 2016. https://bitco.in/forum/threads/buip.010-passed-xtreme-thinblocks.774/
26. Andresen, G.: O(1) block propagation, August 2014. https://gist.github.com/gavinandresen/e20c3b5a1d4b97f79ac2
27. Russel, R.: Playing with invertible bloom lookup tables and Bitcoin transactions, November 2014. http://rustyrussell.github.io/pettycoin/2014/11/05/Playing-with-invertible-bloom-lookup-tables-and-Bitcoin-transactions.html
28. Ozisik, A. Pinar, et al. "Graphene: A new protocol for block propagation using set reconciliation." *Data Privacy Management, Cryptocurrencies and Blockchain Technology*. Springer, Cham, 2017. 420-428.
29. Kan, Jia, et al. "Boost Blockchain Broadcast Propagation with Tree Routing." International Conference on Smart Blockchain. Springer, Cham, 2018.
30. Fadhil, Muntadher, Gareth Owenson, and Mo Adda. "A Bitcoin model for evaluation of clustering to improve propagation delay in Bitcoin network." *2016 IEEE Intl Conference on Computational Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed Computing and Applications for Business Engineering (DCABES)*. IEEE, 2016.
31. Fadhil, Muntadher, Gareth Owenson, and Mo Adda. "Locality based approach to improve propagation delay on the Bitcoin peer-to-peer network." *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*. IEEE, 2017.
32. Owenson, Gareth, and Mo Adda. "*Proximity awareness approach to enhance propagation delay on the Bitcoin peer-to-peer network.*" 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS). IEEE, 2017.