.

NOVEL ARTIFICIAL INTELLIGENCE METHOD FOR DECISION CHAIN WITHIN BLOCKCHAIN TECHNOLOGY

Mohamed Ikbal Nacer

A thesis submitted in partial fulfilment of the requirements of Bournemouth University for the degree of

Doctor of Philosophy

Department of Creative Technology
Faculty of Science Technology
Bournemouth University
Poole, BH12 5BB
United Kingdom

Abstract

The objective of the distributed system is to distribute the resources and the calculations. Blockchain is the art of interconnecting data into a tamper-proof and tamper-resistant ledger. Security is ensured by making the cost of malicious activities very high, transparency is inherited from a high level of duplication, and privacy is the result of using cryptography. Consensus is at the heart of the technology to orchestrate nodes to provide finality. However, it has a disadvantage because it bases the decision on different means, which are votes, stake or resources. The decision makes the system prone to monopoly or inconsistencies. In addition, the system suffers from a high validation lag compared to centralized systems. Thus, the injection of a novel artificial intelligence method that can learn and automate the space of actions allow the technology to respond to criticisms of efficiency. This work introduces a new approach in the maintenance of distributed ledger. It will start with the introduction of TheChain as a platform, which is based on the concept of node independence as incentive for competency. Second, TheCoin is the data that will be exchanged between different nodes, which is flexibly modeled to hold different types of symbolic elements. Finally, TheTree is a sociology-inspired approach to maintain validity. It introduced the concept model as a distributed modeling approach and changed decision and security from a component to a network. At TheChain level, monopoly as a philosophical issue was addressed, a conceptual comparison was demonstrated, a security discussion and an operation scenario were investigated. At TheCoin level, discussion of security, conceptual comparison, system size and performance are demonstrated. TheTree section will provide a safety discussion, formal study, environment modelisation and conceptual comparisons. The contribution is to provide a non-monopoly-prone platform built on a new philosophical principle to solve security problems. Second, TheCoin reduce the size of the block and retain the use of coins to offer parallel transaction processing, in which it has been reported that TheCoin can be with 10% of normal block size in case of micropayment. TheTree defined a new approach to dealing with malicious users by leveraging regional consistency. The propagation and consistency times are faster than any previous work. Moreover, the cost of malicious activities has been shown to be very high.

Dissertation Declaration

I agree that, should the University wish to retain it for reference purposes.

Confidentiality

I confirm that this dissertation does not contain information of a commercial or confidential nature or include personal information other than that which would normally be in the public domain unless the relevant permissions have been obtained.

Copyright

The copyright of this thesis has been transferred to the university because the project was entirely funded by it. Parts of the work have been published previously to follow the university policy of ease on information sharing and dissemination. Based on the publisher, it does not require authorization or notification to be incorporated into the Ph.D. Thesis.

Requests for Information

I agree that this dissertation may be made available as the result of a request for information under the Freedom of Information Act.

Signed: ______________________

Name: Mohamed Ikbal Nacer

Date: 04/06/2022

Programme: Doctor of Philosophy

Original Work Declaration

This dissertation and the project that it is based on are my own work, except where stated, in accordance with University regulations.

Signed:

Name: Mohamed Ikbal Nacer

Date: 04/06/2022

Publications and International Recognition

After the publication of our first paper, titled TheChain, I was invited by ACM Singapore to serve on the technical committee of the international conference on Blockchain and AI. Secondly, we received the best presentation award for our paper named TheCoin before being invited again to sit on the technical committee by ACM Singapore. Following is the list of reports:

1. Ikbal Nacer, M., Prakoonwit, S. and Prakash, E, Missa: a regional approach to maintain validity. Under review.

2. Ikbal Nacer, M., Prakoonwit, S. and Prakash, E., 2021. Thecoin: Privacy and security considerations within blockchain transactions. 2021 2nd Asia Service Sciences and Software Engineering Conference, 10–17.

3. Nacer, M. I. and Prakoonwit, S., 2022. The vulnerability of the blockchain network from the consensus perspective. Regulatory Aspects of Artificial Intelligence on Blockchain, IGI Global, 1–20.

4. Nacer, M. I. and Prakoonwit, S. and Alarab, I, 2020. Thechain: A fast, secure and parallel treatment of transactions. Proceedings of the 2020 2nd International Electronics Communication Conference, 81–89.

5. Nacer, M. I., Prakoonwit, S. and Alarab, I., 2021a. Blockchain as a complementary technology for the internet of things: A survey. Internet of Things, Springer, 1–24

6. Nacer, M. I., Prakoonwit, S. and Alarab, I., 2021b. The combination of AI, Blockchain, and the Internet of Things for patient relationship management. Internet of Things, Springer, 49–65

7. Alarab, I., Prakoonwit, S. and Nacer, M. I., 2021. Illustrative discussion of mc-dropout in general dataset: Uncertainty estimation in bitcoin. Neural Processing Letters, 53 (2),1001–1011.

8. Alarab, I., Prakoonwit, S. and Nacer, M. I., 2020b. Competence of graph convolutional networks for anti-money laundering in bitcoin blockchain. Proceedings of the 2020 5th International Conference on Machine Learning Technologies, 23–27.

9. Alarab, I., Prakoonwit, S. and Nacer, M. I., 2020a. Comparative analysis using supervised learning methods for anti-money laundering in bitcoin. Proceedings of the 2020 5th International Conference on Machine Learning Technologies, 11–17

Acknowledgements

First of all, I would like to praise Allah for enlarging my chest with confidence, making it easy for me, and untying the knot in my tongue.

Secondly, I would like to thank Bournemouth University for the generous funding granted to me during the three years spent on the project. It was a great honor to be among the amazing and highly motivated members of the university.

Thirdly, my supervisors were of great help and assistance in completing the project, in which I tried to distil as much of their wisdom as possible. In particular, I would like to thank my first supervisor, Dr. Simant Prakoonwit, for giving me the honor of working on this project. He was kind, helpful, fun, insightful, generous, experienced and visionary.

Fourth, I would like to thank my parents for their support throughout my life to whom I submit and obey, and I hope that may Allah have mercy on them as they raised me when I was young. Special thanks to my uncle Samir Nacer.

Finally, I would like to express my love to my grandparents. Specially to my grandma, the woman I called mother that left this life few months ago.

Contents

List of Figures

List of abbreviations

ACS Asynchronous Common Subset

AES Advanced Encryption Standard

AGM Artificial Intelligence Platform

AI Artificial Intelligence

AKKA Actor Model Platform

BA Binary Agreement

BFT Byzantine Fault Tolerance

BI-Graph Biparty Graph

BOINC Berkeley Open Infrastructure for Networked Computing

CAP Consistency, Anonymity and Partial Tolerance

CBA corrupt but alive

CORBA Common Object Request Broker Architecture

CPU Central Processing Unit

Curl-P Cryptographic Hash Function designed specifically for use in IOTA

DAG Dynamic Cyclic Graph

DAO Decentralized Autonomous Organization

DNS Domain Name System

DPLS Doubly Parallel Local Search

DPoS Democratic Proof of Stake

ECDSA Elliptic Curve Digital Signature Algorithm

FBFT Federated Byzantine Fault Tolerance

FTP File Transfer Protocol

HTTP Hypertext Transfer Protocol

IBM International Business Machines

ICMP Internet Control Message Protocol

IDL2 The Interface Definition Language

IoT Internet of Things

IOTA-Tangle Blockchain Platform

IP Internet Protocol/ Intellectual Property

ISP Internet service provider

ISP Internet service provider

KYC Know Your Customer

MIM Man In the Middle

NP complete nondeterministic polynomial-time complete

ORB object request broker

OSGi Open Service Gateway Initiative

P2P Peer to Peer network

PBFT Practical Byzantine Fault Tolerance

PoB Proof of Burn

PoC Proof of Concept

PoET Proof of Elapsed Time

PoS Proof Of Stake

PoSp Proof of Space

PoUW Proof of Useful Work

Q/U Query/Update protocol

RB Reliable Broadcast

RBFT Random Byzantine Fault Tolerance

RPC Remote Process Call

SHA Secure Hash Algorithm

SMART-BFT Smart Byzantine Fault Tolerance

SMP State machine replication

TCP Transmission Control Protocol

UDP User Datagram Protocol

UG Universal Generalisation

UK United Kingdom

UTXO unspent transaction output

VRF Verifiable Random Function

ZK-snark Zero-Knowledge Succinct Non-Interactive Argument of Knowledge

1 Introduction

1.1 Overview

The development of the internet was followed by a conceptual goal of distributing resources and calculations. The distributed system was the name for managing memory access exclusively from different processes located on the same machine or connected through the Internet. At the theoretical level, it is seen as a set of autonomous entities interconnected with the non-sequential and non-deterministic state without a centralized coordinator. The conceptual contributions of this style of development are transparency, reliability, and performance (Manfren et al. 2011). Transparency is to give the illusion that the set of machines works as it is locally, in which the user can access many versions from different sources without having the possibility of locating the resource with respect to parallelism and competition (Stroud 1992). Reliability and availability are two related conceptual contributions due to the availability of many geographically dispersed versions (Milutinović and Lučanin 2005). Finally, performance is an expected attribute due to shared access to resources and computations. The different processes can cooperate to complete a task such as a Google engine or participate in a race such as the Bitcoin system. However, the system is prone to hacking, distribution failures, and cooperation delays due to software or hardware faults (Abdullah et al. 2017). Thus, the model inherited from the real physical environment must deal with the asynchrony of processing and communication, the absence of a global clock, and the autonomy of each process (Ganguly et al. 2021, Fischer et al. 1985).

At the logical level, the topological structure of the internet leads to routing problems. In the case of cooperation, the access to memory must be mutually exclusive. However, in the event of a contest, the election of a leader is a solution to force a version. Thus, the global state is subjected to routing obstacles above the cooperative and competitive behavior of different nodes, which making liveliness and security, the metric of the conceptual evaluation (Onireti et al. 2019). Causality over logical timing is the art of managing the order between different autonomous entities in addition to evaluating the overall state (Lamport 2019, Kulkarni et al. 2022, Lamport and Merz 2022). However, scalar tracking does not ensure total ordering over all nodes, leading to the introduction of vector and matrix tracking. Many algorithmic contributions have been made to manage memory access by methods such as interrupt (Clavería et al. 2022), alternate on access, Dekker's algorithm (Martin 1985), Dekker and Peterson algorithm (Meolic et al. 2001) or hardware instruction (Buhr et al. 2015). However, CPU consumption was an issue in all proposed solutions, which led to the introduction of SLEEP and WAKEUP mechanisms (Horikawa 2011).

At the topological level, communication between nodes requires a communication protocol to manage the order of information logic. However, reaching a node requires routing that involves uniquely addressing each element. Locating a node depends on each protocol requirement; however, network discovery is an option in an open system. The centralization of the server and the connection within a known location or not, is possible with different validation protocols. Additionally, other considerations such as blocking or timing, storage, and channel reliability, are very important criteria for overall performance. Static routing such as Myrinet consumes a lot of packet bits (Boden et al. 1995). However, adaptive dynamic routing is prone to packet loss for some nodes. Thus, many protocols have been introduced to manage the propagation of information within the network at the leader election level, such as Byzantine Fault Tolerance Protocol (BFT) (Lamport 1984), Practical BFT, Flexible BFT (Castro and Liskov 2002) and SMART-BFT (Chen and Shen 2013). Moreover, many protocols have been implemented to exchange information for web purposes such as HTTP or Ftp. However, a more advanced level of management has been introduced based on competitive racing that focuses on the gain generated from a ledger such as Bitcoin (Segendorf 2014).

The distributed system architecture aims to deal with heterogeneous entities on the physical and software layers to ensure normal operation. The goal is to ensure that remote process calls (RPCs) work the same as local calls. Interoperability was a problem that was solved by CORBA through the use of pre-compilation of IDL2 (Marvic et al. 2000). In addition, common types of variables are used between the communication channels to ensure good reception. Thus, implementing an N-layer architecture on a single peer will make the system prone to low agility as the business logic is replicated but still high coupling between the different functional components (Richards 2015). Also, performance will be low due to the many layers of processing. Scalability in terms of managed information or business logic is very low due to the nature of the implementation. Breaking down the implementation and interconnecting the different components via event-driven models will make the system agile in addition to being very efficient. Additionally, scalability is easily achieved with proper decoupling of independent components. However, in distributed middleware implemented through the use of space-base architecture, the implementation of microservices forces all testability, deployability, implementability, agility and scalability metrics, but with a downside of low performance (Richards 2015).

The structure of the data exchanged between the systems can take several forms. A proxy object is generally an object that contains a reference to the source provider in addition to the signature of the various processes that can be called. Each called process will return an object container that encapsulates the results. However, many other applications in the system share blocks and transactions regardless of standards. Blockchain technology is a secondary application of a distributed system that focuses on the integrity of many geographically separated copies. Organizing the system into a set of unstructured or structured topology has a huge impact on its performance in terms of liveness, partial tolerance and security. However, many technical decisions have been made to offer synchronization, partial synchronization or asynchrony which make the system prone to attacks and performance issues. Thus, the trade-off between different modes of operation, architectures, shared data, and design choices are normal specification strategies in this development domain.

Distributed Intelligent Ledger (AKA: Blockchain technology) has received a lot of attention in recent years since the Bitcoin system was proposed (Segendorf 2014). The intention is to force global consistency on financial data. It is through the construction of a linear sequence of blocks, which is very difficult to modify or reproduce. It has already been explained in (Nabilou 2022). It discourages system flooding with anonymous transactions generated through the use of proof-of-work (PoW) (Back et al. 2002) as a validation mechanism. The peers are distributed within groups and the information is propagated through the use of the gossiping algorithm, which makes its speed relevant in linearity with respect to the number of groups. The data structure is a very important element for the traceability of the information validation, the size and the rapid convergence for the different peers. Many different proposals have been shared in the literature on how to handle data in terms of the UTXO model, Balance, or mixture (Ikbal Nacer et al. 2021).

The first implementation of blockchain at the architectural level suffers from resource consumption issues, low validation rate per second, and security issues (Hayes 2017, Zargar and Kumar 2019, Karame et al. 2015). However, more specifically to each component. The propagation of information by the system is likely to be tracked due to the anonymous communication protocol that communicates directly to the nodes. However, the conceptual attributes at the architecture level show the lack of modularity to build a global machine. Thus, blockchain has turned out to be one of the hottest research areas in recent years. Research has focused on studying all its implications (Sharma and Kumar 2020, Nacer et al. 2021a), components, and system operation. Many versions and improvements to the system providing different approaches to ensure validity such as Proof of Stake (Gaži et al. 2019), Proof of Authority (De Angelis et al. 2018), IOTA. Reconstruct the topology to study information propagation and improve block transaction and broadcast to ensure reliability such as Velocity (Chawla et al. 2019), Stratum (Dotan et al. 2021), and compacted blocks. Moreover, in the case of modularity, many different platforms have been proposed, but only the management of BFT through a platform such as Hyperledger sees modularity as an issue. Finally, security in terms of a combination of consistency and validity was addressed in double-spending, the vision of the truth, and manipulating the information.

Artificial intelligence (AI) used to be a fancy name for automation in all its forms. In the early days of research, it was more focused on rule inference than pattern detection. In recent years, narrow AI models that go by the name of machine learning have received a lot of hype (Voulodimos et al. 2018). It has shown many significant real-world impacts in terms of security monitoring, support, decision automation and more. However, adoption of the technique in many fields such as robotics is prone to decision fuzziness, hardware limitation, and lack of solid theory injected into computer science regarding belief constructs. However, symbolic methods (Perez et al. 2018) are widely used in industries due to their ease of control, especially in planning. Many other approaches have been developed and reported in the literature with emphasis on transition, pattern and architecture. The implementation in real life suffers from many problems, especially on the understanding of the generated bit space, the unavailability of the technique to handle the iterative belief and the ability to handle an open context (Spohn 2012). Also, at the software engineering level, quality assurance, testability, predictability, and design are not well-structured outcomes. For example, the culture of development based on the valid assumption that perception is an unstructured reception of data leads to the global acceptance of chance to generate binary capacity. Also, there is no understanding of different binary distributions leading to many contents represented by a single form (Spohn

2012).

Ibn Khaldoun in (Khaldun 2015) introduced sociology to the world and stated that the state has always been a human choice to maintain justice but questioned that the state itself is a force that acquires power unfairly. The story of an ancient society is summarized as a long road to sophistication that ends with a huge focus on art before a foreign minority with the foundational skills comes to take over. Solidarity among people who speak the same language was the key to maintaining the society internally. However, the focus has been on the cultural conflict, investing in bureaucracy as an internal issue against solidarity. Blockchain's goal is to eliminate the foundation of normal human society, which is the state. It will eliminate the force that uses unfair means to enforce bureaucracy, which prevents human civilization from growing rapidly. The problem of malicious activities can be summarised in the same conflict of nomads with those who are sedentary. The ability of validators to monopolize the system can be seen as the issue of the periodic existence of a foreign minority that possesses the foundational skills. However, graphical analysis of the blockchain ledger has shown many cycles that can be inferred as ways to increase the value of cryptocurrency through a bogus exchange or double-spend events by investing in the longest chain rule. All of these issues can be justified or denied based on the mismatch, transparency, and truth bias of human psychological interaction. Therefore, it will be difficult and unfair to implement a probabilistic model to deal with these issues. On the other hand, the deterministic approach may be appropriate.

Community fostering is an important factor for platform success, in which systems based on PoW aim to foster a huge number of miners to ensure a fair and competitive race to unlock the puzzle (Johnson et al. 2014), which will lead the system to provide security to the users. In addition, the system's success is based on its high usage, which can be described in the case of cryptocurrency as the exchange rate (Smith 2016). On the PoS, the community is to be fostered through a PoW mechanism, then switched to PoS after ensuring a high trust between the huge number of miners and the users (Bentov et al. 2014). However, the solution foundation allows for monopoly by default. BFT, due to the high complexity of message exchange, is not recommended for implementation within the public environment. Its internal implementation is subject to direct manipulation by stakeholders (Wang et al. 2018). The community is the most important factor to maintain the system's deterrence of malicious behaviour, whereas the system role must ensure the different mechanisms to ensure the road for a community to flourish.

1.2 Motivation

Horizontal blockchain adoption has been very high since 2017. China declared in 2019 that blockchain technology is the objective number one for the state (Pomelnikov 2021). According to the UK report on the fourth industrial revolution (Government 2019), the Blockchain is an important criterion. Many solutions have been proposed and implemented in industry to provide a global machine, but most of them are neither agile nor modular. Cryptocurrency as the initial promoter of the solution has become one of the most valuable markets in the world. Bitcoin, Ethereum,and Zerocash are different flavors of the technologies. IBM and many other foundations observed the need and market pull for decentralization and focused on implementing Hyperledger, which uses the Byzantine ideology of fault tolerance. Thus, there are two types of motivation, on the one hand, the software engineering point of view to break the order and provide a product with new features and on the other hand, the technical point of view by taking the challenge of the consensus layer. The following sections are some sectors with the expected contribution of blockchain.

1.2.1 Blockchain for IoT

The need for a trusted party is the main problem addressed by blockchain technology through automating the verification of information. It can be used to optimize various processes such as trade facilitation, identity verification, privacy and ownership support at the same time, it can be used to trace the classification of expensive objects and authenticate tracking goods all over the world. The rapid development of blockchain has attracted the attention of many inventors, developers, and investors. It has been reported that the technique can make everyday life easier. Singh et al. (Ali et al. 2019) concluded that blockchain technology can be a game-changer in the IoT industry based on fundamental IoT security issues. Additionally, it introduced an architecture that separates the miner from the user in an effort to ensure validity. A comparison test was run by Fakhri et al. (Fakhri and Mutijarsa 2018), in which testing a comparable blockchain system against a non-blockchain implementation demonstrated security in terms of communication between different devices in an IoT framework. It has been reported by Buccafurri et al (Buccafurri et al. 2017) that conceptual contributions such as record keeping, coordination among stakeholders, transparency and finality of transactions are also desirable features in the IoT industry. However, since the first proposal of the blockchain network, including how all nodes reached consensus on the validity of a financial transaction, it was clear that this technique could ensure confidentiality, transparency, accountability, traceability and identity management.

Many benefits can be concluded by integrating blockchain technology within IoT sectors such as (Nacer et al. 2021a):

- Non-central failure: Data will be duplicated between many servers in a ledger that is not easily reversed.
- Security in two respects: The information will be attached to a public key identifier. It can provide a level of secret communication between users. Also, central management oriented attacks are not a vulnerability.
- Transparency: The sharing of knowledge between stakeholders and users of the service gives the vision of transparency on the generated invoice.

Moreover, robust functionalities can be integrated within IoT services, such as:

- Micropayments: Micro-use of the service generates micro-payments that can easily be managed in the blockchain system.
- Data tracking: Transparency and monitoring analysis are the normal derivative virtue of technology on different generated and appended data. This is a highly desirable feature for stakeholders.
- Decentralisation of services: Networking services such as DNS can be attacked due to the centralisation. However, a decentralization of the service by blockchain technology can bring greater satisfaction to the user.

1.2.2 Blockchain for Financial Sector

The injection of technology to automate financial services is known to be fintech. Loans and mortgages are very complicated and prone to many mistakes. Know Your Customer (KYC) is a normal procedure that must be the basis of a decision to guarantee access to services, which cannot be implemented because the need for anonymity is high. Another important component is the smart contracts, which have been widely discussed instead of the tangible version in the case of microinsurance (Vo et al. 2017), shipment tracking (Shi and Wang 2018) or manufacturing in Industry 4.0. Moreover, the smart property is to inject objects which will then be possessed and authenticated by the author of the information, such as the patent. In a world parallel to today's state-run financial sector, cryptocurrency is auto-generated coins unlike fiat, which is government-backed. However, those traded tokens are a very abstract concept which can be money, property or information. Rima et al (Rana et al. 2019) evaluated identity management with the use of blockchain technology for better transparency.

The financial sector after the injection of blockchain technology can find many contributions such as:

- Automation: Heavy processes can be easily automated by implementing a flexible distributed smart ledger and smart contracts ideology solution.
- Lower the fee: the huge bureaucratic process made the service within the sector very expensive to use.
- Fast transaction and fraudless: errors will be eliminated hence no data manipulation leading to fast trade execution.

1.2.3 Blockchain in Law

Handling writing and secure access to documents are important functions of the legal sectors before transmitting them to the authority as evidence, which can be resolved through the use of blockchain technology. A smart contract gives lawyers a huge advantage in authenticating information. However, this requires intensive work and effort on the part of the lawyer. Moreover, issues such as the right to be forgotten are widely debated.

The legal sectors can benefit from blockchain technology as follows:

- Increase the reliability of the submitted documents.
- Robust means to manage identity.
- The machine as the new source of law management and application.

1.2.4 Technically

Consensus is at the heart of Blockchain technology. It is the tactic to make nodes compete in order to maintain validity. Consensus approaches suffer from many problems that prevent technology from the worldwide adoption expected to function as a world computer. Transaction delay of validation is either caused by mining process in the PoW, time to come to global consistency in voting models, or probabilistic finality due to randomness inherited in algorithms such PoET or PoS. Furthermore, the data structure suffers at the legal level from the incapability to respond in the case of the right to be forgotten or the request to trace the exchange of coins. Thus, privacy is questionable and the act of manipulation of the value of information, which is the base for money laundry, is highly enabled. The exchange of private information such public keys is through different online means such as chats, social media, or even by trusting a third party to withhold it. This criterion goes against the foundation of blockchain technology to break the bound of trusted authority. It must offer traceability of the coin generation and its association with purchased information, but preserve privacy. On the state level, technology suffers from an unmet legal requirement. Firstly, the anonymity of businesses that leads to difficulty in extracting taxes. Secondly, the generation of reward must be based on something of values such as the state infrastructure and its taxation service. The latter two arguments of common controlled rewards and tax are foundational for a society to flourish by investing in itself and rewarding the best.

On the artificial intelligence proposed approach there is a huge lack of dynamicity in all approaches. Symbolic approaches based on formal logic have been investigated for a long time. AGM framework is the most complicated implementation to serve in reasoning. The AGM is a framework that has been implemented to study epistemological theory using the qualitative approach of formal logic. The system has three functions that describe its growth: expansion, contraction, and revision (Kern-Isberner et al. 2019). A revision will address rules that can be misunderstood to generate an unpredictable sequence of actions (Spohn 2012). Much work has been done to manage uncertainty above this domain, such as fuzzy logic (Zadeh and Aliev 2018), possibility theory (Mei 2019), and plausibility (Lai 2019). However, based on Gödel's incompleteness theorem, it is impossible to achieve infinite learning using the available formal logic because any system depends on an external assumption made by ourselves in the first place (Iacona 2021) Machine learning models are mean to learn automatically pre-set expected rule of data distribution. Bayesian network was an alternative to managing uncertainty. In the Bayesian ideology, it is irrational to be certain, there is no suspension of belief, it can describe content with many representations, and there is no support for iterative learning (Spohn 2012). Deep learning has enjoyed hype due to the growth in CPU capability. However, it suffers from overparameterization (Zhou 2021) that led to the capability to generate many representations. It is static and close to the impossible to update learning at certain stages due to the use of probability (Spohn 2012). It suffers from the inexistence of theory to manage learning of the final array of map bits. Many other low level learning models and approaches such as decision tree, support vector machine and reinforcement learning have been proposed but all suffer from the incapability to deal with an open context and model the world (Spohn 2012). However, blockchain with what is now called machine learning at the consensus layer and from market perspective are two parallel lines that never meet for two reasons. First, the blockchain basic requirement is privacy based on complete anonymity and user information in the system should be limited to a huge unreadable key, this key is associated with a basic simple information or a value, to make the case worst many users use many keys/addresses at the same time. Secondly, Blockchain operates in an open context and requires a platform capable of providing unlimited, iterative and rapid variables updating. However, user data remains traceable in the network and many machine learning tactics can be applied later.

The solution provided in this work is a means to respond to philosophical limitations by playing up the principles. The thesis is motivated by the following needs:

1. Addressing the data structure to lower the load of data
2. Provide means of traceability but preserve privacy.
3. The need for a novel method to be the background of a world machine.
4. Increase the efficiency of the system by reducing the finality time.
5. Respond to the legal requirement on the personal and state level.
6. Increase means of privacy in the system.
7. Provide a solution that trades off between real world requirements and fast propagation, treatments, and global decision of transaction.

1.3 Aim and objectives

The blockchain system has been thoroughly studied from a philosophical point of view, data structure choices, performance, incentives, and activities modulation. It was concluded that technology suffers from many issues that prevent it from being a mainstream technology. The heart of the technology lies in its consensus layer, as it coordinates between nodes, which means delay of propagation and consistency are inherited from it. Moreover, security problems are derived from it. Thus, the aim of this thesis is the following :

" Propose, design and develop a new approach within the distributed systems methodology capable of orchestrating nodes in a faster consistency time."

The goal is achieved through different steps called objectives. Thus, setting the philosophical background, tactics, model and incentives in a certain order is essential. Follow our objectives in order to achieve the goal.

1. Propose, design and implement a novel platform that can break from the philosophy of CAP (consistency, availability and persistence) theory.
2. Change the consensus layer from the competence on global data consistency to cooperation on data and the competence on customers.
3. Consider the legal requirements.
4. Model a data structure that reduces size for faster propagation.
5. Empower topology hiding, reduce reliance on trusted parties, and provide total order at the transaction level.
6. Initiate the transaction on the recipient side.
7. Change in decision and security from a computing component to a network.
8. Provide a new incentive for cooperation and operation
9. Design and implement an algorithm inspired by sociology to orchestrate all nodes taking into account all the previous objectives.
10. Evaluate, test and discuss different implementations, in which each model or algorithm is verified and validated.

1.4 Vision

Blockchain technology suffers from many problems that have been mentioned earlier. TheChain, TheCoin and TheTree are proposals that address different layers. Each layer has offered solutions to many problems of technology to enable it to serve humanity with a greater cause of global transparency, personal privacy, and speedy processing of bureaucratic service in terms of tangible or intellectual properties. First, TheChain will leverage the structure of the Petri net to build a ledger to force predictability in growth through controlled modularity. On the validation layer, another soft Petri net will be built and through graph reachability, the advanced contract logic can be verified to guarantee its future validity. TheChain section will be dedicated to introducing the concept of region intersections to ensure the validity of the ledger and defining the global governance algorithm. Thus, this will be the basic background that will provide great flexibility for adding data, quick logic checking, and discussion at the general level of TheTree's operation.

Second, TheCoin will address privacy, quick search, and security. As stated earlier, it is an organized and traceable data exchange protocol enabled with exchange techniques. TheCoin will optimize block size by eliminating duplication of data structure, enabling traceability and enhancing security by providing a secure exchange of shared personal information. Thus, the fuzziness will be used to manage the partial use of the sequential use of non-duplicate coins. Nested mobile agents with zero knowledge proof to be either verifier or prover. It will provide topology hiding and secure sharing of personal information. Moreover, the initiation of the transaction by the sender makes probabilistic finality subject to many network attacks. TheCoin will be functioning by signing the coins from the sender and the transaction will be signed and initiated by the receiver. The advanced contractual decision may involve the generation of invoices, which will also be offered as part of TheCoin protocol. Finally, the data structure must be modeled in such a way as to allow the management of more advanced symbolic elements.

Third, TheTree imports social behavior into the system by investing in human nature by recognising that regional exchanges are a normal habit of social interaction. Additionally, social validity and authenticity are verified by providing democratic access that can duplicate and compete over payable knowledge. Above the TheCoin model, TheTree will ask the question: "If the state has always been a chosen force, why are the bureaucratic institutions not distributed among us?". First, It will explore how clusters are constructed, the interaction and belief between different entities are managed, with the goal of ensuring high scalability and validity. Second, knowledge management as advanced decision-making aims to be satisfied by the proposed approach, in which the concept model has been proposed to ensure a highly flexible development approach for a dynamic framework. Finally, the cost of malicious activities is a very important criterion to ensure high performance of the system. The conceptual model will be the basis for modeling decisions as a network based on the social trend, security is also to apply high reputation destruction by taking advantage of the network. Thus, we try to imitate nature in its physical behavior to provide a new vision of artificial intelligence by subjecting it to normal human needs and the results of its daily interaction.

The whole vision of the system is to provide a new web where the user's view of truth is reputable, authentic and part of a regional preference that forces different versions of consistency. The web can be used for any type of value or managed information. Moreover, the modular dynamic growth of the system is based on a conceptual basis to generate a decision based on a network that explores different paths, making regional consistency another term for different objects.

1.5 Challenges

Blockchain as a platform is an intersection of many components in a small domain of operation. Each component belongs to a field that involves many technical and philosophical issues. First of all, networking is prone to many attacks like DDOS, ping flooding, ICMP flooding, eclipsing, man in the middle. Second, node organization is a requirement for rapid information propagation. Third, the networking flow is controlled by (Internet Service Provider) ISPs. On the consensus layer, there may be many problems to prevent it from achieving rapid finality. The agreement between the nodes is conditioned by the propagation of the network, the honesty of the peers, the competence or the conflict resolution mechanism in case of different view from the users. Users may have to trust a third party for their keys. Additionally, sometimes it is necessary to have a highly sophisticated wallet to record zero-knowledge evidence. The data structure is shared with low efficiency on size and likability with respect to search.

A new blockchain proposal must address and balance the different concepts. It has to compromise in a way that can deliver user satisfaction. Satisfaction is seen as a perceived level of integrity and trust in the system. Integrity is a natural derivation from a high level of duplication and authenticity. Bitcoin claims non-reversibility as an additional contribution to strong integrity, but it has been viewed with some skepticism as the ledger is not non-irreversible. However, many other concepts need to be addressed such as robustness through a high-level submission, the view of truth needs to be consistent, and data traceability needs to be modular. The developer side should address concepts such as testability, agility, ease of integration and development. The user is more concerned about privacy which may be compromised by server tracking ability, monitoring which may be compromised by ISP data flow control, responsiveness as a normal opposition to server availability and finality as a normal opposition to server consistency.

The new model proposal must balance the different concepts. It must observe robustness with an eye of security. Consistency from the point of view of preferences. Modularity from a vision of unlimited scalability in terms of data and nodes. If modularity is adhered to force predictability, the system will be flexible, agile and easy to address concepts such as interoperability which can be a derivative of integrating new components. Furthermore, adhering to the principles that define consistency will impact availability and provide users with fast finality and responsiveness.

1.6 Key contribution

TheTree is the core proposal, and it is based on different principles that define consistency within a distributed system. Moreover, it was built on the fundamental proposition, which is TheChain. It uses a new model named TheCoin for data exchange. This part will discuss the main contributions of the thesis.

1.6.1 Critical literature review

An in-depth study was provided in terms of literature review of existing methods at the consensus level. It was followed by a critical analysis of their functioning and limits. A philosophical limitation of their adoption. More specifically, Chapter 2 defined the different components of the system in the literature. It can be summarized as follows:

- The Blockchain platform was explained in terms of different components to build a system and a different conceptual idea surrounding the technology.
- The consensus as a concept was explained before discussing the various proposals within blockchain technology.
- Each consensus proposal was criticized.
- Networking and cryptography as a highly applicable part of the operating system were also covered.

1.6.2 TheChain as platform

TheChain is a platform that plays on the distributed system principle with the assertion that the requirement of the system is to meet user satisfaction. It changed the global consistency to the regional one and argued that the principle of network structure should be used to force rapid gossiping. Design of an algorithm capable of handling these concepts and also generating their existence. It can be summarized as follows.

- The proposal of a novel vision of distributed systems as an overlapping regional consistent set of worlds
- The design of an algorithm that can build region of an operating territories.
- The design of an algorithm that governs the nodes.
- Design of an algorithm that address the layer of injection by forcing a total order over transaction layer.
- The demonstration of the rules of system through the first order logic to show inconsistency in previous claims before building of the new proposal.
- The introduction of the concept of node independency.

1.6.3 TheCoin as data structure

TheCoin is the proposed data structure that handles the exchange of valuable data between different servers. The solutions aim to solve different layers of problems within the blockchain transaction. TheCoin's contribution can be summarized as follows:

- A new approach to initiate a transaction on the receiver side using mobile agents.
- A model to offer flexible and efficient use of coins to preserve parallelism and reduce size.
- The implementation of a search algorithm to operate on the proposed model to demonstrate high performance.
- The introduction of the concept of authenticity of the coin.

1.6.4 TheTree to force integrity

TheTree is the algorithm proposed to force the different nodes to be honest. This will be the core of TheChain system and uses TheCoin model. The solution addresses the jurisdiction between the nodes in the blockchain network by defining different incentives and security mechanisms. It can be summarized as follows:

- The proposal of a new algorithm to maintain the validity of the ledger.

- The proposal of territorial consistency and incentive to force reliability
- The design and implementation of decision and security as a network.
- The introduction of the concept model.

1.7 Method of working

Our working method complies with the usual IT standards. First, we start by doing a survey study regarding the different components of the platform and the techniques previously proposed. In the second step, we describe the proposal by arguing its contribution on the philosophical level. Third, we prove its validity through the use of the proof of concept (PoC) by implementing the different algorithms and testing them under the expected requirements and scenarios.

1.8 Thesis organisation

The following work will be presented in several stages. First, it will be a general overview of the different parts related to the system. The technology will be presented in its general forms. Next, we will discuss the cryptographic scheme used in the technology. Next, it will introduce consensus before embarking on different adopted consensus algorithms such as Proof of Stake (PoS), PoW, BFT, and Useful Work. Also, there are few highly related concepts such as networks and architectures. Second, we first address our specification of the proposed system, described in TheChain. Next, we discuss our proposed data structure and its associated protocol, named TheCoin. Next, TheTree, which is the heart of the system. Each chapter is self contained with specific related work for readability purposes. Finally, the conclusions is a section that provides discussion on the conceptual contribution, future work and main conclusion.

2 Background

2.1 Blockchain

Blockchain technology was born as a distributed approach to undermine the centralization of financial services. The Bitcoin (Nakamoto 2008) system generates a tamper-proof and tamper-resistant ledger. The ledger consists of a sequentially ordered list of blocks, where each block contains a list of transactions, a Merkle tree, a public key or addresses, signatures, and other additional information. It works on a peer to peer network which uses both user datagram protocol (UDP) and transmission control protocol (TCP) to accomplish different tasks. The generation of each block is conditional on solving the PoW (Back et al. 2002) puzzle. PoW is about a random search to find a nonce number which will be hashed with the string aggregated from the block variables to ensure a fixed number of leading zeros.

The use of the sequentially appended information register to manage different accesses to a document was first used in (Haber and Stornetta 1991) by Haber and Storneta. Additionally, the first use of the PoW idea was to combat email spam (Back et al. 2002). The solution was quickly followed by numerous proposals to improve it. The Ethereum foundation has proposed the use of contracts inspired by the work of Nick Szabo (Szabo 1997). Its objective was to provide a blockchain vision for managing the exchange of business information. ZeroCoin (Miers et al. 2013) used zero-knowledge proof to provide a higher level of intrackability to users. The Hyperledger fabric is an approach to meet a high level of information management away from currency. In addition, Ethereum introduced the concept of gas pricing associated with resource's consumption of PoW. However, the intention is to move from using PoW to PoS. Hyperledger releases use BFT techniques with an emphasis on transaction ordering. At the application level, many proposals have been made for crowdfunding (Hartmann et al. 2019), voting (Kshetri and Voas 2018), IoT usage (Reyna et al. 2018), and government enforcement (Hou 2017).

The system has the potential to offer high data maintenance ensuring its validity. However, issues related to trust that validators will not collaborate, trust in the irreversibility of the cryptographic techniques used, trust in the continued interest of anonymous random validators in the system have to be met. In addition, technically, the system suffers from a high potential for forking in the event of a low difficulty number of PoW (Jameel et al. 2020). The network over which the system extracts information from other peers is vulnerable due to the lack of standards introduced for it. Network performance tracking and undermining tactics have been widely discussed in the literature (Ben Mariem et al. 2018, Wüst and Gervais 2016, Ikbal Nacer et al. 2021). The wallet as a component that holds keys, identifies and verifies ownership of data generated in the system, is normally subject to all mobile-related vulnerabilities. Nevertheless, due to the ensuing hype of the cryptocurrency, a lot of research has been published in the literature to demonstrate the ad hoc proposed solution to deal with many problems of the network (Leiding et al. 2016), architecture (Ismail and Materwala 2019), of cryptography (Raikwar et al. 2019), communication (Dotan et al. 2021), protocols (Xiao et al. 2020), consensus (Mingxiao et al. 2017) and use cases (Nacer et al. 2021b). The Blockchain solution has changed a lot since the first white paper, today it is more understood system from a software engineering point of view.

2.2 Cryptography

Cryptography is an element to ensure trust within the blockchain system. A hash function is a basis for generating a hash that secures PoW requirements. Moreover, addresses, signing, public and private key generation are all a hot research era in the field of cryptography (Lipmaa et al. 2022, Fauzi et al. 2021, Yuan et al. 2017). The key size was one of the issues for the post-quantum adoption of blockchain technology. The hash function (H) that takes an input and generates a hashed output must have three properties (Wang et al. 2018). First, preimage resistance is the inability to supply H(y) to find y. second, the second resistance to preimage is the inability to find b, which has H(b) = H(a) by providing a and H(a). Third, collision resistance by finding two hashes generated from different inputs a and b that have H(a) = H(b). NIST research in the 1990s led to the introduction of SHA-2 which was followed by major investments in SHA-3 standardization, testing and implementation. The IOTA solution proposed using the Curl-P function has been widely criticized by crypto communities (Heilman et al. 2019).

The Merkle tree is built through a recursive iteration over the list of transactions. It will combine every two transactions to generate a hash chain before combining with the next hash until there is only one hash that represents the entire block (Bosamia and Patel 2018). Finally, this hash value will be aggregated with other variables and the previous block hash to apply the PoW. Zero-knowledge proof was used to disassociate the data from its owner. It is built assuming two entities, which are the verifier and the prover, the prover will provide witness evidence on the first stage which will force the verifier to draw questions from it before the prover sends these answers to confirm its transfer of assets without revealing the asset itself. Additionally, cryptography has found many use cases in this domain for certain model requirements such as access control in insurance and government approaches (Sun et al. 2020). The encryption scheme was used to provide secure communication between two communication terminals (Mahmood et al. 2018).

Many other techniques have been used such as aggregate signature, secret sharing and verified random function. Nevertheless, the community is focusing on optimization in terms of serialization requirements, signature verification, and expected PoW resources. Finally, it should be clear that cryptographic techniques are a mainstay of the blockchain system. It has an important role in providing authenticity. Additionally, secure a very difficult ledger mutability. This is the basis for a quick check of block content and its order using Merkle Tree. PoW uses the hash function as a random-seeking mechanism to promote resource-conditioned delusional fairness. However, the consensus layer has been widely discussed removing PoW or using these resources in a useful PoW.

2.3 Consensus

Consensus is at the heart of blockchain technology. Nevertheless, PoW and PoS have a more relaxed consensus mode. BFT and other voting-based approaches have more consensual criteria. State machine replication (SMR) in database studies aims to eliminate a single point of failure (Jha et al. 2019). Running on top of a distributed system, an expected atomic message is broadcast to update status or update and responds with another message. Thus, adopting the above consensus aims to ensure system execution even in the event of node misbehavior, node failure, or network latency (Haeberlen et al. 2007, Vukolić 2015). Deterministic state machine and consensus algorithms are the two requirements to maintain validity of consensus convergence. Message broadcasting can be over channels that support synchronous, asynchronous, or simi-synchronous modes. The atomic message must be contained in a protocol that maintains the following properties: first, validity, which is the guarantee that if a message is broadcast in the network, it will be included by the protocol. Second, the agreement guarantees that if a valid message is received by a valid node, it will be delivered to all valid nodes. Third, integrity is about ensuring that a broadcast message must be generated by a valid node. Fourth, total ordering is about ensuring agreement on the global ordering of messages (Chaudhry and Yousaf 2018).

The paper by (Fischer et al. 1985) discusses the possibility of the existence of a case where each consensus algorithm may never reach termination. It starts by defining the scope of the applicability of the problem, especially in distributed databases. Later, the article jumps into the definition of the concept that surrounds the model, which will be the basis for inheriting the different observations. First, a process has an input binary register 0,1 and an output binary register b,0,1. 0 or 1 are decision status while b means undecidable. Every decision is final in a race. Buffer is a middleware between different nodes to broadcast messages and each node has two operations, which are sending and receiving. The assumption is made that if the receive is run indefinitely, all messages will be delivered. A configuration is the internal state of processes and the buffer. However, the buffer will be null in the initial configuration. A step is the application of the transition function on a configuration to generate another one. The event is the receipt of a message, and the calendar is a finite or infinite sequence of events.

Three lemmas have been defined, the second introduces a concept that will be further exploited, and the first helps to justify the possibility of unlimited existence of the concept introduced in the second. First, the first lemma states that the transition function enjoys an associative property over the different places of the automata. The second state in which the bivalent initial configuration can be generated by the previous 0-valent initial configuration. It has been proved by contradiction by providing the state in which two different values can exist as adjacent in the chain of execution. However, running on the configuration in the case of a process failure of P which has a different view can generate bivalent results. Third, this concept will be explored by inference in which if a bivalent has been generated, then, through the accessibility of the graph, the possibility of concluding that an average decision state may exist that may lead to a bivalent value continues to arise infinitely. The article ended with a recommendation on how to evade such a problem by providing what appears to be a checkpoint algorithm with a light view change

CAP theory is consistency, availability, and partial tolerance. It claims that those three properties cannot be happening together within a distributed system. Partial tolerance is always the concept to be ensured by the developer in the first stage and then choose between availability and consistency. If availability is chosen by the system, then client will enjoy responsiveness. If consistency is chosen, then the client will enjoy rapid finality. Simply put, distributed computers require consensus, which makes it difficult for all versions to be consistent. If you offer client responsiveness from servers, the client's prompt request may occur before the server agrees. If you are offering atomic client verification for finality, removing availability is an option until you reach consensus between the servers. However, partial tolerance cannot be suppressed in a distributed system because high packet suppression can also cause low consistency.

At the algorithmic level, the distributed protocol must hold the delta delay time assumption. Moreover, it must also succeed in preserving two of the three properties of the FLP. The three properties are: safety is defined by forcing nodes to act according to protocol rules for the same atomic broadcast. Liveliness is that all non-faulty participating nodes terminate. Fault tolerance is the ability to show the resilience of the network to the failure of certain nodes. Fault tolerance algorithms can fall into two categories, which are either BFT, a term coined by Lamport (Lamport 2001) or crash failure models (Kondo et al. 2010). The last category deals with node failure in the case of unresponsive behavior. Thus, the system uses the notion of views or epochs, in which a leader is selected to make a decision on certain atomic messages. However, if the leader line crashes, a new leader line will be selected in a new view.

ZooKeeper (Hunt et al. 2010) and Paxos (Lamport 2001) are the best examples of these algorithms with the best performance that can reach $f<n/2$ (n represents the number of nodes and f the number of faulty nodes). BFTs take place when certain nodes behave arbitrarily or maliciously. However, it uses the same view concept for leader election. Generally, the algorithm can achieve $f < n/3$. The first successful proposal is the practical BFT (Castro and Liskov 2002). It was widely adopted and then followed by many algorithms. In the era of the open context of distributed implementation, more relaxed approaches such as PoW and PoS take precedence due to the expected anonymity of users and validators.

2.4 Proof of Work

The basis of PoW is the hash function. The intention is to invest in the unique chaotic result based on random search to be found in a limited space of time. The desired property of avalanche effects, which ensures that every slight change in the input will generate a completely different result, is the basis of the design decision to ensure equitable work between different parties. The search will take longer in time because constraints are imposed on the output and then the input will be manipulated with different numbers. Many nodes will search for a result for a hash generated from cancatenated strings extracted from different transactions. Once found, the output will be broadcast among many nodes. Each node in the system will store the results generated by the other nodes in a tree-like data structure. However, if a node has not generated the expected requirements on the result, it will accept the first output received.

Nodes tend to compete and keep their version of the output sequence until other nodes have preceded them with many instances (normally seven). Many versions of PoW have been proposed, in which a memory or CPU bound approach has been proposed. The first bitcoin proposal(Nakamoto 2008) was based on HashCash PoW (Back et al. 2002), which was used to discourage spammers. However, it has also been used to demonstrate the appendence to authenticate multiple user access to the same documents. CryptoNight (Tuzi 2018) is a memory bound approach. It is based on numerous reads and writes from memory to generate a result. The algorithm has three steps. First, it will generate scratchpad memory (Banakar et al. 2002) which will be manipulated on a generated SHA3 Keccak (Bertoni et al. 2013) of 200 bytes. An encryption sequence will be executed until the memory is full. Second, a loop function will be called passing the memory scratchpad with two integers. The integers are initiated by the first 32 bytes and the following ones up to the 64 bytes applied by an XOR function. A series of XOR encryption, addition and multiplication will be applied until nearly 2 million random reads and writes are performed. Finally, another encryption cycle will be done via the usesafe of AES-256 applied to the bytes [64..191] before the final generation of the hash.

Ethash is another flavor of the same approach based on instructions modified on the philosophy of the Dagger-Hashimoto algorithm. First, the seeds are generated by calculating the SHA3-512 hash value of the previous seeds. Second, a lightweight cache will be generated and populated in sequential order by SHA-512 before applying the RandMemoHash algorithm (Lerner 2014). Third, the DAG, which is a memory slice that forces the size to grow as the ledger leads to ASIC resiliency. Finally, the mining loop is 64 iterations. Each iteration contains an extraction of 128 bytes of the DAG to be mixed with other results from the previous one with reading of the memories

2.4.1 Criticism Proof of work

PoW critics come from different perspectives, it can be seen from security, performance or liveliness angles. On safety, many studies have been educated to show that coordination between miners after the election is a matter of rational decision to maintain a rhythm in the race (Lewenberg et al. 2015, Dey 2018). Probabilistic finality is the behavioral state of a system that occurs due to unintentional cooperation. The system nodes prefer to work under a mining pool. Selfish behavior introduced at the atomic level will take on different complex forms at the cartel level. A mining pool is normally managed by a coordinator who distributes the rewards among the different participants. Many studies have addressed the distribution of income between pool members (Salimitari et al. 2017), the type of communication (Göbel et al. 2016), a studies to build security (Lewenberg et al. 2015) or the distribution of maintenance within the pool itself.

The selfish mining intent is to make honest miners waste their resources on meaningless work (Sapirshtein et al. 2016). The selfish attitude is to not broadcast the founded block and keep working above it to always guarantee the longest chain compared to other pools. Eyal et al (Eyal and Sirer 2014) discussed the selfish mining approach raising the fact that majority is not enough in the bitcoin system before introducing a split revenue approach. Block withholding (Bag et al. 2016) has been deeply discussed, in which a miner can invest in the computer hash splitting protocol to mine with another pool in an attempt to sabotage it. Lie in wait (Haghighat and Shajari 2019) is a purely selfish attitude by which the miner will hide his block found in a pool search aiming to obtain the highest reward from the coordinator. Pool hopping (Belotti et al. 2018) consists of placing an impostor in the communication channels of other pools. Thus, the generation of tasks by the coordinator will be spied before building on it an expectation of directions. Another conceptual problem that underlies technical choices is centralization, which results from the longest chain rule with applied competitiveness.

The growth in the number of concurrent nodes will lead to a high level of forking and if combined with a high number of difficulty, it will make reaching the finality at a dramatic time. The lack of penalty towards malicious behavior is a disadvantage compared to other approaches. Additionally, the tragedy of the commons, which is that validators focus on selfish gain instead of user satisfaction, can be exploited by many tricks such as whale transactions (Liao and Katz 2017). Whale transaction is a transaction with higher fees to entice another miner to aim for the fork. At the level of communication, the solution suffers from the lack of incentive to cooperate. Thus, many protocol variants that have been implemented have focused on high propagation time and rarely on discussing the rewards of cooperations. The resource consumption of PoW must lead to a lot of research either to make it useful or to find a compromise between resource expectations and other security criteria. Many approaches have invested in the timer such as Proof of elapsed time (Chen et al. 2017), which invests in the Intel processor by randomizing the generation of the waiting time. Space Sharing Proof is another solution that use computational resources and focuses on space sharing. However, it is still very expensive.

2.5 Proof of Stake

PoS is a solution to remedy the high resource consumption of PoW. The incentive is that stakeholders are not interested in destroying their own funds. Thus, generating a random choice among these members to be a leader of the next validation session is a safe consequence of the first hypothesis. Many proposals have been developed under this philosophy, in which either a random choice among stakeholders is generated to guarantee the competence to be completed, a PoW-based race before the competence to be completed, or a community with a random generator of the next session. The logic to finalise by communication uses the BFT or competence on gossip. The proposal by Bentov et al. (Bentov et al. 2016) was the first complete concrete algorithm to implement the philosophy. It starts with a race to generate a hash for an empty block header. Second, it will be spread among validators who will derive from this hash N pseudo-random stakeholders by calling follow the Satoshi algorithm (Wang et al. 2020), which runs by finding the seed of initiation of these coins and tracing them to find the last holder. Third, other stakeholders will check the validity of the hash and generate a new list of validators, in which the validators will encapsulate as many transactions as possible in a data structure and sign it in sequential order. Fourth, the Nth member of the committee will submit the final packaging and move on to the next session.

Algorand (Gilad et al. 2017) is a proposal based on the use of BFT by which it contrasts Autoboros (Kiayias et al. 2017), which uses the Follow the Satoshi algorithm claiming that the use of a timer will reduce the possibility of malbehavior by participants. However, in Algorand, balance is the basis of selection by passing a seed to the verifiable random function (VRF) (Dodis and Yampolskiy 2005). The VRF will be responsible for generating community members. Casper (Buterin and Griffith 2017) is another proposal that uses BFT to finalize a checkpoint, how Casper's main attribute takes BFT from a closed environment to work in the open context. Finality on different checkpoints was the basis for ensuring accountability and dynamics within the system. Tendermint (Buchman 2016) is just a random PoS before finalizing it using BFT, it uses BFT spinning to handle the throughput. Degraded PoS, uses the same selection ideology through empty block hash generation, which will be followed by a random community vote on a leader, and then finalized by signing the transaction wrapper.

2.5.1 Criticism of Proof of stake

PoS also suffers in different dimensions of the theory. Logically, the choices to base the selection on the value of the stake make monopoly a normal habit of the system because normally few nodes will seek an infinite reward. Moreover, the selection is subject to many conflicts, which lengthens the probabilistic finality. On the communication side, mining cartels are an obvious trend among the highest stakeholders in order to get the reward in their direction. The inherited randomness of PoS makes it susceptible to manipulation by the randomly selected node withholding previous work. Additionally, the mining cartel can undermine any generation alternative to its own to secure the ledger monopoly. Forking is not an easy task to overcome even with the use of BFT due to each node's selfish interest in gain. The checkpoint introduced in Casper to finalize a sequence of blocks is not highly secure with respect to the number of participants. Moreover, asynchrony with a high delay can be very distributive for the stability of the system. In the work (AlMallohi et al. 2019), a selection of multivariate checkpoints was proposed, in which the block, active users and stake were entered for the algorithm. However, the security discussion was declared to be future work.

The stake bleeding was discussed in (Gaži et al. 2018) and explained that it is quite impossible to exist in its first introduction, before introducing another form named stake bleed based on Eclipse. The work (Zhang and Lee 2019b) have focused more on inducement as a driver to motivate honest behavior. The work ended with the definition of several barriers to the proper functioning of the system. Delegation in delegated PoS adopts its own techniques of delegated BFT (He et al. 2018). It imposes a higher level of monopoly among major coin holders. Nevertheless, the idea may find philosophical functioning in a certain use case but its acceptability in an open context is not welcome. Lee and Kim (Lee and Kim 2020) stated that the 51% POS attack is contrary to what was believed to be beneficial using short selling tactics. The Reorg attack is a form of using the Tezos PoS protocol. As the attacker owns 40% of the stake, the intention is to pass 20 reorg blocks with one per day in order to double-spend (Pantano et al. 2002).

The implementation of PoS at the communication level finds an incentive to cooperate but the coupling of communication with the logic of finality makes the different PoS proposals have many philosophical decisions that expose security on many dimensions. PoS has been extensively coupled with delaying or BFT approaches to produce probabilistic finality that is hardly to be changed. However, the open context remains a major obstacle for the PoS ideology to overcome it. Competitive dynamics have not been an option due to the expected credibility attached to each mining node.

2.6 Proof of Useful Work

Many other approaches have attempted to twist the underlying concept over which PoW operates. Some approaches have focused on delay as a core mechanism, such as proof of elapsed time (PoeT). It used the Intel hardware processor as a random delayer before focusing on convergence on one version of the ledger. Proof of space (PoSp) focuses efforts considering that the cost of malicious activities is the most important concept. So it used partitioning disk space as another way to run many iteration cycles. Proof of useful work (PoUW) tries to make use of spent resources in a task that can later be useful for another domain. In Coin.AI proposed by baldominos and saez (Baldominos and Saez 2019), the goal is to train a deep learning model to identify the most active nodes in the system. It concludes based on rules defined from a hash value a deep learning model. The function is based on a context free grammar G=(V,R,S) where S represents symbols, V non-terminal symbols and R a set of production rules. Chen et al's work in (Chen et al. 2018) pointed out that the trade-off within each transaction in the blockchain is the most significant drawback. So they tried to fit an Alex network to a capacity assessment system. The nature of the network, the security and the independence of the variables are the properties to generate a trained matrix.

Lihu et al (Haouari et al. 2022), proposed the first PoUW protocol by which they explained all the working steps. However, the proposal is aimed at the machine learning community. The basic idea is to offer coins to miners in exchange for tasks submitted to them by clients. The three stages of validation reveal the data to miners, allowing the miner to work on them until a certain requirement is met before the test data is revealed to the evaluator to choose the winner. Panagiotakos and Russell proposed the use of a specially designed local search algorithm to achieve consensus named doubly parallel local search (DPLS). Proof of search (PoSe) relies on the definition of an exchange protocol between the miner and the clients whereby the clients provide a list of solutions and puzzles to the user to search for the approximate solution. GridCoin(Halford 2014) is another search proof scheme, is a reward-based system implemented to solve a search on the Berkeley Open Infrastructure for Networked Computing (BOINC). Proof of Burn (PoB) is a protocol that provides community voting on the use of dropping coins in exchange for validation before getting more reward. The work of Chen et al (Chen et al. 2018), aims to categorize nodes into super, random and unknown nodes. The use of thresholds for categorization subjects the system to static standards applied to the characteristics. Thus, we make the choice to lack dynamism and to be subject to a monopoly based on the sophistication of the environment.

2.6.1 Criticism of proof of Useful Work

All approaches that have attempted to isolate a concept within the PoW to invest in it as a security mechanism have failed due to the trade-offs applied on the various participant-related attributes. The delaying approach such as PoeT suffers from the non-existence of the global clock, the ease of regenerating another version of the truth, the low cost of malicious activity. Although it was proposed to operate in a highly trusted environment, the approach did not address many concepts related to consistency and how it will be enforced. PoSp has invested in the cost of malicious activity by making it difficult to generate a block and therefore alter the truth. However, memory is very expensive in addition to dedicated computation and time to confidently commit a transaction. The work of Baldominos and Saez (Baldominos and Saez 2019) is conditioned by the static model of deep learning in the case of decision-making, which does not meet the need for dynamism required by consensus. Moreover, it showed the inability to cover the open context of miners. The problems of the PoUW protocol are the lack of understanding of user needs, consensus requirements are not met, and finally some proposals do not meet the market as impractical. However, Bitcoin-NG requires some synchronization, which will expose the system to a DoS attack and face issues such as accuracy, latency, and targeting by undermining the leader.

2.7 IOTA Tangle

IOTA Tangle is a proposal that was shared in a whitepaper in (Popov 2016) by Popov. The goal of the solution is to increase validation time, reduce fee, and provide a backend for a solution that can handle the high level of submission with the IoT industry. Directed Acyclic Graph (DAG) is the form of data structure which will contain different transactions like leaf and nodes. Each transaction is a site and the final transaction is a tip until cleared. However, each newly submitted transaction will be responsible for selecting two previously submitted unvalidated transactions to validate before adding. A transaction is first validated by checking the balance before internally executing a small PoW before being added. However, the weight represents an accumulation of power to be the link between two transactions, as the graph grows the selection takes many directions. However, a transaction is committed as long as it aggregates a high weight. The Genesis transaction contains all the coins to be used later in the system. As the graph grows, the search for related information increases dramatically. Many random selection search algorithms have been proposed to work to select two transactions or three

TangleCV (Rathore et al. 2020) is another proposal based on the IOTA tangle. It aims to optimize efficiency and scalability over information accuracy and integrity. The solution relies on finding a way to secure the use of public key infrastructure. The work (Lathif et al. 2018) proposed a configurable and interactive distributed ledger simulation framework (CIDDS) based on DAG, which is a simulation that can run thousands of nodes and study different characteristics of the network. G-IOTA (Bu et al. 2019) is another selection algorithm used to overcome left-behind tips. The algorithm modifies the random selection before increasing the validation of the selected transaction from two to three.

2.7.1 Criticism of IOTA Tangle

The IOTA approach has been widely criticized in terms of technical choices, network vulnerability and logical inconsistency due to architectural choices. Curl-P is the hash to sign a transaction or generate a hash proof. However, it was found in (Heilman et al. 2019) that the function has many vulnerabilities, in which cryptanalysis showed many hash collisions. It can be exploited in the system by asking the victim to sign a bundle before swapping it with other addresses. A bundle is a set of input and output transactions. The replay attack (De Roode et al. 2018) is another method to resubmit the same transaction until it exhausts its resources. The attack can be executed by the initiator sending a transaction to validate without indicating another address to submit the remaining fund of the coins. Thus, the attacker takes the opportunity to keep replaying the same transaction until it empties the initiator's fund.

A proposal is made by Bu et al. (Bu et al. 2020) to introduce fairness by modifying a search algorithm that thwarts splitting attacks. The adoption of the IOTA solution has been widely discussed among the IoT communities. However, it has been clearly stated that the solution suffers from high duplication which necessitates the need for an expensive search algorithm. Moreover, using a light PoW on the user side does not remove the fee from the system but delegates it to the client side. IoT devices might not be able to handle random searches in some cases. IOTA studies have many missing concepts to analyze, such as consensus in its logical form, finality and consistency. Additionally, the platform's lack of incentive to motivate different maintainers to participate is crucial to its effort. Thus, using a simple distributed system can collectively provide the same conceptual attribute of IOTA without the random search load.

2.8 Byzantine fault tolerence

Lamport et al. in (Lamport et al. 2019) introduced the general Byzantine problem as a substantive philosophical problem to describe the state of distributed consensus among different processes. The goal is to come to an agreement on the order of an event. Process of studying collision failure protocol in terms of normal failure, while BFT addresses the issue of random behavior of nodes. However, Byzantine state machine replication is the closest system requirement for adopting BFT to blockchain technology. The BFT algorithm is divided into three stages, which are the consensus protocol, the change of view protocol, and the checkpoint protocol (Xiao et al. 2019). First, the consensus protocol is to provide initial consistency on the order of events. Second, the View Change Protocol is to vote on a new leader in case of misconduct or leader crash. Third, checkpoint protocol is about finalizing previously agreed-upon events and ensuring there are no conflicts. Thus, agreements, termination and validity are the stabilizing concepts to be achieved from the algorithm. First, an agreement consists of declaring that all non-failing nodes have agreed on the same value. Termination means that there will eventually be a decision among non-defective nodes. Validity means if the algorithm fulfills the termination and the agreement can be extended to deal with Byzantine behavior (Wierman and Tastle 2005).

By extending the SMR criteria, the algorithm must also deal with security and liveness. Equivocation is the ability of a node to send inconsistent messages to different peers (Li et al. 2016). This is a problem in BFT approaches due to the ability to delay the decision. Thus, the delivery time assumption is very important in this approach. Two different categories of thinking are either Quarum-based approaches or agreement approaches (Naor et al. 2019, Lao et al. 2020). Paxos was the first solution to fault tolerance with the assumption of intersected quorums. However, Practical BFT (PBFT) was the first proposal to agree on an approach by extending Paxos (Lamport 2001) to crash faults. It secures normal operations in a partial synchronization mode but with very high message complexity. The system changes leader via a step called view, each view starts by selecting a new leader. The view operates by running the main consensus algorithm, which results in the exchange of three types of messages. Consensus starts with initiating a request from the customer. Second, the leader will assign values and send a pre-prepare message to all replicas. Third, on reception, the different replicas acknowledge the command and broadcast a Prepare-Message to synchronize the decision. Fourth, upon receipt of 2f+1 messages by a node, it will broadcast the commit message, where 3f+1 represents the number of replicas and f represents the faulty nodes. Finally, upon receiving 2f+1 prepare messages, the replica accepts the new value and adds it to its log.

The Query/Update (Q/U) protocol (Abd-El-Malek et al. 2005, Lamport 2001) aims to provide better throughput and fault scalability. Each quorum is responsible for the complete processing of a transaction. However, the approach has a lower tolerance to Byzantine behavior, where 5f+1 represents the number of nodes. The client contacts their preferred quorum and since each replica view is consistent with the client's request, it will be updated directly. The client will receive 4f+1 responses that indicate completion. Zyzzyva (Kotla et al. 2007) is a speculative approach above the PBFT. The approach assumes that most replicas are safe, which leads to abandoning the overuse of expensive commits. As the client sends a request. Upon receipt of a replica request, the replica will send a response. The receipt by the client of the 3f+1 responses will be followed by the initiation, N of the commits request. In the case of receiving between 2f+1 and 3f responses, another non-speculative BFT algorithm will be executed. The leader-based approach is due to the expected high trust transferred to leaders. Thus, another approach called for taking the responsibility of the leader. Proposed weak leader in the Democratic BFT (D BFT) (Bonniot et al. 2020). The weak characteristic is based on the idea that a leader does not impose its selected values but simply coordinates between replicas.

Circumventing the FLP impossibility, which states that it is impossible to reach consensus within an asynchronous network, is through the use of random BFT (Zhou et al. 2017). RBFT is all about making a quick follow-up random decision on the leader and eliminating the opponent's focus on an expected leader. HoneyBadgerBFT (Miller et al. 2016) uses a new scattering algorithm to achieve optimal asymptotic efficiency. Additionally, the use of encryption to ensure freedom from censorship. It is made up of two main components, which are the Asynchronous Common Subset (ACS) and the Threshold encryption. A selected node transaction will first be encrypted according to the threshold scheme before passing it to the ACS module. ACS will serve as a consensus layer divided into Reliable Broadcast (RB) and Binary Agreement (BA). Another important proposal in the field of blockchain technology is Hotstuff of Malkhi (Yin et al. 2019), which is an approach to optimizing throughput with a focus on responsiveness. The key criterion is the collection of (n-f) quarum certificates. It has three stages, which are preparation, pre-commit and commit.

2.8.1 Criticism of Byzantine fault tolerence

Malki et al. (Malkhi et al. 2019) introduced the flexible BFT which develops a dynamic quorum and addresses the issue of living but corrupt members. They used adaptive quorum as a way to fend off any malicious nodes. However, BFT suffers from a high level of message complexity, which makes it impossible to adopt it in an open context of a distributed operating environment. In addition, the different stages of the messages lead to many series of strategic attacks to delay the system. Conceptually, the vulnerability lies in the leader-based approach, which leads to targeted malicious behavior and the manipulation of elections or holding positions to monopolize the system. Central decision-making power makes consensus suitable for a private blockchain. Although many proposals have been generated in the literature to speed up transaction processing in terms of throughput and Goodput, the scalability in number of nodes is still a big challenge for the technology because the need for global synchronization which requires the vision of absolute truth is nearly impossible to handle

2.9 Functions over data

The decision function to add a data structure to the database may have a different conceptual contribution to the system. The ledger can take many forms, which can be a linear, tree, or directed acyclic graph (DAG). The linear ledger leads to a competition between participants on the longest chain, so that each participant, in the case of the same resources, has an equal probability of having a block added. The tree ledger was first introduced in the Ghost protocol to increase the number of validations. it found a solution to injecting orphan blocks. However, it was later dropped due to security requirements. DAG was used for IOTA Tangle to enable parallel transaction processing. Although the system is significantly improved in terms of validation, but on the price of memory. Erasure-coded BFT is proposed by Hendrick in (Hendricks 2009), in which he uses encryption of erasure codes. It generates an M to N erasure code. A block will be encoded into N fragments and each M corrects a fragment. It was later adopted in many approaches such as PASIS(Wylie et al. 2000) and AVID (Hendricks et al. 2007). It is good ideology to increase privacy. Zeno is a proposed feature for BFT state machine replication(Singh et al. 2009). Semantics is at the center of algorithms such as linearizability. It focuses on availability and provides weak consistency.

Spinning is a technical choice made at the system level to preserve performance. It was proposed by Veronese in (Veronese et al. 2009) to address a certain type of leader delay and attack discussed in Prime (Amir et al. 2010). Zzyzx introduced a Byzantine locking scheme to allow the client to extract state from an underlying replicated state machine. The fork can be soft or hard. The soft fork is still in its infancy and may be modified in the near future. However, the hard fork cannot be changed due to the division of miners over the reward of conflicting transactions. Thus, the finality is based on the number of blocks added above a certain block containing a transaction. Bitcoin developers have been offered the use of Github's version registry as a benchmark of truth. However, the idea was dropped due to user perception on centrality. The different choices are all a set of trade-offs to manage the extraction of conceptual stability.

2.9.1 Criticism of functions

Misconceptions have circulated around the technology to promote the sector for financial reasons. Immutability as a strong concept cannot be ensured in technology and it has been explained by the normal exposure of competition among miners. The correlation between the difficulty number and the number of miners is negative. Moreover, the monopoly state is a normal case of existence due to the use of means on the rule of the longest chain (Nacer et al. 2020). Thus, the control of the network is decentralized between the owners of the means. Cybersecurity issues can undermine the existence of honest users due to the lack of understanding of the world that has been exploited by malicious users or miners. Finally, the reward for high skill can cause the system to be monopolized or undermined.

2.10 Networking

The Peer to Peer (P2P) network is the platform over which blocks are exchanged between nodes. The dissemination of information can be done in the form of a broadcast, multicast or unicast mechanism. However, the policy applied above the broadcast type may be through the use of complex multi-hop routing (Vinodhini and Gomathy 2020) or flooding. Flooding is the technique used in the blockchain system. Another important influencer on the rapid diffusion of transactions is the organization of nodes. Blockchain open network uses flat random graph topology (Dotan et al. 2020) and private network uses star topology between important nodes (Spasovski and Eklund 2017). Propagating transactions is the first step after initiating the client to its first collection of peers. Each user has their own peers as a view of the truth, their first transaction submission will be replayed in gossip approach to all other nodes. It has been noted that 25% will receive the transaction within 17 minutes on the early stages (Neudecker 2019). It has been optimized to be 50% for 15s. Once the transaction is successfully propagated, another step takes place, which propagates the block. The block will be replayed under a certain protocol due to the size which can have a huge impact on bandwidth.

Protocols have been proposed based on the fact that efficient block propagation is the key to accelerating probabilistic finality, reducing the number of forks, and meeting security measures. Compressed block encoding tries to use encryption techniques to deliver the block to reduce the load on the bandwidth. It also claims that most transactions have already been propagated to most nodes. The approach has two variants. The first is dedicated to high-bandwidth and the second to low bandwidth. The low-bandwidth variant begins by broadcasting the hash found by a winning node. At the reception, it will inform the neighbors by inviting them to retrieve the data. Neighbors will respond to the get data function which contains a request to submit knowledge about this new block. The node will submit a list of transaction IDs to neighbors. The neighbors will respond with the necessary transactions and the node will eventually broadcast the hash and the necessary transactions. In the case of high bandwidth, the protocol will suppress the invite and request from neighbors messages after it.

Thus, few proposals have been implemented to address efficiency through the use of cryptography. However, they all kept the same ideology of inviting followed by requesting the necessary transactions in different encrypted forms before the final block request stage, such as Xtreme Thinblocks Propagation and Graphene (Dotan et al. 2021). At the topological level, the node must build the table of neighbors. In the case of the bitcoin implementation, a node will either request a list of peers from other nodes or use a well-known DNS to retrieve seeds. IOTA takes further options by selecting a previously known node to share the peer list. An important direction associated with a strong association with networking choices is off-chain sharing and payment which requires an exchange scheme between different nodes in addition to many data replication considerations with many servers.

2.10.1 Criticism of blockchain networking

The efficiency of miners in case of conflicting transactions can delay the increase in the number of forks and in case of propagation time, it will delay the probabilistic finality. Thus, many proposals have been generated in the literatures to meet this criterion by emphasizing responsiveness such as the Replay network and FIBER. However, the solution cannot fill the gap because it provides selected nodes with the ability to organize miners, which will lead to manipulation on the selection of transactions or mining benefits. Intent delay of miners can be demonstrated by performing a DDOS attack (Rodrigues et al. 2017). The opposing user will generate many transactions and take advantage of the random gossip approach used for transaction propagation. The attacker aims to fill the bandwidth with unnecessary transactions to delay the propagation.

The eclipse attack in blockchain technology is the act of isolating a user from honest nodes (Dai et al. 2022). It will be done by balancing DNS servers or submitting a list of peers belonging to the same miner to a client. The attacker will submit two transactions, one attached to the victim's address to be extracted among the victim's peers to provide an illusory view of the truth, and the second will address most of the network with a different address. The eclipse can take another form, such as network splitting, to create two different forks that can be based on two different conflicting transactions. The man-in-middle (MIM) attack is another approach to gather network information to prepare for a DOS attack or a split attack (Kumar and Tripathi 2019). It has been demonstrated the impact of such an approach on the Ethereum network (Devi et al. 2021) to build the basis for double spending. Thus, the organization of the network is very important due to the anonymity of the nodes. Unlegalized trading within the platform makes it very difficult to seize funds without proper user identification. Thus, many techniques have been developed in the literature relating to the identification of the IP address of users.

Organization can have a huge impact on finality, system performance correlating with throughput. Thus, the mapping of the network has been studied and many libraries have been implemented. The goal is to try to identify the trade-offs between different random flat topologies, cryptographic techniques, propagation time and data size to provide efficient, fast, robust and reliable transaction processing. After understanding the disadvantages and performance of the network, many solutions have been proposed to either help hide the topology or force reputation into the network and undermine malicious activities such as eclipse. Protocols such as Xtreme Thinblocks Propagation and Graphene suffer from a high level of error when restoring. So implementing the approach to operate on a global stage can have a huge impact on forking. Finally, propagating the transaction over a globally dispersed topology considering a large number of participating nodes for a reward, forces the system to engage in selfish miner behavior to withhold transactions. Therefore, scalability is a very difficult concept to achieve.

2.11 Data Structure

Blockchain as presented in the Bitcoin paper (Nakamoto 2008) aims to secure a tamper-proof and tamper-resistant ledger. Centralization of data management via banks or government is aimed be distributed. However, an open context for distribution leads to the question of whether it is permissionless or permissioned participation. The permissionless network cannot guarantee security due to the ability to hog the ledger at any time in the case of low number of miners. Transparency as a concept may be the goal of only stakeholders in some cases, which makes the permissioned network more appropriate. The network is based on exchange transactions, which will be wrapped in blocks. It has two types of users, a simple user and a miner. A user submits a transaction to authenticate and validate the exchange of a good. A property is data that represents physical or virtual entities. Transactions will be wrapped by a special node to be mined. A transaction will be authenticated by a signature using a hash function. The hash function will take the variables attached to the transaction as a string to digest a hash value. The function must secure three properties, which are pre-image resistance, second-image resistance, and collision resistance. The transferred property will be pegged in terms of UTXO model or balance-based model.

The UTXO represents the values managed in a certain blockchain system. It will be aggregated into an input and an output attached to a transaction before being added to the general ledger. The output consists of newly generated objects which can be a reward for the miner or represent the transferred value. However, the input represents a copy of the output in another transaction which must be associated with the same identity. Thus, the sender will always be associated with the input coins. However, the output coins will be associated with the block's receiver or validator (Delgado-Segura et al. 2018). The input value must be greater than the output value, which forces another transaction to be a value returned to the initiator. The fees that apply to each transaction are also associated with the validator. A list of transactions is injected into the block. The transactions along with other associated information will be responsible for generating the hash value. However, the coins will be of no use except to waste a large amount of memory, which leads to many other questions regarding the right to be forgotten. Merkle Tree can help, but the long linear register is highly questionable. Research within the Bitcoin platform is developed as a knowledge base that duplicates all coins classified according to the identity of the owner. Thus, the size is an issue because the heaviest part of the ledger, namely the coins, is duplicated. Additionally, exchanging micropayments can increase memory size.

The Balance-Approach has been widely used in banking as a simple status update system. The adoption of this model within blockchain technology was first modeled by the Ethereum project. The data structure contains sender and receiver information in addition to the updated balance. A replay attack (Pries et al. 2008) is a way to take advantage of the simplicity of the balance model which can be reduced to a simple manipulation of numbers. The nonce will be the sequential number that will eliminate duplication of the transaction. Moreover, strict conditions will be put on the number of nonce, which leads to the elimination of any parallelism. The balance model suffers from a single entry point because balances are accessed sequentially. TheChain (Nacer et al. 2020) used two models in one, which is the UTXO and the balance variable. However, the balance is just to speed up validation. Thus, it can benefit from privacy and parallelism in addition to the easy implementation of contracts above the balance model.

A user does need a software called a wallet to store keys. The wallet manages the user of the system as a central actor within the platform. It can initiate transactions and manage the fund. The wallet contains information on the peers to connect within the platform. Due to the unorganised random connection among peers, many kinds of attacks or theft can appear. The web-based management brings a third party to the process of validation; however, the user may trade this for the benefit of the security provided by a different server, such as a firewall. NiceHash is a wallet that was hacked, leading to the exposure of many keys, the result of which is that $63 million was reported stolen (YUEN Man-Ching et al. 2020). The work in (Koblitz et al. 2000) discussed the leakage of keys by taking advantage of the elliptic curve cryptographic (ECDSA). It discussed how many attackers have taken advantage of this vulnerability. Moreover, the user wallet can be subject to direct attacks due to direct connection with anonymous peers that can profile the device used.

Blocks are another important component, which contains set of transactions, block sequential number, hash value of previous block, timestamp, size and the two most important variables, which are nonce and hash representation of the block using Merkel Tree (Bosamia and Patel 2018). The Hash representation will be generated before being aggregated with an incrementing nonce number to generate a new hash that meets the criteria. A smart contract is another data structure that can be attached to a block to be executed periodically. Nick Szabo defined the concept, which aims to virtualize the contractual relationship to ensure complete digital transformation. The Ethereum Foundation was the first to adopt the concept within blockchain technology. The contract can handle different conceptual properties and is not limited to values. However, the deterministic behavior of the contract always generates a transaction which will be managed like another but with an additional verification criterion which is the contract itself.

2.12 Blockchain Platform

Bitcoin was the first platform implemented to serve as a complete blockchain system. It was followed by many approaches built on it such as Ethereum (Dannen 2017), ZeroCoin (Miers et al. 2013) or IOTA with different advantages such as the use of the contract for an automated continuous relationship, the disconnection between data and the user and the parallel processing of transactions. However, on the private network, BFT as a consensus approach has been the main solution to enable fast synchronization between different machine states and provide customer responsiveness. The Hyperledger (Androulaki et al. 2018) structure has been implemented with many versions of BFT such as BFT-smart or PBFT. However, the architectural choices have an impact on the conceptual functioning which leads to questions about the adoption of the technology within the different industries. Distribution is the nature of technology with the aim of ensuring transparency between participants, whether they are simple users or maintainers. Persistence is another important concept because linear technology uses RAM for a block sequence before saving it to disk after being processed. The DAG in the IOTA tangle primarily relies on memory access to link transactions. The Hyperledger Fabric uses the world state as an external institution of validation and makes persistence a matter of time. Anonymity is a very interesting concept in the field of cryptocurrency. It attracts users because their storage of values cannot be linked to their real identity as a form of security without the need for a trusted outside institution. Auditability is the ability to trace the source of funds or information between different data structures. It allows stakeholders or government to want such technology to ensure the irreversibility of executed knowledge.

Financial services have sought to embed the virtue of blockchain technology within the industry. However, due to legal requirements such as a tax, Know Your Customer law, or verifiability, it will be very difficult to attract cryptocurrency users. The insurance claim is a case to be automated by blockchain technology due to the need for transparency between the user, the insurance company, the underwriter, the repair shop, and the police. Thus, the use of smart contracts is very useful to force control, transparency, and traceability. Global commerce may also force smart contracts as a means of exchanging goods and services between different interbank and banks. The IoT sector with the field of smart cities requires the conceptual contribution of blockchain with the aim of ensuring access to authentic communication between the various components, negotiation, and owners. Decentralized Autonomous Organization (DAO) is the concept above which businesses are expected to operate by applying the smart contract as a medium of exchange. However, many transmitters are popping up as servers will be spread around the world and cross-border legislation is not well standardized. Additionally, the DAO and its creator are complete virtual entities that require authentication by a third party, making it difficult to implement and execute the due diligence process normally performed in the financial industry. The smart contract as an independent solution is attractive to many industries, but the generated transaction is subject to low system performance, lack of interoperability management, and inability to scale.

2.13 Summary and direction

It has been noted that the trade-off between different concepts within a distributed system is a normal decision in order to achieve overall consistency. Ways to force a version of the ledger can be problematic in discussing the concept of fairness and integrity. Resources as means is another efficiency issue. Finality is relevant to annexation rather than authenticity and duplication. Responsiveness is not addressed but refactored with asynchronous notification. Scalability in terms of nodes and data growth is very problematic. Additionally, different platforms suffer from a lack of modularity, interoperability considerations, and data structure agility. Propagation is subject to a single space of group of chained nodes to chat indefinitely, which subjects it to linear growth in terms of correlation between time and groups. The data structure suffers from duplication, a lack of means of traceability of coins and a single point of entry in the case of the balance model. Exchange methods suffer from exposure due to the use of trusted parties to exchange data. Defined incentives lack competitiveness and lead to cooperation to provide safe space of delusional distribution.

The approach proposed in this thesis questions the need for global consistency. It proposed a new sociology-inspired algorithm to enforce integrity through reputation management. The only branching needs for regional consistency frees up scalability in terms of nodes. In terms of data structure, growth was controlled using transaction links and fuzzy traceability. Transaction propagation is maintained with high performance due to special network organizations. The exhibition has been retained and confidentiality is increased. Incentives have been defined to force consistency.

3 TheChain

"Why is there a need for consensus where all participants can make a quick and correct decision"

3.1 Introduction

Cryptopolises (Swan 2018) is a world where the crypto citizen acts freely outside the bonds of the trusted authority. Blockchain has enabled cryptocurrency in real life, and it is the key to the world of cryptopolises. Blockchain is a data structure that is built upon a hashed function, then distributed among different nodes interested in its validity. The data structure is wrapped into a list of blocks linked together with sequential use of the hash function on the content, and each block uses the Merkle tree (Merkle 1980) to guarantee the order of transactions. Therefore, the ledger is immutable, and a minor change such as an order of two transactions requires readjusting all the hashed values. The received transactions are encapsulated into blocks and subjected to a consensus mechanism aiming to ensure that the entire network updates the distributed ledger with a valid transaction.

Primarily, the first implementation of the blockchain was a solution to process financial transactions without the participation of a trusted party. It was an integration of different techniques to secure a chain of blocks, and the first proposal for this type of chain was to guarantee the integrity of a document by keeping records of each access and providing a secure history. It was another approach to the digital safety-deposit box that suffers from a lack of privacy, bandwidth storage, incompetence, and trust (Haber and Stornetta 1990). Afterwards, optimisation is added to the next work by the usage of a Merkle tree (Bayer et al. 1993). Finally, comes the adoption of a consensual mechanism that can eliminate the sibling attacks by reusing the HashCash PoW in a race setting. Applying the same technique in the other field has the potential to facilitate trade, identity verification, secure diamond grading, tracking the shipment around the world, and cross-boundary payment without fees (Dillenberger et al. 2019). However, the technique suffers from scalability problems, because a search over the data structure is costly, and the consensus with a permissionless network such as Bitcoin is limited to seven transactions per second (Xiao et al. 2020).

This work aims to answer the question of why there is a need for consensus where all participants can make a quick and correct decision by taking advantage of the structure of the Petri net, leading to intersected regions of interest and increasing the importance of certain concepts within the network. The next is the basic motivation for this work. The third section is the related work on consensus within blockchain technology. The fourth section is devoted to data structure shared in the system. The fifth section is a description of the data flow on TheChain level. The sixth section is the introduction of the approach by discussing the validation layer and governance within the network. The seventh section compares our proposal with the works available in the literature and how it can show better performance, before finishing with a conclusion claiming the suitability for banking and micropayment for the Internet of Things.

3.2 Motivation

CAP theory states that consensus requirements force different nodes to make trade-offs between consistency, availability, and partial tolerance. It will take time to reach consensus as the system scales in terms of the number of nodes. Thus, the purpose attribute required from the user is conditioned by security. Security is ensured by the high rate of duplication of certain information between honest nodes. Thus, building a global machine must recognize that users are dispersed and their perception of reality at the physical level is as different as the logical level. Thus, the need for global consistency is not a requirement for the system to work.

3.3 Related work

The Bitcoin (Nakamoto 2008) proposal introduced the use of Hashcash (Back et al. 2002) to deter a participant who attempts to attack the security or the liveliness of the system. The goal was to make it virtually impossible for them to invest IT resources before dealing with a massive number of nodes interested in the validity of the ledger for their financial benefits. Several works have studied the Bitcoin proposal and its vulnerability, including

(Liao and Katz 2017, Eyal and Sirer 2014). However, computer resources are the only condition for having a better probability of winning the race by solving the problem of the NP complete puzzle box leading the consensus to rely on the honesty of 50% of the nodes (Kroll et al. 2013). This has led to the introduction of mining cartels and various selfish mining strategies related to it (Bonneau et al. 2015). However, the dishonest nodes will invest in the longest chain vulnerability to alter the global belief in which version of the data structure is valid (Jang and Lee 2020, Xiao et al. 2020).

PoS was a solution to solve the problem of computational resources, inheriting from the PoW its randomness by implementing the Follow the Satoshi Algorithm (Bentov et al. 2014). It comes from the incentive that stakeholders such as miners within the Bitcoin network are very interested in keeping the ledger valid. However, the idea leads to a monopoly exhibited by 50% of stake value (Xiao et al. 2020). Moreover, the 'nothing at stake' attack from a random node can coordinate a long-range attack by investing in the vulnerability of following the longest chain and building side one (Deirmentzoglou et al. 2019).

The discussion on the adoption of the BFT technique within the blockchain technology may open up a new possibility of solving it. Lamport first proposed the problem of how to make the different processes reach a consensus on the order of an event. Castro et al. (Gramoli 2020) proposed the Practical BFT that is considered the most widely used approach currently in the industry. Malkhi in (Malkhi et al. 2019) proposed the Flexible BFT and introduced the alive but corrupt attack, in which the attacker is interested in keeping the network alive but threatens its safety. Nevertheless, the epochs of messages that the community goes through with the elected leader have a high level of message complexity that makes it hard to implement for a permissionless blockchain.

The IOTA foundation proposed the use of a DAG by removing the concept of a block and allowing a different search algorithm to find the associated information on the graph and the transaction finality does depend on a cumulative weight rule (Xiao et al. 2020). However, the splitting attack is discussed and addressed in G-IOTA (Bu et al. 2019) by proposing a new search algorithm. Moreover, the approach claims the zero-fee transaction, whereas it implements Hashcash PoW within each participant transaction. Wang et al. (Wang et al. 2019a) proposed the use of ReRam, a non-volatile memory, and raised concerns about the computing resource, which can grow massively when DAG also grows.

The Petri net is a BI-Graph which has two different types of nodes. The network is constructed from a marking vector and two matrices, which are a Pre-Matrix that describes the outgoing value to the transition from the engaged places, and a Post-Matrix that describes the outgoing value from the transition to receivers' places. Also, the marking vector describes the different places with the number of the token included, and due to the network suitability for formal analyses within the real-time system, different work is built upon it to adjust it to particular use cases. The coloured Petri network is the technique of associating an identity to different values within the place. The object Petri network is an extension of the coloured network to give more formal descriptive implementation with more functionality such as abstraction and inheritance. Ramchandani in (Ramchandani 1973) proposed the timed Petri net, which is a time-oriented performance evaluation network that is defined with an association of firing duration linked to a transition. Mathematically the Petri network is modelled as follows:

$PN = (P,T,Pre,Post,M0)$

P : stands for a list of places

T : stands for a list of transactions

Pre : pre-transaction matrix

$Post$: [p×T],the post-transaction matrix

$M0$: the initiation of the marking vector

The calculation of the incidence matrix from the Pre-Matrix and Post-Matrix:

$C = post - pre$ (1)

The work in (Liu et al. 2019) discussed the different game theory analyses dedicated to the PoW; it concluded that the PoW is vulnerable to 50% attack and to various attacks which depend on the selection of the forks. Moreover, it can be subjected to latency due to the selfish behaviour of miners or pools. PoS suffers from various disadvantages such as monopoly, long-range attack, uncle's block and pool cartels (Nguyen et al. 2019). The IOTA approach suffers from centralisation and resource consumption that can grow massively (Wang et al. 2019a). Moreover, BFT suffers from a high message complexity toward the leader that makes it unsuitable for permissionless blockchain. Thus, all existing methods may suffer from either a heavy state transition, a resource consumption mechanism or vulnerability to attacks. According to the European Parliament, this technology has the potential to change the lives of many people. Consequently, this work tries to drop the consensus, aiming to look at the old problem from a different angle where intersected regions of interest are implemented by taking advantage of the Petri network structure and raise new possibilities of solving it.

3.4 Data Structure

The data structure on TheChain (Nacer et al. 2020) level will be addressing the general philosophy and data structure on the architectural level. Transactions will be kept in a sequence of related spending. Moreover, wallets are an ensemble of associated balances to provide fast reconstruction of the validation layer and tracking of each identity. It can also be as well the state of difference of the IoT data. However, in the validation layer, the manipulation of matrices will provide simplicity and fast verification. Thus, the incidence matrix will describe the transition by translating the transaction into a credit rule to an account debited from it. The transaction has reference variables possess the memory addresses of the next transaction and previous types of participants transaction to build a sequence of the linked related list. The search will be based on referencing to the first unused coins and it will be later described within TheCoin data structure. At the other level, linking blocks require counting the power of previous transactions to associated in regional space. It will allow later the construction of regions to be seen by maintainer as space of hot operations that lead to competition.

The transactions will be associated within the later stage to allow total order among maintainers on the memory reference layer. It will eliminate confusion in the case of not similar results. Figure 3.1 will demonstrate the construction of the block, which contains two main components. Wallets are the result of the interaction of different rules over the elements. It can describe the value of balance, the difference, or the state. The sequence of wallets will be aggregated into a vector named node and transaction will be aggregated into a matrix named transition. On the persistence and validation layers, the representation will be different in terms of details, as validation will be more abstract for operational reasons. A definition of the used data structure can be written as follows:

```
Class Walet:
  publicKey identity;
  double localSequentialNumber;
  Balance blnc;
Class Coin:
  String identifier
  double value
  publicKey sender
  publicKey receiver
```

```
$Class$ Transaction:
    $Received$ timestamp
    $reference$ sender
    $reference$ receiver
    $list < Coins >$ Values
```

At the general level of the platform. This is a simplification before giving more details. The wallet class contains an identifier, a local sequential number which gives a precise number to the number of transactions applied by this identity and a balance. The class coin contains a unique identifier attached to it alongside its value, the last sender and the last receiver. A transaction is constructed from a coins list, timestamp, and memory references for management purposes. The block shown in Figure 3.1 will not gossip into the network in this form, but will be the result of encapsulating received transactions and generating hash values from them. However, when a peer requests a version of the latest update, it will submit transactions with a hash value and a few less crucial parameters.

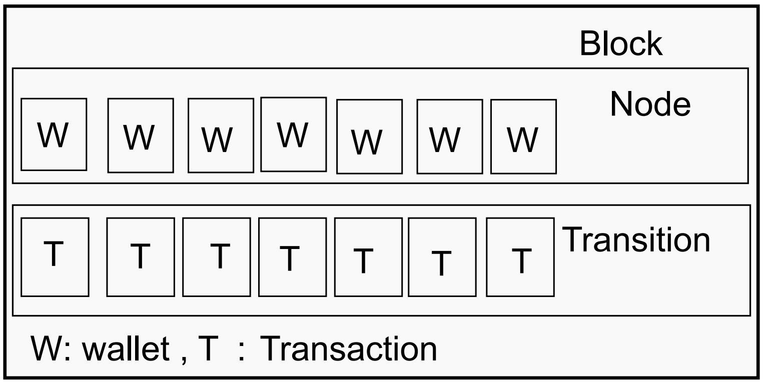


Figure 3.1: Block data structure

3.5 Data flow on TheChain level

The emphasis on TheChain level is to highlight the fact that the consensus is subject to complete graph agreements to duplicate the same decision. However, making decision verification traceable makes the need for consensus irrelevant to add a block. As demonstrated in TheCoin, by providing the recipient with the ability to initiate a transaction. In TheTree, highlighting the unmet legal requirement if the validator as a business does not pay taxes. The combination of the different concepts allows the system to provide users with the same level of privacy, stakeholders with transparency, and business owners as validators with fair competence.

Figure 3.2 illustrates the abstract picture by duplicating the graphical ledger between the different registered businesses while users focus on scaling the system. The building is the different kiosk that validates and manages the ledger to ensure high duplication. The left user initiates a transaction while the right user initiates information. The kiosk manages the concepts and their associated validation rules. Thus, the reception of information or value is linked to the rule. It will be appended from a regional perspective to logic and physical existence. Different files represent different types of information. A detailed discussion of how users are managed and part of the system will be provided in the Algorithm section.

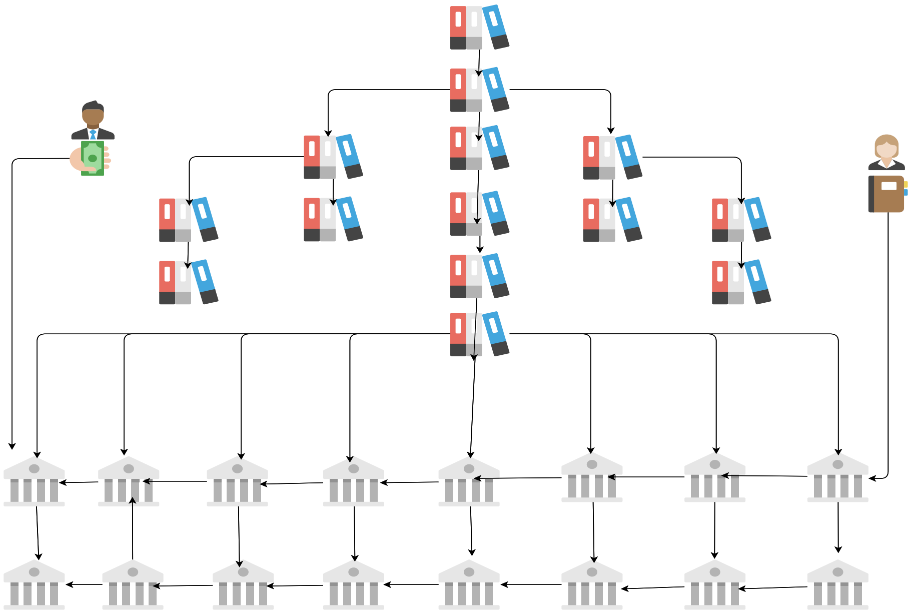


Figure 3.2: Abstract data structure flow

3.6 TheChain as a platform

3.6.1 Validation Layer

The blockchain network produces a large number of partially independent transactions recorded together within blocks before making the whole chain the subject of a search for related information for validation. The process is burdensome, and the need to validate the whole chain to append a new block is inefficient. Therefore, a validation layer capable of handling parallel processing of transactions quickly and correctly to disseminate the validity of the block to peers is necessary for scalability purposes. Algorithm 1 made use of the graph reachability to verify if the graph can reach that state with available criteria.

Moreover, it will be used to verify the logic of a submitted contract or a bill.

Algorithm 1: validation

```
   Input: listOfTrs
   Output: listOfvalid, VectorMarking, IncidenceMatrix
1  $wallets \leftarrow [wallets \times N]$
2  $Transactions \leftarrow publickey \times TranferredValue$
3  $PreMatrix \leftarrow keys \times TransferredValue$
4  $PostMatrix \leftarrow keys \times TransferredValue$
5  $vectorMarking \leftarrow findBalance(listpfTrs)$
6  $listOfTrs, listInvalid, c \leftarrow verifyTrs(listOfTrs)$
7  $BlockIp(listInvalid)$
8  $PreMatrix, PostMatrix \leftarrow buildMatrices(listOfTrs)$
9  $IncidenceMatrix \leftarrow (PostMatrix - PreMatrix)$
10 while $i < columnSize(IncidenceMatrix)$ do
11     $VectorTemp \leftarrow VectorMarking + IncidenceMatrix[i]$
12     if $NotAllPositive(temp)$ then
13         $DropNegative(IncidenceMatrix, listOfTrs, VectorTemp)$
14     $VectorMarking \leftarrow VectorTemp$
15     $i++$
```

In Algorithm 1, the information from the list of received transactions is used to produce a vector and three matrices in line [3-4]. It uses two defined data structures, which are the wallet and the transaction. The wallet must contain a balance, which represents the value of the aggregated coins created to the attached public key in line 1. The transaction contains the timestamp, the identity of the receiver, sender and the transferred value in line 2. The algorithm starts verifying the validity of the transactions, such as the digital signature, time attached and existence of the sender in line 6. It generates two lists of valid and invalid transactions and flags the IP addresses from which invalid transactions are received in addition to c, a confidence value discussed later in line 7. Findbalances is a method to return from the tree a partial marking vector related to the attached public in line 5. keys.BuildMatrices() generates from the transactions two matrices used to calculate the incidence matrix in line [8-9]. A matrix is a vector of vectors. Therefore, the loop continues to add each element to the temporary marking vector until the end. Within the loop, the checking of the temporary vector from negative value is held, and the function DropNegative(,) will drop the change and abandon the associated transaction

Table 3.1: Validation Tree

Tree Trend	Transaction Matrix and Wallet Vector
0	VM=[[1,2,3,4,5,6][100,100,100,100,100,100]]
01	TM=[[1 -30 -205] [2 20 10 -5] [3 10 10 0]]VM =[[1,2,3] [55, 125, 120]]
00	TM=[[4 -20 0 10] [5 20 -30 0] [6 0 30 -10]]VM =[[4, 5,6] [90, 90, 120]]
011	TM=[[2 -30 00] [3 30 -20 -10] [7 0 20 10]]VM =[[2,3,7] [25, 95, 30]]
000	TM=[[5 -40] [7 40]]VM =[[5,7][50, 70]]
0000	1. Sleep(1 Month)2. Apply (TM=[5 -40] [7 40]]) [3.back to1

to it. If all elements are positive, the temporary vector is affected to the marking vector. Moreover, for search limitation reasons, the public keys will be mapped into internal numbers. The transaction validation process is done through model checking by verifying the possibility of the graph to reach this state with the available criteria. However, each node nests a pool of transactions, with time processing that will vary with geographic distance dependability.

In Nick Szabo discussed previous work by Wei Dai (Popper 2015), with additional usage of cryptographic techniques which aim to automate the contractual relations because of the ability to virtualise the organisation, the intellectual and physical properties as entities within a distributed system. Ramchandani in (Ramchandani 1973) proposed the timed Petri network by attaching the value of time to model the temporal dynamic behaviour of a system. The contract proposed by Szabo and implemented by the Ethereum foundation functions as a proxy interface within the distributed system. This work injects sleeping Programming Threads within the validation layers and the associated time to apply the rule, which leads to the periodic application of the algorithm on the associated public keys.

Table 3.1 shows the growth of the tree within the validation layer. Element 0 contains six wallets; each public key mapped to an internal identification number associated with the balance. Element 01 contains the application of the incidence matrix with the use of masking and indexing on the partial vector, which contains three wallets to generate a new vector for this state. The transition matrix is a vector of vectors that always contains as the first element the internal identifier, followed by the balance gain for each transaction, wherein the vertical side describes the transaction with gain to each identity. The same application is applied to element 00, but in the element 011 the injection of

Table 3.2: Reference HashTable

Internal identifier	Trackers
158	Tracker1
25	Tracker2
856	Tracker3

a new public key is mapped to the value 7, and the system processes the new account by injecting it beside the most relevant balances, along with the sender. The goal is that applying the same philosophy leads to regions on the graph identified as a separate component. The earlier elements of the tree are periodically removed as they come with no use to validate the next transaction. A contract is a Thread embedded within the tree, and the element that falls under the branches 0000 is a contract that applies the transaction matrix every month. VM and TM stand for marking vector and transition matrix.

3.6.2 Injection Layer

The validation layer wrapped the validated partial vector with the associated transaction in a block and distributed it to the responsible nodes, and internally to Algorithm 2. The injection layer is a persistence layer within the node, and it must secure the backup of transactions on the hard drive. Each transaction refers to the next transaction with the same identity, and the network declares the finality of the transaction when it converges on the total order for a certain global and local sequential number. Moreover, a hash table 3.1 maps the identification number to a data structure defined in the class tracker, which contains the public key, the references to the first transaction, and the last injected transaction. The appending block algorithm begins by finding references to each identity on the list of transaction. In the case of the sender, it will change the first transaction reference in the tracker object, as the first coins will be used. Moreover, it will change for both the sender and the receiver the last reference as transactions added and updated.

```
$Class$ tracker:
  $publickey$ identity
  $reference$ first
  $reference$ last
```

The Algorithm 2 will receive from the first layer an ensemble of information regarding the validity of transactions. It will go through each identity separately and get their tracker from the reference hash table demonstrated in Table 3.2 in line [2-8]. UpdateFirst is applied to the sender tracker to change the first reference variable as the first coins are used in line 4. The UpdateLast method is applied to both the receiver and sender to refer to the last appended transaction in line 5. In line 7, the UpdateHashTable, updates references for the next search before building the block in Block() that calculates the most relevant previous block in the graph by ComponentPower before adding it to the graph in line [10-12]. ComponentPower will not be expensive because of the assumption that each cartel will maintain a region. Figure 3.4 is an architectural demonstration of the decision chain within the ledger, in which Mv stands for a partial marking vector, IM for incidence matrice.

Algorithm 2: Appending

```
   Input: listOfTrs, Graph
   Output: Graph
1  $i \leftarrow 0$
2  while $i < columnSize(IncidenceMatrix)$ do
3      $tracker_r, tracker_s \leftarrow getTrackers(t)$
4      $UpdateFirst(tracker_s)$
5      $UpdateLast(tracker_s, , tracker_r)$
6      $list \leftarrow [tracker_s, tracker_r]$
7      $UpdateHashTable(list)$
8      $i++$
9  $HashValue \leftarrow MerkleTree(listofTrs, timestamp)$
10 $blockprevious \leftarrow ComponentPower(Graph)$
11 $B \leftarrow Block(listOfTrs, blockHashValue, blockReference)$
12 $Graph.add(B)$
```

3.6.3 Miners Governance

Governance is the art of orchestrating nodes to work together to finalise transactions by identifying regions of exchange. However, the whole system is managed by how many coins with a unique identifier do exist. Consequently, adding to the fact that regions intersect makes it easier to track any fake coin. Moreover, a linear registry will work with DNS, in which it stores the history of IP addresses that have committed malicious behaviour in the past. Each block contains information about a node with proof of its previous behaviour. Consequently, it will derive a table for networking information. In

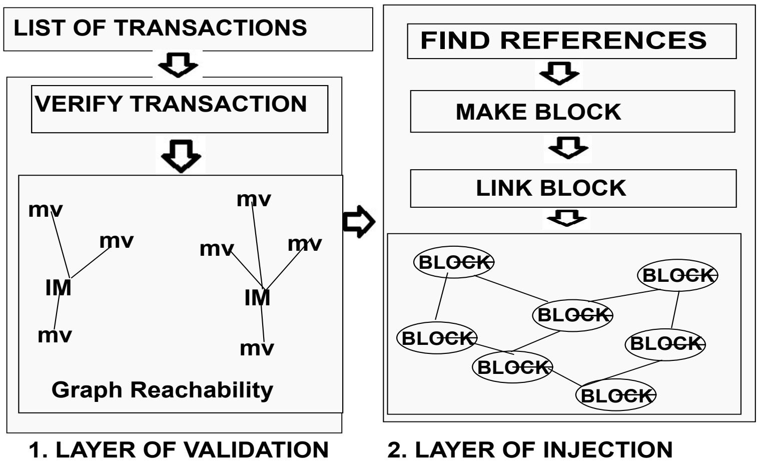


Figure 3.3: Decision Chain

addition, each node of the system is exceptional, with material resources and efficiency based on the following metrics:

$$solitude = \frac{1}{devices}$$

$$rapidity = Validation_{devices}$$

$$power = \frac{min(size(thp), Mspace)}{Rapidity(pr)}$$

$$conscience(node) = \frac{numberOfBlock(publicKey))}{numberOfBlocks()}$$

$$Confidence(block) = \frac{\sum_{n=0}^{n} verifySignature(t) \times validate(t))}{N}$$

thp : Throughput to the system

$size$: The memory space held by the throughput

$Mspace$: The Alive memory space

Pr : The peer identifier

$numberOfBlocks$: Number of a block with the associate public key

N : Number of transactions

The confidence metric aims to evaluate the validity of the block, and for each transaction, it gives a boolean whether it succeeds or not. At the block level, it will produce a percentage c value disseminated in the event of fault behaviour to be recorded with the block in the DNS linear register. The solitude is a metric that evaluates the node by its reliability on the external computing machine. Moreover, the rapidity metric does depend on hardware devices available to the validator and the used programming language; however, in the initial stage, the value will be CPU clock dependent. The leadership of broadcasting (power) consists of assigning responsibilities and nodes with superior equipment to be candidates for the role of a star after having also evaluated their consciences. Conscience means the number of blocks processed and broadcasted by the node. The values will be calculated periodically and disseminated to the peers to readjust the network nodes by the governance algorithm described in Algorithm 3.

The intersection among regions is key for not exhibiting any elimination behaviour from one part of the cartel against another. The system grows gradually from a few maintainers that nested client directory and kept serving them by making their transactions public, to cartels that are responsible on a regional exchange, in which each maintainer is responsible on a partial part of the region. Maintainers will grow to understand that their advantage lies in cooperation with each other because each region is serving clients, but there is always a dependency on other regions.

Algorithm 3: Assign Responsabilities

```
Input: chain
Output: treeMap
$List \leftarrow getList(chain, [blockid, recipient, sender, validator])$
$G \leftarrow createGraph()$
$G.addEdges(TupleList(list[sender, recipient]))$
$Components \leftarrow G.getConnectedComponents()$
$parse(DNSledger, List)$
$responsible \leftarrow Intersect(components, List)$
$rankedResponsbile \leftarrow rank(responsible, listOfProprieties)$
```

Algorithm 3 begins by filtering the data in the chain, which contains only the transactions of the last two months and obtains the characteristics of the decision. Later, the penalty DNS registry will be continuously analysed in the data to generate a list that eliminates any previous malicious node from participation. Intersect will generate a list of maintainers that crosscheck the associated components before comparably classifying them into the three leaders in diffusion, consciousness, and solitude metrics. The region assumption is based on the centralisation of the graph on some data point. However, regions will be interested in maintaining other regions due to some exchange of values between them. In the case of reward, it will be set automatically by the governance algorithm or manually by the user who is responsible for validating the transaction.

3.7 Node Independency

TheChain objective is to build self-validating nodes that are enabled with a layer of validation for fast and parallel treatment of transactions. The network is governed by an algorithm that builds intersected regional maintainers. The proposal dropped consensus, which means the absence of convergence on a unique ledger. The normal function of the system is by setting a limited number of coins with unique identifiers that will be exchanged between the different users. The maintainers will operate in their region to make their customers' transactions public in exchange for a reward. It is in the interest of all the nodes to be up to date with the different exchanges to eliminate any fake coin generation. However, all the nodes will not be recording all regions' transactions due to the limited resources, but the closest regions keep up to date with the next regions' ledger, to build a complex sequence of regions that watch over the next to secure integrity, as illustrated in Figure 3.5 The nodes are independent of any exterior dictation of data, consequently eliminating any double-spend or fault injection of data that have a high impact on the network as a whole. Moreover, nodes operate in regions that lead to the elimination of any attacks that target the network liveliness. Figure 3.5 is a demonstration of the regional operating territories.

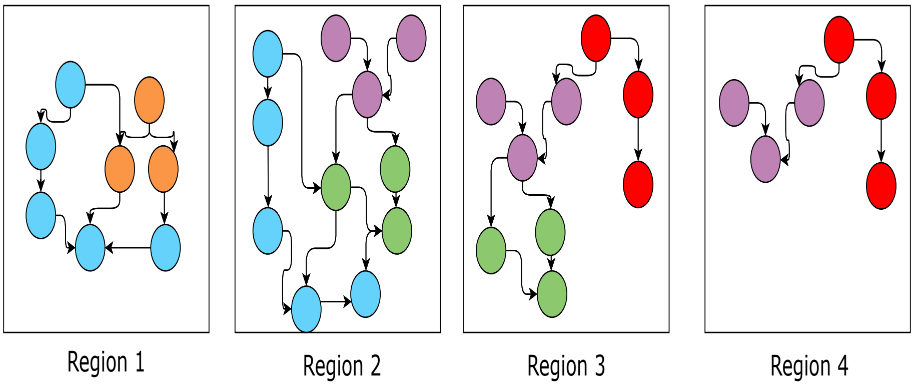


Figure 3.4: Regions

3.8 Safety Attacks

This part discusses the type of attacks that intend to threaten the safety of the system, which means the robustness to secure regular operation. It focuses on a convergence of nodes on the same chain to finalise several transactions. The longest chain rule within the linear ledger approach leads a malicious node to invest in its vulnerability by aiming to create a side chain to double-spend. The attacker starts by sending a coin, treated and appended, while spending the same coin in a different place within a side chain. The attacker keeps working on the side chain to make it longer before making it public, consequently cancelling the transactions within the previous one. However, in TheChain, nodes do not accept ledgers but a list of blocks that passes the same validation process before appending it. The usage of referencing of the transaction into a linked series allows the different nodes to identify the doublespend while converging to finalise the last transactions, because it will yield to a unique local and global sequential number.

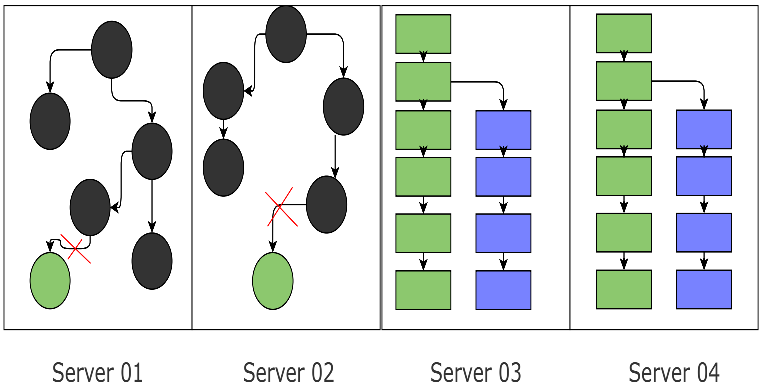


Figure 3.5: TheChain vs LongestChain

Figure 3.6 shows the difference between the two approaches. It shows as well that the graph could take many forms. However, the total order of the chain is secured on the memory reference layer in which all the transactions are appended sequentially and generate a precise local sequential number and a balance. Different kinds of attacks are investing in the longest chain vulnerability, such as a sibling attack that invests in creating many identities within the system and manipulates different peers' table to discover their neighbours. Nevertheless, the metric of confidence with a hard penalty makes this kind of attack inefficient.Another type of attack tries to invest in the vulnerability of the search algorithm over DAG or a tree structure.

The Tangle (Bu et al. 2020) runs over a DAG by adding a transaction called 'tip' before waiting for another user to append a transaction and validate two previous tips. The vulnerability is that if an attacker adds two conflicted tips in different leaves, this leads to what is called the splitting attack (Bu et al. 2019). TheChain addresses this problem with two technical choices, firstly by searching for the latest transaction to link the new one to it, and secondly by the validation layer which builds a balance from the different coin objects and ensures the validity before appending it.

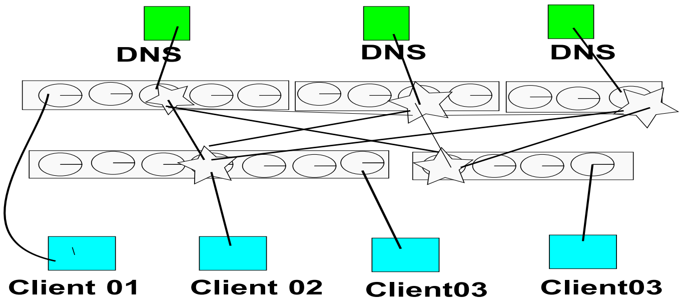


Figure 3.6: Liveliness Attack

Figure 3.7 is for demonstration purposes, to define how the whole network is linked. Even the clients are connected via a gossip protocol to keep up to date with DNS ledger. Nodes have the capability of communicating out of their cluster. On the other hand, different selfish attacks in Bitcoin ideology will be advantageous among different nodes, such as Block withholding (Bag et al. 2016), pool hopping (Belotti et al. 2018), and selfish miners (Eyal and Sirer 2014). However, this work is invested in such behaviour in order to build clients' directory for the cartel maintainer, in which each cartel must secure the finality of the transaction by making it public to get rewards, leading the system to function as a combination of many transfer companies.

3.9 Liveliness Attack

The work in (Kroll et al. 2013) introduced the Goldfinger attack, where platform competitors try to fail the system by different means; the paper discussed the vulnerability of a 51% attack within Bitcoin. However, as discussed in the previous part, the mining cartels concept, combined with partial responsibility, leads to the assumption that validators must be cooperative in exchange for a periodic reward. The illustrated network in Figure 3.9 shows how a malicious node must obtain the cooperation of all broadcasters to pass the fault transaction because of the inconsistent local and global sequential number that it will produce. However, if the number of nodes increases, the message complexity will increase and may lead to time delay due to difficulty to converge into a unique regional graph. Consequently, the resilience of such liveliness attacks will increase (Chari 2003).

3.10 Proofs

This section starts by giving proof of convergence to the solution. Secondly, it discusses the monopoly issue in previous works and how the proposed system has fixed this. It demonstrates as well how the system deals with a huge number of malicious nodes. Based on the universal generalisation theorem (UG), if an element of the disclosure universe has proved an assumption with a chain of rules deduced from axioms, this means the proposition applies to all elements.

3.10.1 TheChain Proof of Convergence

A distributed system is a set of entities connected through a network. If they are functioning as a system, then there is an updated sequence of information sent by an initiator to the system. These updated sequences are subject to validation rules. Validation rules are generally based on an incentive to drive the system. Additionally, the existence of a malicious member of the environment should be considered an anti-convergence concept. However, the high level of duplication associated with incentives is conditioned by the rapid dissemination of information to reduce malicious activity.

UG can be explained by the fact that an action performed in a space is conditioned by the property of the universe where it exists. The question is how to make the concept of rapid propagation a normal inheritance of a topology through its properties.

If we observe a topological architecture such as the gossiping model in figure 3.9. It can be noticed that the whole network in the propagation case grows linearly in time as each group chats with the next.

In TheChain, the first rule is If a node (e) receives an invalid block (b) it will flag the sender:

$\forall e \in D, b \in B \; invalid(e,b) \rightarrow flagSender(e,b) \wedge updateDNS(e,b)$ (1)

If we observe the first group in Figure 3.9 . The rapid spread was derived from a direct connection. However, if we add more groups, the concept of union in a space (not a set) has a big burden. Thus, if we tend to separate the groups, we will keep for each group a rapid propagation. However, if the notion of convergence must be satisfied, we must first allow them to perceive the same logic.

Firstly, assume the network is with a small number of nodes. Consequently, the time to receive a block (b) is neglected between nodes (e1, e2) (3). Moreover, if a block is invalid in e_1 it will be invalid for all other nodes (2)

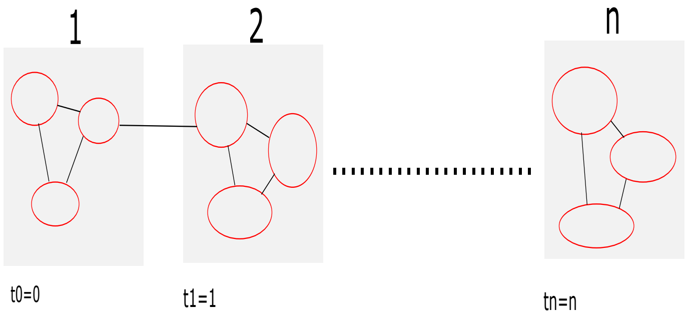


Figure 3.7: Gossip topology

$\forall e \in D, \exists e_1 \in D, \exists b \in B, invalid(e_1, b) \rightarrow invalid(e_2, b)$ (2)

$\forall e_1, e_2 \in D, b \in B, TimeReceived(e_1, b) \iff TimeReceived(e_2, b)$ (3)

Based on UG, if an element reaches that assumption and all nodes share the same proprieties, then it is valid for all of them by converging on the same domain (D) (5a,5b):

$D = D - r$ (4)

$\exists e_1 \in D, \forall e_2 \in D \; updateDNS(e_1, r) \rightarrow conv(e_1, DE1), Conv(e_2, DE2)$ (5a)

$\exists e_1 \in D, \exists b \in B \forall e_2 \in D \; UpdateLedger(e1, b) \rightarrow conv(e_1, b) \wedge Conv(e_2, b)$ (5b)

The second step is that we are going to separate from each group a special member and we are going to make sure that they hold the fast propagation properties between them. It can be summarized in direct connection.

The network will grow massively, leading (3) to not hold any more. However, the governance algorithm will lead to direct contact between the star nodes (s), and by the addition of (6), the UG is valid again. $\forall e \in D, \exists s \in D, knowledgeable(s) \rightarrow knowledgeable(e)$ (6)

NB: In this case, we are not discussing performance in terms of 0.1 seconds and 0.5 seconds, but architecturally how to preserve the fast propagation.

Thus, we took advantage of space separation to dictate different logical worlds of existence that preserve the property of rapid propagation to force UG to hold the validity of a conceptual action.

In Algorithm 3, a sorting list of nodes is returned, and each participant must pick the first B element as stars. If the probability of success in sending to the cluster is P and failure is F, picking B nodes receives the transaction to broadcast will make it with a probability of success as:

$P = 1 - F^B$ (7)

Lastly, adding the following rules that stated regions (r) would intersect, be driven by a reward and force nodes to be up to date with their regions:

$\exists e_1, e_2 \in D \exists r_1, r_2 \forall b \in B \; regionUpdate(e_1, e_2) \rightarrow UpdateDNS(e_1, r) \vee UpdateLedger(e_2, b)$

(8)

$\exists e_1, e_2 \in D \exists r_1, r_2 \in R, interest(r_1, r_2) \rightarrow regionUpdate(e_1, e_2)$ (9)

$\exists s \in S, \forall e \in D \exists r_1, r_2 \in R, \forall b \in B, Reward(b) \vee ExpectedReward(b) \rightarrow incentivized(e) \rightarrow FinalizationOfRules(s) \rightarrow interest(r1, r1)$ (10)

Rule (11) states that the open context of the system leads to a duplication of the general ledger translated into several assets. Next, rule (12) ensures integrity as a duplication inheritance.

$\forall l \in L \exists n_i \in D, Opencontext(n_i, l) \rightarrow ManyPosses(n_i, l)$ (11)

$\forall l \in L \exists n_i \in D, Manypossess(n_i, l) \rightarrow integrity(l)$ (12)

If many possess the information, then there is integrity, TheChain will converge in a regional way, making every node possess a special graph that contains its region and the intersected with it showing high integrity.

3.10.2 Longest Chain rule

1. If a node (n_2) has more means than any node (n_1), this means n_2 generates more blocks than n_1

$\exists n_2, \forall n_1 \in D, means(n_2, n_1) \rightarrow generateMore(n_2, n_1) \wedge leadMore(n_2)$ (1)

2. A miner maintains the ledger for a reward.

$\forall n_1 \in D, \exists b \in B, generateBlock(n_1, b) \wedge appendedInLedger(b) \rightarrow getReward(n_1)$ (2)

3. The choice between two valid ledgers (L1, L2) is for the longest

$\forall L_1, L_2 \in L, valid(L_1) \wedge valid(L_2) \wedge Bigger(L_1, L_2) \rightarrow choose(L_1)$ (3)

4. A transaction (t_1, t_2) can be valid in one ledger (l_1, l_2) and invalid in another

$\exists l_1, l_2 \in L, \exists t_1, t_2 \in T, validTransaction(l_1, t_1) \wedge validTransaction(l_2, t_2) \wedge \neg validTransaction(l_1, t_2) \wedge \neg validTransaction(l_2, t_1)$ (4)

5. A node (n2) is interested in maintaining the version of the ledger where it gets the reward.

$\forall n_2 \in D, L_1, L2, containReward(L_2, n_2) \wedge \neg containReward(L_1, n_2) \rightarrow maintain(L_2)$ (5)

6. $\forall n \in N, l \in L, s \in S \; leadMore(n) \wedge getReward(L) \wedge maintain(L) \rightarrow Monopoly(s)$ (6)

Based on (1)(3)(5), the network can be monopolised by the nodes that have an important part of resources because of the intention to keep getting rewards.

The following example is to explain how these 3 rules (1,3,5) are the basis of monopoly in blockchain. First, the other two rules (2, 4) should be taken for granted as an internal

Table 3.3: Miners sequence of blocks

	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
Miner 1:	A1	A2	A3	A4	A5	A6	X	X	X	X
1 switch to at T7	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
Miner 2:	B1	B2	B3	B4	B5	B6	X	X	X	X
2 switch to at T7	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
Miner 3:	C1	C2	C3	C4	C5	C6	X	X	X	X
3 switch to at T7	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
Miner 4:	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10

driver.

If we assume the existence of four nodes with the following percentage which represent their resources in the system: Miners one, two and three have 1/5 of the resources. However, miner 4 has 2/5 of the resources. We will condition the function chosen in (3) with 3 blocks to switch. In the longer term in a probabilistic way, equity will be applied. Table 3.3 is a demonstration of the growth of the ledger for each miner.

Miners 1, 2 and 3 from T1 to T3 were incentivized by rule (2) to obtain a reward and aware that the transactions for which they received a fee may be invalid in other versions. However, miner 4 will take advantage of rule (1) to generate more and invest in rule (5) to not risk his reward. At some point, rule (3) will take place and everyone will switch to the ledger force by minor 4.

The conclusion is that the rule of the longest chain applied on the means leads to the monopoly of whoever generates the most blocks. However, in the case of cooperation between 1, 2 and 3, it will be a question of turning to another foundation of trust and collective monopoly rather than an autonomous distribution.

3.10.3 IOTA Tangle Approach

First, assume that the blockchain is a combination of miners and stakeholders.

$\forall n_1 \in D, l_1 \in L, network(L) \rightarrow stakeHolder(n_1) \vee Miner(n_1)$ (1)

In the Tangle, there is no reward to the maintainer, but users validate transactions; consequently, maintainers are not interested in being part of the platform.

$\forall n_1 \in D, \neg reward(n_1) \rightarrow \neg mining(n_1)$ (2)

The blockchain makes the transaction public among many maintainers to increase its integrity by eliminating the monopoly.

$\forall l \in L, \exists n_i \in D, ManyPossess(n_i, l) \rightarrow \neg Monopoly(l)$ (3)

By the addition of the assumption that a SmallPossess is the negation of manyPossess (4).

$\forall l \in L, \exists n_i \in D, smallPossess(n_i, L_i) \iff \neg manyPossess(n_i, L_i)$ (4)

Based on (3) and (4)

$\exists l \in L, \forall n_i in D, smallPosess(n_i, l) \rightarrow monopoly(l)$ (5)

In conclusion, IOTA tangle, by not providing a reward to a random validator, is conceptually vulnerable to monopoly, and from rule (12) of section (3.10.1) and (5) there is no integrity either.

3.10.4 Proof of No Monopoly in TheChain

Automatically, the client will be assigned to a cartel. However, it was stated that the role of the broadcaster is to make the transaction public in exchange for the reward, and a client (c) can also choose the validator (e) manually.

$\forall c \in C, \forall e \in E, chose(c, e) \rightarrow validate(e)$ (1)

Moreover, if a validator (e_1) is up to date then it will validate, get a reward and be interested in the ledger validity.

$\exists l \in L, \forall e_1 \in D, \forall t_1 \in T, upToDate(e_1) \rightarrow public(e_1, t_1) \wedge reward(e_1) \wedge maintain(e_1, l)$ (2)

Monopoly is defined as the capability to dictating information on the ledger

$\exists l \in L, \exists e \in E, \forall d \in D, dictate(e, d, L) \rightarrow Monopoly(e, l)$ (3)

It must be accepted that the capability to choose will never lead to dictation.

$\exists l \in L, \exists e \in E, \forall d \in D, \forall c \in C, chose(c, e) \rightarrow \neg dictate(e, d, L)$ (4)

The rules (1, 2) represent the operation of the system. However, rules (3,4) demonstrate that the basis of monopoly cannot exist.

3.10.5 Scenario

The following scenario in figure 3.9 was implemented and tested to demonstrate the case if all but one element of the regions start acting maliciously. It was created by running four nodes to communicate with a set of blocks where a transaction intended to double-spend, while the high probability of generating blocks on the right side due to the use of PoW has forced node 1 to settle for the red register. TheChain has dealt with the problem, as shown on the left after communicating the local and global sequential number.

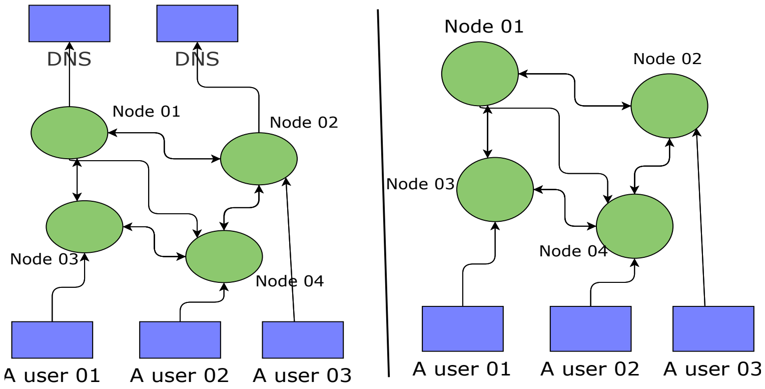


Figure 3.8: TheChain Vs Bitcoin

Table 3.4: Conceptual comparison

	PoW	PoS
Network failure	50% hash power (Kroll et al. 2013)	50% stake value (Xiao et al. 2020)
Fault data injection	50% hash power (Saad et al. 2019)	50% stake value (Xiao et al. 2020)
Double spend	Longest Chain (Jang and Lee 2020)	Long-Range
Transaction finality	Longest Chain (Xiao et al. 2020)	Longest chain (Xiao et al. 2020)
	IOTA	TheChain
Network failure	1/3 of network hashing power	Nodes are independent
Fault data injection	Splitting attack (Bu et al. 2019)	Nodes are independent
Double spend	Splitting attack (Bu et al. 2019)	Nodes are independent
Transaction finality	cumulative weight rule (Xiao et al. 2020)	Initially 100% of stars

Although the example was for the internal behaviour of a region, regions as a whole will grow to understand that any proof of malicious behaviour will cost them their customers to the intersected regions.

The attacker may choose to attack stars nodes in order to fail the system temporarily. However, the massive number of nodes will lead to a large number of star governments by an algorithm, and all the nodes in cartels will converge to the next in the waiting list if the main star fails. It must be clear that stars are not leaders of validity as used in BFT, but leaders of broadcasting.

Table 3.4 shows a comparison of TheChain with other consensus-based approaches. Consensual approaches aim to converge on one version of the general ledger at a time. These approaches are vulnerable to monopoly and manipulation due to dependence on resources, stakes or votes. Therefore, PoS and PoW are subject to fault data injection, network failure and double-spend by the monopolist due to the reliability on the longest chain rule, as explained in subsection 4.3.3. IOTA has a central feature, which makes it vulnerable to more than the third attack alongside the data structure choice which is vulnerable to double spending and the injection of fault data by the monopolist via the splitting attack. However, independence in TheChain is achieved by stopping the convergence on one ledger to eliminate any dictation of data and investing in the broadcaster's intention to make the transaction public to secure a reward. The nodes of the chain are independent of the other dictations of an order. Their operation is limited to being up to date with their environment because total control is forced on the transaction layer. Thus, the node will not wait for an agreement on a certain command. Network failure is not subject to the means used to vote or win a race as users assign their validators. The eclipse within the network is driven by the tragedy of the commons, which means validators' interest is not the user but the rewards. Thus, the user is assigned to a validator who provides his signature as proof of validation. As a derivation of the last statement, double spending requires the validator assigned to the initiator by a pre-verification, which will be explained in the TheTree chapter, or by an incentive, which will be explained in the TheCoin chapter.

The transaction finality depends on the longest chain convergence in PoW, PoS or the cumulative weighting rules in the IOTA approach. However, in TheChain is based on broadcasting transactions initially to all the stars before converging on the most relevant regions. The integrity of TheChain and PoW is very high due to the openness to any participants, which lead to high duplication, and the motivating reward to maintain the ledger. Consequently, there is a high distribution among nodes. Privacy is the state of being free from public attention; the various consensual approaches, as well as TheChain depend on public and private keys to validate a transaction. On the other hand, the problems of the link between the public key and the real identity, as well as the right to be forgotten are confidentiality problems; these problems arise from the data structure and the networking choices. Therefore, Table 3.5 shows high privacy of all techniques for validating a transaction within blockchain technology. Finally, IOTA tangle and TheChain operate over a graph that enables parallel treatment of transactions leading to high scalability.

3.11 Criticism

Although nodes must be up to date to attract more customers, some regions will act selfishly by abandoning the processing of some other regions' transactions, due to the zero exchange of money between the two parties. Consequently, it declares the lack of theory

Table 3.5: Criteria comparison

	Integrity	Privacy
PoW	High	High (Zhang et al. 2019)
PoS	Low	High (Zhang et al. 2019)
IOTA approach	Low	High (Zhang et al. 2019)
TheChain approach	High	High
	Distribution	Scalability
PoW	High (Wang et al. 2021)	Low (Zhou et al. 2020)
PoS	Low (Moindrot and Bournhonesque 2017)	Medium (Zhou et al. 2020)
IOTA approach	Low (El Ioini and Pahl 2018)	High (Zhou et al. 2020)
TheChain approach	High	High

on the regions' behaviour. Thus, the next work will invest in the region's intersection to build a solid theory for the relationship between region size and the network, besides enabling the multi-label classification to attach one transaction to many validators. This ensures that the whole network is an intersection of many regions that are partially watching each other to increase integrity and get neighbours' customers in case of malicious behaviour.

3.12 Conclusion

This work is suitable for the IoT sector and banking systems due to the introduction of partial responsibility on the ledger that leads to territories, besides the zero computational fee invested on transaction validation. To sum up:

1. Taking advantage of the Petri network structure to build the ledger and enable total order among the participant on the memory reference layer, leading to the elimination of attacks based on forking.
2. Validation layer that uses the graph reachability to enable fast and parallel treatment of the transaction.
3. Introduction of the concept of region intersection to ensure the validity of the ledger.
4. Definition of a governance algorithm that keeps clustering to lead to a rapid convergence within the network
5. Comparison of the proposal with previous work found in the literature.

4 TheCoin

“If cash is paper signed by central banks, can it be replaced by data signed by the central bank?”

4.1 Introduction

Money laundering, counterfeiting, and theft are the results of information manipulation of the recorded financial ledger or the lack of means to trace and verify the authenticity of a claim. The use of electronic payments has eliminated many of the problems that tangible payments pose (Brunnermeier et al. 2019). Centralisation has always been an issue due to concerns about privacy and high transfer fees. Consequently, distribution within blockchain technology has not only eliminated the high costs, but also introduced pseudonymous management of funds. However, many techniques have been found to link the real and pseudonomic identities’. It can be concluded that the violation of the right to be forgotten and public exposure are drawbacks of this technology. Money laundering is based on the manipulation of the value of information and that manipulation is helped by the probabilistic finality and the lack of traceability’. Therefore, although public transactions are a strong deterrent’, they must be embedded with traceability techniques.

Tracking finance is very ancient and can be traced back in time to the Babylonian, Egyptian, and Sabaean civilisations. Today, double entry accounting is mostly used, and the use of triple-entry accounting is on the rise as well. Recording all information within the blockchain ledger to make it public diminishes the ability of malicious users to manipulate or elude traceability. The adoption of blockchain technology by the financial sector has been a hot topic of discussion and research in recent years (Luz and Farias 2020). The first proposal of the system was to develop a prototype that exchanges financial information and eliminates double-spend. The bitcoin (Nakamoto 2008) proposal aimed to create a new type of money and eliminate the trusted party, but this solution is vulnerable to monopoly (Nacer et al. 2020). The monopolist is turning out to be the new foundation of trust. It has provided a new approach to validate transactions by making a malicious node weak compared to those interested in the ledger validity for their financial benefit. However, it was clear from the first project that the aim is to eliminate banks rather than to target the concept.

The ledger is organised as a sequence of transactions nested with owned objects that have been generated during the mining process or through an exterior investment as another type of fiat money. However, within the transaction, there are two types of information: the unspent transaction model (UTXO) and the balance model. These two types of solutions have many disadvantages in a distributed environment. However, many types of transaction initiation can take place depending on the type of wallet used. Pa per, hardware, and phone wallets make the owner of the funds the initiator while the web solution assigns a trusted agent to manage the fund. The audit system in banking is based on the use of the internal network run by local servers and mirroring techniques (Weatherspoon et al. 2009). These kinds of systems intend to eliminate any double-spending by fostering membership and the centralisation of decision-making.

4.2 Related work

Bitcoin (Nakamoto et al. 2008) or Blackcoin (Vasin 2014) are proposals that have used the Hashcash PoW (Vasin 2014) in the validation of a list of transactions. Each transaction does hold an ensemble of valuable objects. The exchange model is named UTXO. The Ethereum Foundation preferred the use of the balance model as an updated account value over the state transition system. Zerocash proposed the use of Zero-Knowledge proof over the data structure to delink the transaction from the identity. However, it led to the need for a sophisticated wallet that can save all the related proofs. Although there have been many proposals to switch from PoW to PoS (Tang and Bennett 2010) or to take advantage of graphs in Tangle IOTA (Picco et al. 2000), the exchanged datum was always to be chosen from those two models.

The UTXO model is based on the continuous exchange of values described in terms of input and output attached to a transaction. The input can be seen as a list of duplicated coins that have been attached, in which the aggregated values must be equal or superior to the transferred value. It will generate an output that represents new coins. The transaction will state the sender associated with the input coins, and the receiver and the validator of the block will be associated with the output coins, in which the transaction will stand for the total transferred value and fees of validation. The transaction list will be wrapped into a block that generates a network reward for the validator (Damba and Watanabe 2007). It can be observed that, at some point, coins will turn out to be of no use except for wasting a great amount of memory. Moreover, in the case of the bitcoin platform, it is developed as a knowledge-base that duplicates all the coins attached to their owner identity. The drawback of this solution is the unexpectedly massive growth of the micropayment exchange. The criterion for the validation of a transaction is that the value of input must exceed the value of the output over and above the transaction fees and the new transferred value.

The Balance-Approach is a more natural approach for the management of funds. The adoption of this technique in blockchain technology started with the Ethereum project. The solution models the system as the growth of an updated balance. However, the data structure of the transaction contains the sender and receiver keys beside the transferred value. The updating of the account will be subject to normal number manipulation, but the distributed criteria make it vulnerable to the replay attack (Hoffmann et al. 2019). The solution introduced the nonce number that will provide each transaction with a unique identifier and eliminate the threat of a replay attack. The Ethereum approach has a strict condition on the nonce number, which leads to the elimination of any parallelism and the approach suffers from one point of entry requirement. TheChain (Deirmentzoglou et al. 2019) proposed the use of a data structure that contains a balance and UTXO model in which the balance variable is just used to accelerate the decision making. Consequently, it can benefit from privacy and the parallelism of coins besides the easy management of the contract with the balance model. The special aspect of the transaction that follows the UTXO model is its capability of holding many receivers, whereas, in the account approach, it is more appropriate to associate each transaction with a receiver and a nonce number (Balaji and Srinivasan 2010).

The zero-knowledge succinct non-interactive arguments of knowledge (ZK-snark) have been implemented in blockchain technology to provide privacy by unlinking the data from the identity (Jang and Lee 2020). The ZK-snark simply hides the true value by obfuscating submission to other peers within the network. It is based on the generation of verifier and prover algorithms that are cropped into many small steps. Each step is converted through a Rank-1 constrain system (Bu et al. 2019) with three matrices that contain, as elements, a simple number of Boolean objects, each of which stands for an existing variable The prover must send relation generated from a witness matrix to the verifier that validates and ensures the knowledge of the solution. The solution is used to ensure the secure exchange of information. It has attracted much attention since the ZeroCash proposal (Jang and Lee 2020). It aimed to solve many problems, such as tracking of identity online or analytics to understand the different exchanges (Bentov et al. 2014). However, the user may be a subject of tracking through IP addresses due to the rigid connection with peers that stand for miners through the DNS server that returns a specific peer for each user.

Previous works in the literature discussed the UTXO model that can be used to enforce high privacy, but the model leads to a high duplication and an expensive search in the ledger or in terms of memory with massive growth in the coin's knowledge base. On the other hand, the balance mode leads to a high validation schema at the price of privacy. TheChain has combined both approaches to introduce a model in which the user can benefit from easy management of funds through a balanced approach and the mixing of public keys to increase privacy. The adoption of the technology on a large scale needs a more convenient approach than entrusting keys to the third party to manage the user's personal wallet. Consequently, this work aims to use mobile agent nested zero-knowledge proof as a way to exchange public keys between the initiators and the receivers of transactions. TheCoin uses a fuzzy reference to associate the spent and unspent coins.

4.3 TheCoin Data Structure

The transaction as a component is the most exchanged element in the technology of the technology. this section will introduce the TheCoin model. TheCoin (Ikbal Nacer et al. 2021) is a proposed model to adopt the monetary ideology within the blockchain system from a fiat perspective based on authenticity criteria. TheCoin specifically addresses transaction modeling, research, linking, security, and privacy. It will be part of the persistence layer within TheChain system. The following is a definition of the data structure used:

```
Class Coin:
  double MAXVALUE;
  double MINVALUE;
  String identifier;
  double value
```

```
    $String[]$ parentIdentifier;
    $Map < Transaction*, double >$ fuzzinessMap;
    $int$ layer;
    $int$ issuerType;
    $Byte[]$ IssuerSignature;
    $Byte[]$ validatorSignature;
    $Byte[]$ OwnerProofSignature;
  $Class$ Transaction:
    $Integer[]$ sequentialNumberSender;
    $Integer[]$ sequentialNumberReceiver;
    $publicKey[]$ sender
    $publicKey[]$ receiver
    $double[]$ fundTotransfer;
    $List < Coin >$ coins;
    $Vector < Transaction* >$ nextSender;
    $Vector < Transaction* >$ nextReceiver;
    $Byte[]$ receiverSignature;
```

The class coin stands for an element of the data structure that holds a unique identifier saved in the identifier variable. The issuer, validator, and owner proof signatures are generated on a different level. The first coin is generated and signed by the issuer. Therefore, issuerType will be zero, and a sub coin that derives from the main coin will be signed by the validator. The signature of the new owner is a must in the exchange. Value is the variable that stands for a value of this coin, fuzzinessMap represents a pointer to the next transaction where this coin has been used partially or totally. Layer stands for the potential application of fuzziness over the controlled space described in MAXVALUE and MINVALUE in which each incrementation/decrementation stands for multiplication or a division by a hundred. The class transaction is a wrapper of many coins in TheChain. In this system, the user coins will be linked consecutively through the use of pointers. the nextSender and NextReceiver serve the system by providing a total order on the memory reference layer between all the transactions, generating an accurate sequential number for the sender and the receiver. Sender and receiver are public keys that stand for the management of the fund. However, the use of a ring signature provides the user with a high level of privacy. The fundTotransfer is the value transferred to the receiver who will be the initiator of the transaction with the signature.

Figure 4.1 shows the sequence of a transaction and the way each transaction is referred to in the memory reference layer. Fuzziness in the coin's data structure refers to the amount of unused value. However, in the transaction layer, the map will show how much has been used as a total and transferred to the next transaction.

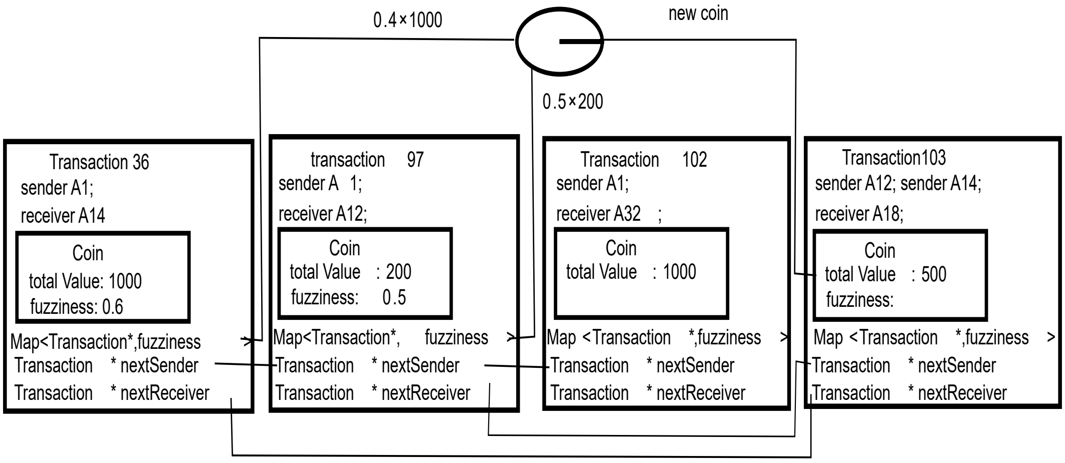


Figure 4.1: TheCoin Data Structure

4.3.1 Data flow of TheCoin

TheCoin data flow will be initiated by the first two processes, one on the sender and the other on the receiver. The sender will submit a Prover Agent and the Receiver will submit a Verifier Agent. The two agents will be nested with code that follows a zero-knowledge proof. During the validation process, the agents will communicate to complete the exchange before the TheCoin transaction is initiated by the verifying agent. Finally, the broadcast process to the neighbor will be started after the receipt of the TheCoin transaction from the component. The data flow is demonstrated in Figure 4.2.

4.4 Information's search

The search for related information can be a very expensive process in an open system. The Ethereum Foundation Balance-Approach can be faster for information retrieval as it is always a subject of addressing the last element to be updated, but it lacks many advantages of coin management, such as privacy and validation parallelism. TheCoin runs over TheChain data structure with the construction of transactions that are initiated. Algorithm 1 below describes the search for related information that is later sent from the validator to the sender to be signed before being injected into the transaction initiated by

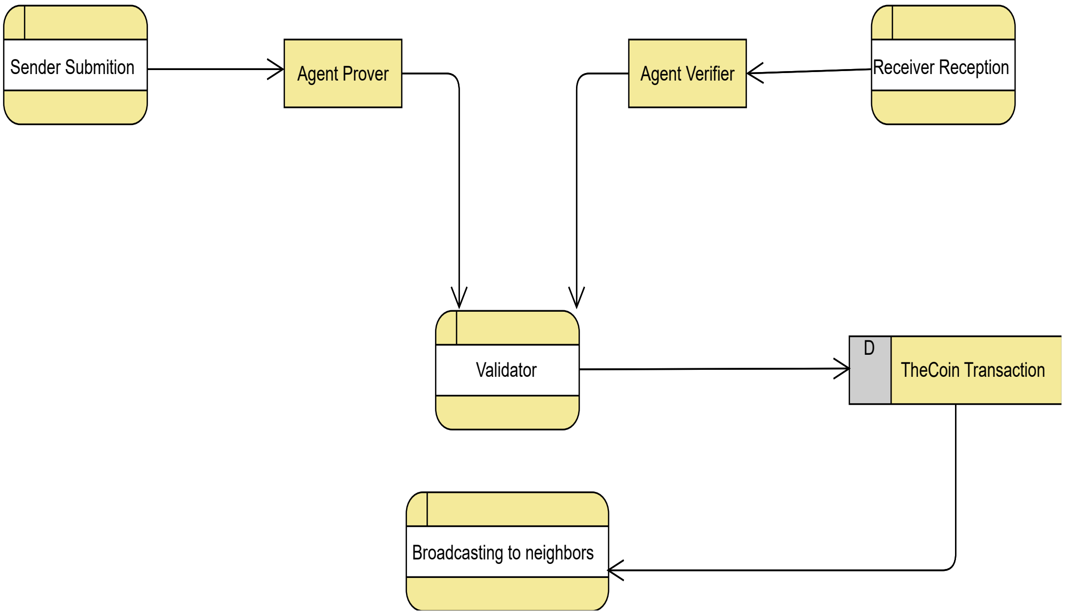


Figure 4.2: TheCoin Data Flow

the receiver.

The search for related information can build a vector from one or more. Consequently, the tracker, which stands for an object that saves the reference for the first and the last transaction of unused funds by the owner, is requested to give the memory reference of the attached sender's first transaction. It extracts a list of coins attached to the transaction before entering a loop to calculate the total value of the coins. Some coins are used totally. In the UTXO model, such a case is handled by using input coins on which the output is based. This approach causes duplication of an object that will not be used again. TheCoin proposes the fuzziness with a layer to identify the exact position of the last coin that was partially used. The setCoinfuzziness method takes the coin and the difference as parameters to generate a layer that represents the number of decimal places after a number in the hundreds. For example, a coin that holds 1000 as a value and then the owner pays 999.999, it will generate a new coin for the new value with a unique identifier. The remaining value of 0.001 is set in the fuzziness variable. The new coin is created through the generation of the unique identifier and its signing by the validator before being injected in the specific order to secure traceability to the original issuers.

The system checks the sender's balance before calling Algorithm 4. It first starts by setting a few variables such as fund to zero, boolean to false, coin vectors to later transfer between lines [1- 3]. It will point to the first transaction in line 6. Then it will enter a loop with relevance to complete the search by the done variable. It will extract the coins from the transaction on line 8. Then it will loop over the coins. GetTotalValue() will provide the first submitted value with the applied fuzziness attached to the coin. This means how much of the value is left to use. ValueToTransfer is a variable set whenever a pointer to a sender's first transaction is requested in relation to the submitted transaction. It will continue to add coins to the list at 19 and then point to the next one, which can be the sender or receiver of the transaction. However, it must be because the balance is already verified. If the expected value is reached by searching for the specific sender, it will enter an if statement at 13 to finalize the whole thing by defining the fuzziness associated with the last coins.

Algorithm 4: TheCoin Search

```
Input: listOfsenders, values
Output: vector<Coin>
$Int\ fund \leftarrow$ 0
$Boolean\ done \leftarrow$ False
$vector < coins > toTransfer$
for $(inti = 0; i \leq$listOfsenders.size();i++$)$ do
    $done \leftarrow$false
    $Transaction*\ first \leftarrow$Tracker.get(listOfsenders.get(i));
    while !done do
        for $(j = 0; j \leq$coins.size();j++$)$ do
            $fund \leftarrow$fund + coin.get(j).getTotalValue()
            if $(fund - allPrevisousValue(i)) \geq$values.get(i) then
                $done \leftarrow$ True
                if $(fund \geq$valueToTransfer$)$ then
                    $double\ value \leftarrow$ coin.get(j).getTotalValue()
                    $double\ difference \leftarrow$ valueToTransfer-fund
                    $first \rightarrow$setCoinfuzziness(difference, values.get(i))
                    $toTransfer.add(newCoin(value - difference))$
                    $break$
            $toTransfer.add(coin.get(j))$
            if $(!doneOR(fund - allPrevisousValue(i))$ ==$values.get(i)$ then
                if $(first.getsender()$== $listOfsenders.get(i))$ then
                    $first \leftarrow$first$\rightarrow$getNextSender();
                else
                    $first \leftarrow$first$\rightarrow$getNextReceiver();
        $updateFirst(listOfsenders.get(i), first)$
```

4.5 Transaction Validation

The transfer of money with blockchain technology suffers from unreadability of the public keys leading to its exposure on different forums or entrusting the wallet management to a trusted party. However, even if the user is capable of handling the last issue, the receiver may have to deal with the malicious activity of the sender by investing in probabilistic finality that may go through many stages by playing on rules such as the longest chain or network convergence (Ye et al. 2018). Moreover, users may use analytical techniques or sniffing to locate the current owner of the fund. TheCoin introduces the concept of mobile agents as the mechanism of exchange of keys and validation of transactions based on solving a zero-knowledge proof puzzle between the two parties. Though the initiator of the transaction may be either the sender or the receiver, by default, that person is set as the receiver, and the mobile agent is dedicated to verifying the proof of transfer. The agent is defined as an extension of the object and enriched by the concept of autonomy (Balaji and Srinivasan 2010). Autonomy is the capability of the agent to seek only its own interest, which is derived from its ability to decide.

The message contains the basic variables that are shared to validate the identity and transfer; it also contains the public keys that are the signature that validates the transferred value generated from the owned coins. The message also generates the prover signature and the shared sentence and the public key for the prover algorithm. The verifier agent expects an ensemble or a sender to participate in the transaction, in which it will be saved in a $senderNickNames$ vector initiated by the user. The same $AgentID$ helps in identifying the agent to serve in the exchange. The list of messages contains an ensemble of the received messages from the expected senders. The agent will be moved to a specific location that stands for the container ID or platform ID of a validator. The agent will register its service on that page, stating that it will appoint a validator to a transaction with a specific ID number. The agent will enter a loop that is initiated with the size of the expected sender for this transaction. The method, $receiveBlocking()$, will be blocked till a message is received that the sender will be checked to see if is a member of the senders' list before being added to the list of the messages. Finally, two behaviours, named $verifierBehaviour$ and $intiatTransaction$, will be added to the agent to collect proofs. Finally, the agent will be back to the receiver with a transaction to be signed before the broadcast for validation.

The prover agent is sent from the transferer wallet to the validator container or platform, to share the public key and/or help in the initiation of a transaction. In the setup, the method is moved to the expected location of the exchange. Secondly, the yellow page of the platform is searched for the expected service to validate a specific transaction. Finally, both agents execute the expected behaviour, and another agent, named Agent-

ProofProvider, provides a signature to the verifier agent. The verifier behaviour plays the role of a recipient who aims to validate a transaction. The transaction that holds no value has the goal of sharing public keys between the two parties. The behaviour is a one-shot behaviour in which a loop runs to extract specific information from the messages and to verify the identity before initiating a transaction with no value. That transaction will enable the senders to identify the public key of the receivers to sign the data. The prover behaviour is set to transfer values. By this behaviour, it generates a signature to build the message containing the agentID of the verifier and sender public key. The PublicKey is the password that serves as a mediator to verify the signature of the owner. Besides the signature generated from the the PublicKey, the shared sentence validates the transfer of value.

AgentVerifier is the name attached to the verifier agent. It contains the name of the sender chosen for references, the list of messages to be later a stock of knowledge exchanged with other agents. The method Setup is defined as a special constructor that defines various components. The agent will be prompted to move to a validator's location. Then it will be registered in the services yellow page. It will enter a loop but will remain blocking for the reception of a message and will exit the loop when it reaches the size of the expected messages, derived from the number of opposing interactors. If the number of messages is reached. Then it will add two behaviors, which are verifier and initiator. The last step is to ask the agent to return to the recipient's side.

The following is a mobile agent that will run on the platform:

```
$Class$ AgentVerifier:
  $Privatevector < String > SenderNickNames;$
  $Privatevector < Message > messages;$
  $protected\ void$ setup():
    $move(location);$
    $registerInYellowPage()$
    $for$(i =0;i<SizeExpectedMessages;i++):
      $Messagemsg = receiveBlocking();$
      $if(msg! = null) :$
      $if(senderNickName.contain(msg.getNickNames()) :$
      $messages.add(msg)$
    $addBehaviour(newVerifierBehaviour())$
    $addBehaviour(newIntiatTransaction())$
    $move(receiverLocation)$
```

AgentProofProvider is the provider agent as explained above. It will be a request to move to the validator's location, a request to register, and initiate its behavior.

```
$Class$ AgentProofProvider:
 $private\ String$ service;
 $protected\ void$ setup():
    $move(location);$
    $AgentDescription()$
    $Agent = SearchYellowPage(service)$
    $addBehaviour(newProvideProof(Agent.getName()))$
```

The message is a simple data structure that contains the sender, his public key. Signature of the prover, sentence to be served for zero-knowledge purposes, Signature.

```
$Class$ Message:
    $String$ sender;
    $Byte[]$ thePublicKey;
    $Byte[]$ signatureProver;
    $Byte[]$ sentence;
    $Byte[]$ Signature
```

AgentProver is the proof agent. It will contain a service variable. The configuration method will be responsible for the constructs. it will first be asked to move to the location of the validator. ask for the description and register on the yellow page before adding the prover behavior. $Class$ AgentProver:

```
 $private$ String service;
 $protected\ void$ setup():
    $move(location);$
    $AgentDescription[]$
    $Agent = SearchYellowPage(service);$
    $This.addBehaviour($new $proverBehaviour(Agent.getName()));$
```

The other two behaviours that are executed perform the same two purposes of collection of proofs and verification of the signature as the validator before initiating the transaction. However, the introduction of the code mobility by Picco (Picco et al. 2000) coupled with the concept of agent that was introduced by Russell and Norvig (Tang and Bennett 2010) can raise many issues and concerns of security relating to the host site. The user may also be subject to tracking through sniffing. However, the authenticity of the transaction lies in the digital signature. The inter-platform transfer of code can lead to rigorous interoperability standards that lower the performance. However, the concept has many advantages, such as loose programme modelisation, which leads to easy integration, maintenance, and introspection, and the server host is expected to be well-equipped.

Below is an expected implementation of the prover and the verifier behaviour:

```
$Class$ VerifierBehaviour:
 $Private\ VerifierModel$ model;
 $Private\ vector < Message >$ messages;
 $Private\ Transaction$ transaction;
 $Public\ void$ action():
 $Foreach$(Message msg: messages):
    $Byte[]$ public = $msg.getPublic()$;
    $Byte[]$ sentence =$msg.getSentence()$;
    $If$(model.verify(public,signature, sentence):
      $transaction.addsignature(msg, getSignature()$;
     $transaction.setType(msg.getType());$     $transaction.addPublic(msg.getPublicKey());$
      $Broadcast(transaction);$
$Class$ ProverBehaviour:
 $Private\ ProverModel$ model;
 $private\ Byte[]$ thePublicKey;
 $private\ String$ sentence;
 $private\ String$ agentID;
 $Public\ void$ action():
    $Byte[]$ Signature = model(password,sentence);
    $Message$ msg= MessageFactory(agentID, sender, thePublicKey, signatureProver
,sentence, Signature)
    $sendMessage(msg)$
```

4.6 Validator Billing

The initiation of the transaction within TheChain depends on fees designed for a specific validator to secure its public state. The public state is verified through the duplication of the same information with all-region validators besides the intersected regions (Nacer et al. 2020). The bill is a data structure that contains a map from services to a sequential number or contracts with the associated fee per transaction for each service and the total expected fees.

Algorithm 5: Bill Management

```
Input: listOfTrs
Output: map<transactions>
$intsize \leftarrow listOfTrs.size()$
for (int i= 0; i< size; i++) do
    $boolean\ submitted \leftarrow false$
    $Transaction\ trans \leftarrow listOfTrs.get(i)$
    $List < Bill > Bills \leftarrow getBilltrans()$
    $Profile = getProfile(trans)$
    $doublerisk = 0.0;$
    for (int j= 0; j< size; j++) do
        $risk$ += $calculateRisk(profile, Bills);$
        if (isRiskHigh(risk)) then
            $SubmitBill(trans, profile)$
            $submitted = true;$
    if (trans.containContract()) then
        $updateBill(trans.getContract());$
$updateBill(trans);$
```

There are two types of billing, the posterior and the anterior. The anterior is based on the new client buying contract that depends on the number of validations for a specific service, implemented above the graph of the validation layer , and will be growing based on two factors: the bought token related to the anterior billing and the current balance. The posterior option is based on a special service given to some users for which they pay after the service has been consumed based on the level of risk predicted on him. The receiver is obliged to pay the funds or lose the associated balance. The approach follows normal economic behaviour in which the user is subject to paying for retrieving information coming from government-related institutions, from the management of funds to the management of more complex information.

Algorithm 5 is executed before the injection of any block and the facture is associated with each receiver before calculating the risk implied by their profile and their current balances. Each user with a high risk receives facturation immediately.

4.7 User Centric measure

The idea of the UTXO model depends on the continuous generation of coins in which the validation of transactions requires the duplication of the previous coin and the generation of a new coin that holds the same value as the input. It states that the element of truth is centred around the transaction and forces the whole network to converge on one version of the ledger, which introduces a high competency between the nodes to converge.

The transaction, as central to the truth, is fed by several UTXO. In bitcoin ideology, the search for related information uses either a brute-force search from the root to the leaf or a bank of coins with a pointer to the attached transactions. The malicious user may invest in such vulnerability by building duplicate transactions with dispersed coins that can cause a delay in convergence and competency among nodes and by investing probabilistic finality and even double-spending through the generation of many meanless transaction. TheCoin's coin takes a different approach by using coins in a sequential order because it is expected to run over TheChain data structure and reputation-based system that forces the sequential number to follow the standards. Thus, TheCoin choices unleash parallelism executed within the same region comparable to rigid standards for out-of-region execution. Moreover, it allows the ring signature to be used, leading to a higher degree of privacy. Figure 4.3 depicts the difference, in which the link between the sequential number and the used coins eliminates the huge number of delays introduced by the competency model of PoW. The duplicated coin on the right side, which is number three, has been used on both newly initiated transactions leading the two portions of the network to compete to secure the reward.

The blockchain is a distributed peer-to-peer system that needs bootstrapping mechanisms. The use of a DNS server is one of the solutions. However, the returned peers are the source of truth for the user. This renders the system vulnerable to various kinds of attack such as RBG hijack. A study has shown that an RBG attack can eliminate 50% of

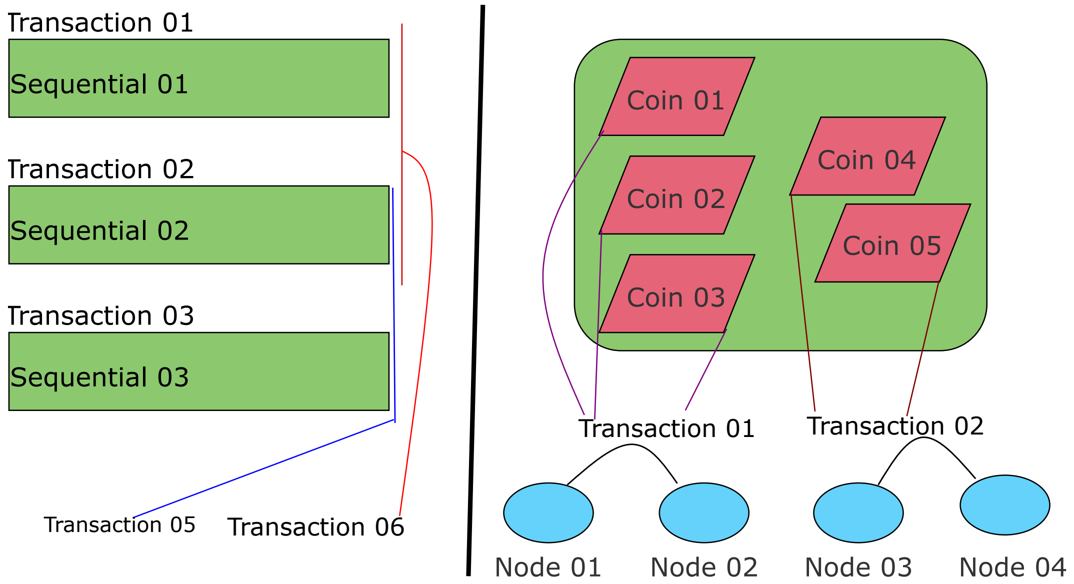


Figure 4.3: TheCoin model vs Bitcoin UTXO model

the hash power by the elimination of less than a hundred gates.The eclipse attack aims to isolate a portion of the network to provide a unique history. The initiation of a transaction by the sender increases the risk of double-spending by investing in providing a partial truth to the receiver. However, TheCoin runs over a protocol that uses a reputation-based system in which the validator is subject to continuous validation that may cause the loss of its business in the system. On the other hand, the validation mechanism allows the receiver to be the initiator of the transaction. The capability of the receiver to aggregate proofs of transfer from different senders taking a validator as the host for the exchange of keys and proofs diminishes the possibility of isolation because, unlike previous works, it imposes a structured network with a hidden topology. However, the unstructured building of the network makes users connect to unrelated peers who might be maintainers working on different partial centralisations with low interest in ensuring the validity because of the lack of reward from it or its traceability. Consequently, a peer-selection approach must be adopted. Figure 4.4 is a simple demonstration on how theCoin through the use of two agents that have chosen a specific validator as host, exchanged keys comparable with previous work that depend on the broadcast success of the sender.

The users in previous works suffered from poor key management and the sharing of the public key on social media or forums, which violated the privacy standards. Moreover, the lack of understanding of the aim of the blockchain system led many users to trust their keys to third trusted parties, the avoidance of which was the first reason for switching to the blockchain network from normal financial behaviour However, the protocol proposes the use of a mobile agent as the mechanism for exchanging public keys between the two parties. The users are subject to solving a zero-knowledge proof based on a password and a shared sentence generated and known to both parties. The users are subject to solving a zero-knowledge proof based on a password and a shared sentence generated and known by both parties. Although the solution can offer the user the promises of blockchain technology by eliminating dependability on exterior parties, it raises security concerns by the host that can be solved by forcing an internal template to object in reference.

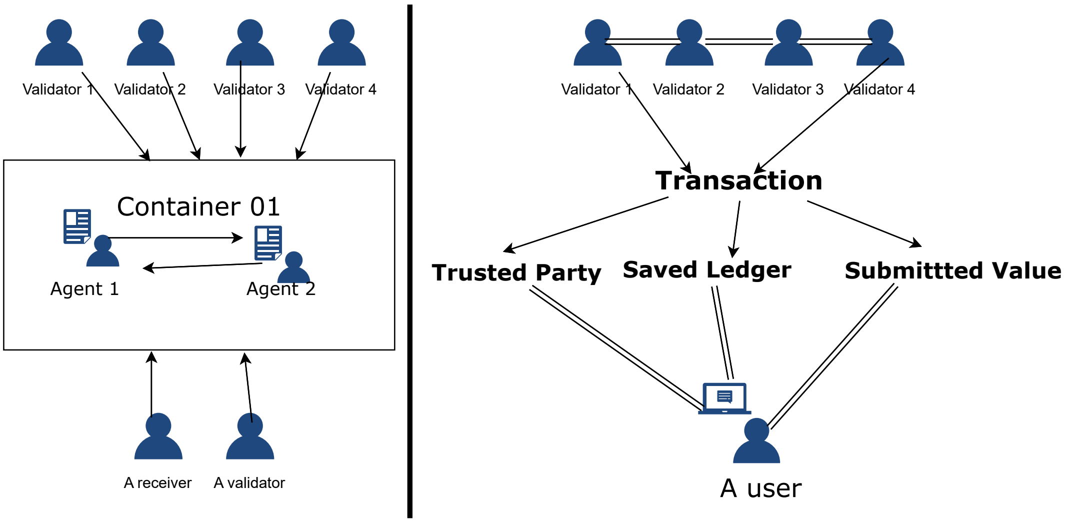


Figure 4.4: TheCoin exchange of keys

4.8 Data structure measure

TheCoin model has been implemented over TheChain data structure that has been built over the Petri Network model in which the choice of modelling offers the opportunity to link all transactions and updated wallets together. TheCoin objects are attached to transactions that are linked sequentially and tracked using memory references. The elimination of the input values reduces the size of the block by reducing the size of each transaction in it. Table 4.1 presents 100 blocks in detail in one ledger before the calculation of the size of each object. As can be observed, in the case of micropayment, the mean size can be ten times more on the UTXO than TheCoin. The object size can be larger if micropayment is lesser than the size used.

In our experiment, which was 0.001. In the case of normal payment, which is not micro, the average size of the object in the case of the UTXO model is 20% more than

Table 4.1: Block Size

	Micro (UTXO)	Not a micro (UTXO)	Micro(TheCoin)	Not a micro(TheCoin)
Transaction Nbr	100	100	100	100
Block depth	100	100	100	100
Mean	2217 kB	276.046 kB	204 kB	221.105 kB
Std	985 kB	7.833 kB	20 kB	4.498 kB
min	262 kB	247.648 kB	3 kB	208.008 kB
25%	1408 kB	271.728 kB	204 kB	217.976 kB
50%	2424 kB	276.568 kB	207 kB	220.824 kB
75%	3068 kB	281.432 kB	209 kB	223.672 kB
Max	3503 kB	291.128 kB	223 kB	239.336 kB

Table 4.2: Transaction Validation

	Implementation A speed	Implementation B speed	Implementation C speed	Implementation A with I/O	Implementation B with I/O
Transaction Nbr	100	100	100	100	100
Block depth	100	100	100	100	100
Mean	2.83 (ms)	4.013911 (ms)	631. (ms)	615 (ms)	722 (ms)
Std	1.02 (ms)	0.765979 (ms)	234 (ms)	290 (ms)	750 (ms)
min	1.88 (ms)	3.298200 (ms)	353 (ms)	292 (ms)	285 (ms)
25%	2.15 (ms)	3.493325 (ms)	455 (ms)	417 (ms)	383.5 (ms)
50%	2.36 (ms)	3.731950 (ms)	563 (ms)	551 (ms)	413.5 (ms)
75%	3.19 (ms)	4.187250 (ms)	775 (ms)	668 (ms)	467 (ms)
Max	6.93(ms)	6.490500(ms)	1249 (ms)	1727 (ms)	2994 (ms)

TheCoin. Moreover, it has been observed that since the standard deviation is not significantly large, there is no need to generate many new coins. It can be argued that the size of the block really depend on the implementation, and the duplication drawbacks can vary among the blocks. However, eliminating the duplication lowers the size, which makes a huge difference, particularly in the case of micropayment.

The execution of TheCoin shows the same negligible performance as expected when all coins addresses have been saved in a separate base of knowledge. Moreover, another implementation that can be very expensive is the brute-force search in which the search for related information can sequentially use coins as done in TheCoin model but it is very expensive because there is no sequential link between the different transactions in previous works. The used machine is a 64-bit Processor Intel(R) Core(TM) i5-8250U, 1.60 GHz, 1.80 GHz, and 8 GB. Table 4.2 is a presentation of three techniques. A stands for algorithm 4 injection with TheChain implementation. B stands for bitcoin ideology with a bank of knowledge that saves all coin references. C stands for the brute-force search within the bitcoin ideology, and the same implementation can be found in the work of (Nacer et al. 2020, Zhou 2019). As can be observed, the execution of 100 blocks in depth leads to the results discovered in which the difference between A and B is negligible with a mean of 2 and 4 milliseconds and with a low standard deviation due to the use of trackers for each element. However, C shows a very expensive search with a mean of 631 milliseconds and cumulative growth that affects massively the speed. The implementation with the use of IO access has demonstrated as well a negligible difference between the implementation A and B. However, in the use of micropayments, it is recommended that a base of knowledge with predictable models be implemented to manage the coins. Although the two approaches A and B perform very similarly on search, our approach eliminates the use of bank knowledge for coins through the use of pointers trackers.

4.9 System Measure

The exchanged datum has a great impact on system performance. Table 4.3 shows the difference in the aspectual contribution that the different implementations can have. TheCoin runs over TheChain. Therefore, it absorbs the different criteria from both approaches before making contributions due to the use of mobile agents. Parallelism is very high within the UTXO, but it may cause more delay due to jurisdiction between different nodes and transaction initiation from the sender side, who may be interested in double spend. However, TheCoin used the sequential order to eliminate this. Moreover, the use of the coin objects model gives the system the advantage of the ring signature that can enhance privacy as compared to the balanced approach, which, derived from this single point of entry, is easily exposed. TheCoin has facilitated the billing approach that uses contracts over TheChain system. However, it is hard to implement business logic over the UTXO model because it requires the generation of rewards, which does not fit within a fiat protocol. Finally, a consensual delay that can be derived from the valuable datum model can be very high in the UTXO model if it is implemented over a probabilistic finality with no sequential estimation. However, TheCoin with a regional perspective can control parallelism, which leads to low consensual delay.

Table 4.3: Conceptual comparison

	UTXO	Balance	TheCoin
Specific use of parallelism	High	Not applicable	High within the same region. It is controlled outside.
Privacy	High	Depends on pseudonymity	High
Bussiness logic management	Hard to implement	Easy to implement	Easy to implement
Consensual delay	High	Low	Low

4.10 Conclusion

This work has introduced TheCoin protocol. It is proposed to be run over TheChain system due to the risk involved in running randomly referred coins over a regional space. The solution, as has been discussed, has shown an optimal performance in terms of size and the security issues that it mitigates. The paper can be summarised as follows:

1. Use of fuzziness to manage the partial use of sequential use of unduplicated coins.
2. Use of the mobile agent as a method to exchange public keys between the different parties.
3. Introduction of the reverse approach in which the receiver is the initiator of the transaction.
4. The introduction of the concept of the bill within the permissionless blockchain technology.
5. The concept of the coin authenticity.

The next work will address the ledger where the token will be changed to more complex symbolic elements than numbers. It will invest in the concept of coin authenticity by changing the value to information and build logical chaining mixed with a distribution capturing approach from statistical methods to be applied to algorithm 1 and build a decision within TheChain.

5 TheTree

"If the state has always been a human choice, why is the bureaucratic institution not distributed among us?"

5.1 Introduction

Ibn Khaldoun in (Khaldun 2015) introduced sociology to the world and stated that the state has always been a human choice to maintain justice but questioned that the state itself is a force that acquires power unfairly. The story of an ancient society is summarized in a long road to a sophistication that ends with a huge focus on art before a foreign minority with the foundational skills comes to take over. Solidarity among people who speak the same language was the key to maintaining the society internally. However, the focus has been on the cultural clash, investing in bureaucracy as an internal issue against solidarity. Kansas City has experienced a large number of crimes involving special areas, into which considerable research has been invested in finding the best tactics for dispersing the police. The solution was found through the use of coupling within graph theory by associating dangerous places with a high number of police comparable to peaceful regions (Weisburd 2021). However, many social issues related to personal psychology can lead to social punishment, such as mismatch, transparency, and truth bias (Weisburd 2021).

Epistemology is the science that explores belief, truth, and justification from a fundamental perspective. The discussion of its philosophical background began in earnest in the 16th century. However, one of the greatest philosophical battles was between Averroes and Al-Ghazali (Zampaki 2018) over the impact of the philosophical discussion on metaphysics, which occurred before discussing the concept of necessity between effects and actions, which was profoundly discussed by Avvicina (MacIntosh 2017), in which Averroes accepted, to some extent, the role of philosophers and Al-Ghazali denied with a tautological rule, their influence. The modeling of the problem can be both probabilistic and deterministic. However, the probabilistic approach, led by the Bayesian network, presents many philosophical problems which exclude it from the field of epistemology (Spohn 2012). Any solution based on probability must investigate fundamental rules, such as Aristotle's rule of the excluded middle before any major ramifications can be drawn. Therefore, Spohn, in (Spohn 2012), adopted a deterministic approach called ranking theory to overcome the revision problem with the AGM framework (Delgrande et al. 2018). The ranking theory has not yet solved the problem, but it is a solid way to build a self-adapting system and to solve the problem of the prior extrinsically.

Blockchain's goal is to eliminate the foundation of normal human society, which is the state.The problem of malicious activities can be summed up in the same conflict of nomads with those who are sedentary. The ability of validators to monopolize the system can be seen as the issue of the periodic existence of a foreign minority that possesses the foundational skills. However, graphical analysis of the blockchain ledger has shown many cycles that can be inferred as ways to increase the value of cryptocurrency through a bogus exchange or double-spend events by investing in the longest chain rule. All of these latter issues can be justified or denied based on mismatch, transparency, and truth bias of human psychological interaction. Therefore, it will be difficult and unfair to implement a probabilistic model to deal with these issues. On the other hand, the deterministic approach can be appropriate.

This work imports social behavior into the system by investing in human nature. The authors proposed a model in (Ikbal Nacer et al. 2021), introducing a transaction initiation from the receiver side, in which this type of initiation is a driver of solidarity. This article asks the following question: "If the state has always been a chosen force, why are the bureaucratic institutions not distributed among us?". Specifically, the primary contribution of this work is the following:

1. The introduction of a novel approach to maintain ledger validity and preserve high scalability at the same time.
2. The introduction of the concept model, which can provide a simple, agile and flexible development approach for a dynamic framework.
3. The provision of security and modelized decisions as a network connection instead of a computation component to ensure reliability.
4. A theoretical study has been provided in terms of a security discussion, a formal

study of different components, modeling of the operating environment, and a conceptual comparison. Finally, the implementation of an actor model, network simulation, and unit tests have been demonstrated.

The whole vision of the system is to provide a new web where the user's view of truth is reputable, authentic and part of a regional preference that forces different versions of consistency. The web can be used for any type of value or managed information. Moreover, the modular dynamic growth of the system is based on a conceptual basis to generate a decision based on a network that explores different paths, making regional consistency another term for different objects. The following section will provide related work in the literature on different components, in addition to the existing parallel solutions. The third section will discuss a conceptual view in which knowledge is preserved through connections in addition to the ability to model everything with high security using simple concepts. The fourth section will present the different components of the proposed solution. The fifth is an evaluation of the approach. The sixth section is dedicated to testing, before the paper ends with a conclusion.

5.2 Related work

The blockchain as presented in the Bitcoin report(Nacer et al. 2021a) aims to secure a tamper-proof and tamper- resistant ledger. It was later explored as a way to maintain the validity of a ledger through many consensus techniques. The transaction is the user's initiative element, it contains an exchange object, which can be a UTXO, balance information or a different modeled token and a generated and verified signature with user's pair keys (Tuzi 2018). A list of transactions will be hashed using Merkel Tree and then injected into a block containing other information and especially the nonce number. It will be used to generate a unique hash value through random search to represent the high cost of malicious activities. Banks play the role of mediator between the depositor and the borrower and have developed massively in recent years. The use of technology and in particular blockchain can be another way to reduce the bureaucratic burden of the financial institution. The blockchain differentiates between two types of user, which are the simple user who exchanges values and the maintainer who validates these values. Proficiency is the key element among validators to ensure validity either through stakeholder decision or miners in the case of Bitcoin models.

PoW is an approach that designs a framework where a sibling attack cannot be practically performed. The hash power increases massively as the requirement for the leading zero increases. The ledger is built through competency to generate the longest chain. Many pieces of research have studied its distributed execution. For example, Eyal et al. (Eyal and Sirer 2014) studied the mining strategy, in which the race led to collisions within the system called pools. Each pool executes a specific protocol to divide the search space among the participants (Nakamoto 2008). Other selfish mining strategies that have been explored, such as block withholding (Wu et al. 2019a), lie in wait (Vyas and Lunagaria 2014), and pool-hopping (Belotti et al. 2018). Liao and Katz (Liao and Katz 2017) investigated Bitcoin ledger bribes using a whale transaction (Wt), which represents a high validation fee to trick the validator into aiming to fork. Many variants of Bitcoin PoW that invest either in compute-bound or memory-bound have been proposed, such as (Wu et al. 2019b). PoW suffers from high resource consumption, subject to 50% attacks (Shalini and Santhi 2019), monopoly (Nacer et al. 2020) and double-spend (Zhang and Lee 2019).

Many proposals have been published to improve Bitcoin implementation, such as improvised Bitcoin-NG (Das 2021) or Subchains (Rizun 2016). For example, the author in (Das 2021) focused on increasing throughput and fairness but the approach was prone to flooding attacks (Wang et al. 2019b) besides an incentive consideration (Yin et al. 2018). In addition, many proposals have invested in the random delayer such as PoeT through the use of Intel hardware (Kumar et al. 2019). PoSp is achieved by switching from the dedication of computation resources to the sharing of disk space (Tang et al. 2021). PoUW (Loe and Quaglia 2018) is achieved by ensuring that resources have been used to solve a useful task. However, the different implementations have been criticized due to security requirements. PoeT suffers from the lack of global control over the clock and PoSp suffers from the expected high level of resources required. The PoUW protocol suffers from the lack of incentive, unmet consensus requirements, and the impracticality of some proposals. Moreover, improvised Bitcoin-NG requires some synchrony that exposes the system to a DoS attack, and faces issues such as correctness, latency, and targeting through undermining the leader.

BFT was introduced by Lamport (Gramoli 2020) to solve the problem of the order of events. It was followed by Paxos, who came up with a solution to fault tolerance. Castro et al. (Haldimann et al. 2021) proposed the PBFT by extending Paxos to crash failures. It secures normal operations in a partial synchronization mode but with very high message complexity, it has been followed by many proposals to optimize its execution, such as Zyzzyva (Sohrabi and Tari 2020). Therefore, its suitability in the realm of permissionless consensus (Gramoli 2020) has been widely discussed. Hotstuff (Yin et al. 2018), implemented in Libera, aims to optimize the throughput by using BFT pipelining but this has introduced a longer chain of causal links between initiation and finality. Streamlet (Chan and Shi 2020) aims to increase fairness through the rotation of leaders. It has decreased the number of messages but still suffers from O (N^3) of communication costs applied at three rounds. Malkhi et al. (Abraham et al. 2021) introduced the flexible BFT which develops a dynamic quorum and addresses the issue of Alive-but-Corrupt members. Nevertheless, due to the high complexity of messages with bandwidth restrictions, the adoption of BFT in a permissionless blockchain has been met with skepticism. Thus, most BFT approaches have been proposed for use in a permissioned environment such as (Stathakopoulou et al. 2019).

PoS (Kim et al. 2018) is a solution that attempts to remedy the PoW consumption of resources. The incentive for valid participation lies in the fact that stakeholders will be interested in the ledger's validity, in which the validator selection process must follow a random algorithm such as Follow-The-Satoshi, Coin-Age, PoW random selection, or validator random selection. Many proposals in the cryptocurrency sector incorporate PoS and BFT as a voting mechanism to finalize a block, such as Tendermint (Buchman 2016), which uses BFT-spinning to manage throughput, or Ouroboros-BFT (Kiayias and Russell 2018). Chained PoS is based on a combination of PoW and PoS by securing a large number of participants via PoW and then switching to PoS. The delegated PoS (Fan and Chai 2018) is based on a community selection of validators, it is more closely associated in its philosophy with the delegate BFT. PoS, in its philosophical context, suffers from monopoly and mining cartels because an alternative chain is easy to generate (Zamani et al. 2018). However, the various hybrid solutions have not shown any advantages but have inherited the disadvantages of each technique at each level.

Tangle (Silvano and Marcelino 2020) is a proposal to solve the high fees within an open blockchain system. The solution offers a directed acyclic graph. The submission rate is the factor that eliminates manipulation with the use of a small Hashcash PoW puzzle on the user side. However, Tangle suffers from high consumption of distributed resources, which IoT devices may not be able to manage (Wang et al. 2019a), is prone to splitting attacks (Silvano and Marcelino 2020, Bu et al. 2019), 34% attack (Sayeed and Marco-Gisbert 2019), and monopoly (Nacer et al. 2020). G-IOTA (Wang et al. 2019a) is another selection algorithm used to overcome the left behind tips. TheChain (Nacer et al.

2020) has proposed solving the monopoly problem by introducing a capitalist ideology into the system, allowing different maintainers to nest a client directory. Two levels of maintaining the validity of the ledger have been proposed. It aims to build a regional intersection to ensure validity through the intersection of interests. The model proposed by the authors (Ikbal Nacer et al. 2021) is the datum that will be managed by TheTree, in which the proposed solution consists of the exchange of keys via the use of mobile agents in addition to the initiation of the transaction on the receiving side.

Peer-to-peer implementation is the basis of blockchain dissemination of information through the propagation of transactions or blocks. The topology of the network above in which the system is functioning is very important for its security. Network discovery is the first step for the new joiner, in which IOTA uses peers' gossip to forward neighbors' tables and Bitcoin uses DNS servers to extract seeds. On the other hand, a proposal such as Kademlia suffers from a lack of proof of its real performance (Dotan et al. 2021). However, restrictions on the inbound and outbound number of connections lead to forking when it is correlated with a high number of miners. Moreover, DNS poisoning (Al-Mashhadi and Manickam 2020) or RBG hijack (Awe et al. 2020) may undermine the network. Transaction propagation occurs through gossiping (Nencha 2021) or the use of the Geth protocol (Delgrande et al. 2018). Finally, block propagation is through the use of protocols such as weak block (Roy et al. 2018), Graphene (Ozisik et al. 2019), Velocity (Chawla et al. 2019), high and low compact encoded block, or Stratum (Recabarren and Carbunar 2017). Nevertheless, the network lacks a complete incentive that forces cooperation due to the functioning of the tragedy of commons embedded in the system, in which miners are not interested in clients' satisfaction but selfish gain. In addition, the geographic concentration of miners can be the cause of an RBG hijack, in which a study has shown that it is possible to remove 50% of the hash power is possible by eliminating fewer than 100 gateways (Saad et al. 2020).

AGM is a framework that has been implemented to study epistemological theory using the qualitative approach of formal logic. The system has three functions that describe its growth: expansion, contraction, and revision (Kern-Isberner et al. 2019). A revision will address rules that can be misunderstood to generate an unpredictable sequence of actions (Spohn 2012). Much work has been done to manage uncertainty above this domain, such as fuzzy logic (Zadeh and Aliev 2018), possibility theory (Mei 2019), and plausibility (Lai 2019). However, based on Gödel's incompleteness theorem, it is impossible to achieve infinite learning using the available formal logic because any system depends on an external assumption made by ourselves in the first place (Iacona 2021). In Bayesian network was, for a time, an alternative to managing uncertainty, but numerous epistemological refutations have been posted in the literature, such as (Spohn 2012). In the Bayesian ideology, it is irrational to be certain, there is no suspension of belief, it can describe content with many representations, and there is no support for iterative learning. Thus, Spohn (Spohn 2012) proposed the ranking theory as a deterministic approach to representing the dynamics of belief. The following is a formal representation of its conditional function and negative ranking:

Let R be a negative ranking function for algebra ,$a \in B$, $x \in R^{*}$, and $R(b)$, for $b, b \in B$, R is a ranking function from b into $R^{*} = R^{+} \cup \infty$.

$R(B)$=0, $R(\emptyset) = \infty$

$R(a \cup b)$ = min $(R(a), R(b))$

$R(a)$=0 or $R(\neg a) = 0$ or both

$R(\cup b)$ = min $(R(b))$

Spohn proposed a conditional function, but was criticized by Shoney for relevance and proposed a modification for evidence lead tracking [10]. The following is the function proposed by Shoney: Let R be a negative ranking function for algebra B, where b ∈B,x∈R*, and $R(b)$, $R(\neg b) < \infty$.

$$f(x) = \begin{cases} R(a \mid b) - y & \text{if } (a \in b) \\ R(a \mid \neg b) + x - y & \text{if } (a \in \neg b) \end{cases} \quad where = \begin{cases} min = R(b) \mid x \end{cases}$$

Community detection has been one of the main areas of research in a social network in which many greedy search algorithms have been proposed. A tree is a special data structure that is useful in many applied fields. It is a restricted graph that is directed and does not contain cycles. Many algorithms have been proposed to process the learning of trees. However, many questions have been raised based on the philosophical question of when to stop, in which post-pruning and overfitting, with some randomness, were the two choices (Chourasia 2013). The splitting of a node, in which a distance measure has been incorporated or an impurity function has been evaluated, has also been widely discussed. One of many implementations is FastXml (Prabhu and Varma 2014), which is a ranking algorithm that builds a random forest, taking into account the division of a node with the use of SVM (Yan and Jia 2018) and a stop condition based on entropy gain.

5.3 MOTIVATION

The motivation for this proposal is to address the bureaucratic workload of government by providing a social science-inspired algorithm to construct a new mode of belief as an adaptable internal decision-making system as users control its growth based on their needs provided by the concept manager, who are validators. The basic proposition is to provide a way to apply the same techniques that humans apply socially to gain power or deter against threads, which are reputation building and destruction. This work provides a new way to reinforce the belief in the distributed system named the concept model to be coupled with a sociological algorithm to act as a means of reputation building by providing new concepts to be used by customers or reputation destruction to deter malicious users. The ranking theory proposed by Spohn observes belief as a ranking of possibility rather than its exclusion from the space of actions. Thus, truth as an absolute entity does not exist because people's perception of reality is different, which makes Lewis' possible world another word to search for a coherent belief to integrate. The solution provided in this work is a way to respond to philosophical limits by playing on the following principles:

1. Increase efficiency of the system by lower the time of finality.
2. Provide a novel artificial intelligence method to be the background of a world machine.
3. Respond to the legal requirement on the personal and state level.
4. Increase means of privacy in the system through flexible modulation.
5. Provide a solution that trade off between real world requirements and fast propagation, treatments, and global decision of transaction.

Figure 5.1 is demonstration of the whole vision of the system.

5.4 TheTree

5.4.1 Data Structure

Profiling is an approach to measure the subject's tendency, risk, and normal behavior that can be targeted or evaluated based on it. It can be as well the modulization of the object of belief, in which running peers within a distributed environment must be seen

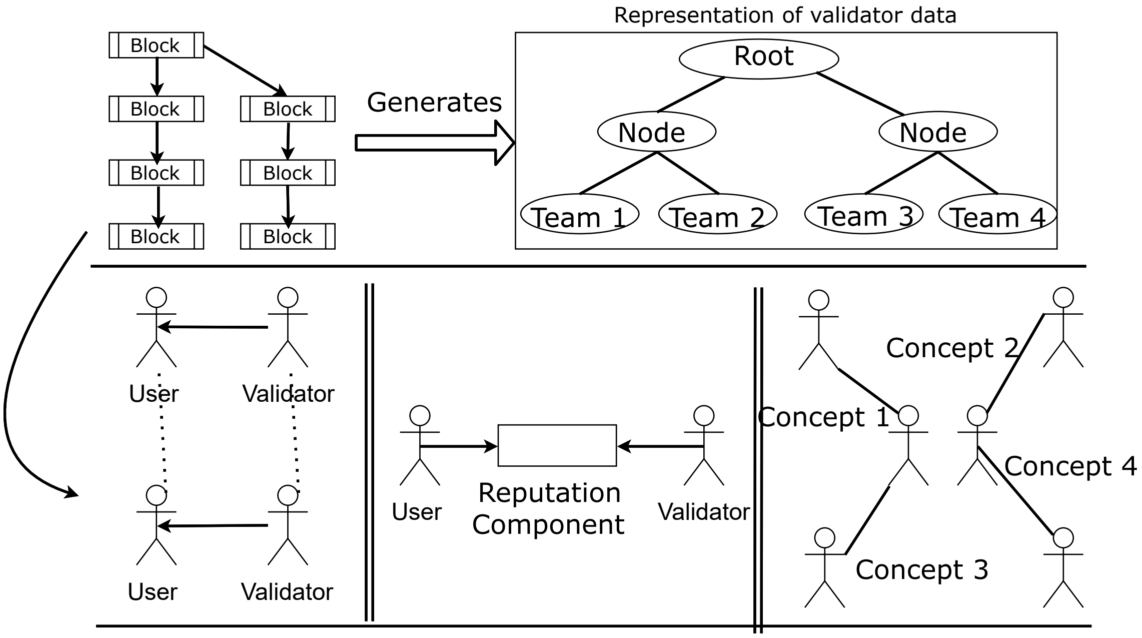


Figure 5.1: Modular flow

as Lewisian possible world (Nencha 2021), in which inter-world concepts can build an inferential belief. The belief as an object that does depend on properties and relations must not be taken in the relative sense of a concept but to a distributed entity itself. This section will introduce the management of belief within the peer, in which the different data structures that have been implemented will be demonstrated and analyzed. The classes are demonstrated in Figure 5.2.

The user profile contains an identity which is a public key. Risk represents disbelief in a connection, and it is bound to zero to be its suspension. Neutrality means the opposite of risk, which is a list of evidence of malicious activities. The risk is derived in terms of ponderations; in other words, each data point contributes differently to the risk metric. These ponderations must be constantly updated by an expert in the field to be up to standard. In addition, the client class contains a community number identifier and a list of bills (Ikbal Nacer et al. 2021).

The validator profile contains a business number, which provides a good measure of confidence for the user. History represents different peers that have a high exchange rate with the validator. Intersection represents the regional number of intersections. The client directory represents all the clients registered with the validators. The remainder is variables related to the validator's physical device noted in (Nacer et al. 2020). Relations between peers are managed through concepts, which have a name, a surprise factor that represents the mean of the relational values, and a list of relations. The relation class contains a partner name, a name or reference, and a conditional value handled with Shoney conditionality.

TheTree will be presented in a node and built recursively. It contains a list of validators who have reached the node and a logistic map that assesses the level of randomness in the choice. Concepts represent the instances of concepts managed between validators. Communities represent the communities detected via the Louvain algorithm (Singh and Garg 2021) which will only be applied to leaves of TheTree.

The peer in Figure 5.2 comprises three different lists of entities: concepts, a validator profile, and a user profile. The concept comprises many relationships that manage interaction with validators or users. A validator profile contains a list of clients that will be modeled in terms of communities, and a user profile contains a client object.

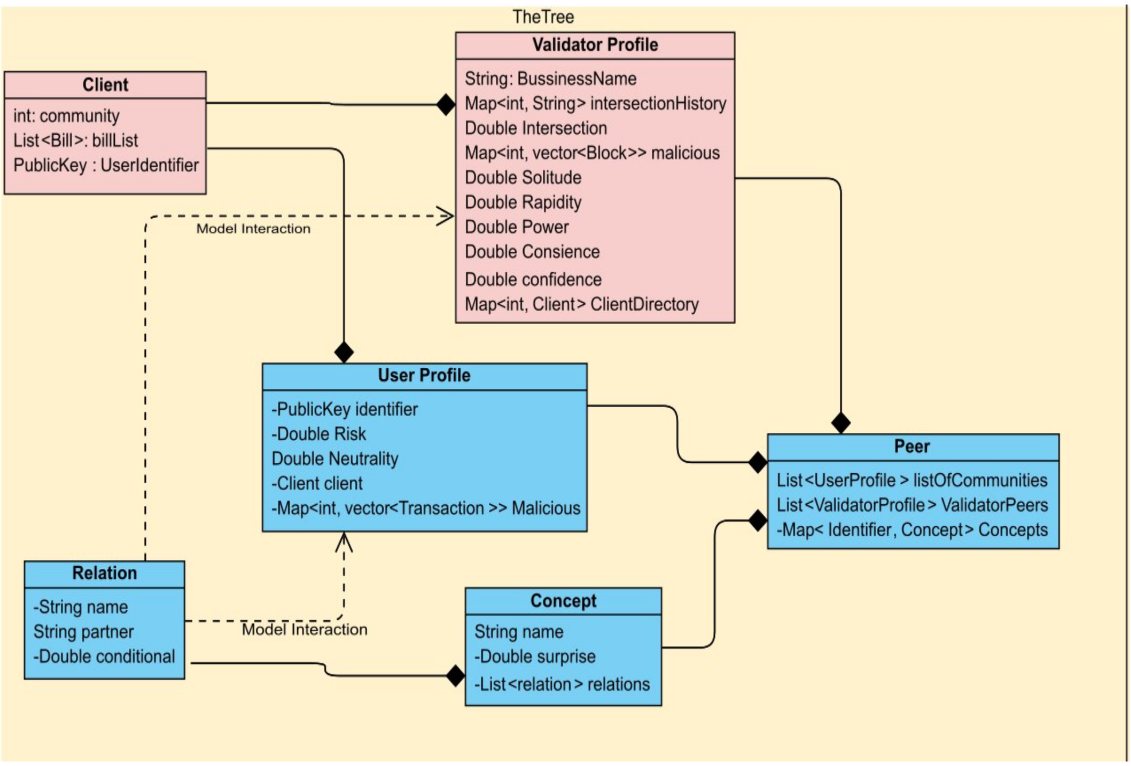


Figure 5.2: Class diagram

5.4.2 Data flow of TheTree

At TheTree level in Figure 5.3, the data structure is seen as an action-reaction set. The listening process will deserialize the data to be formed in terms of transaction, contract, bill or information. It will be passed to the processing process will result in different actions to be assigned to different entities, which update the profile of the user or validators, add the ledger, update the concept or add a new one. Finally, the data structure will be passed to the broadcast component to broadcast the information to the peers.

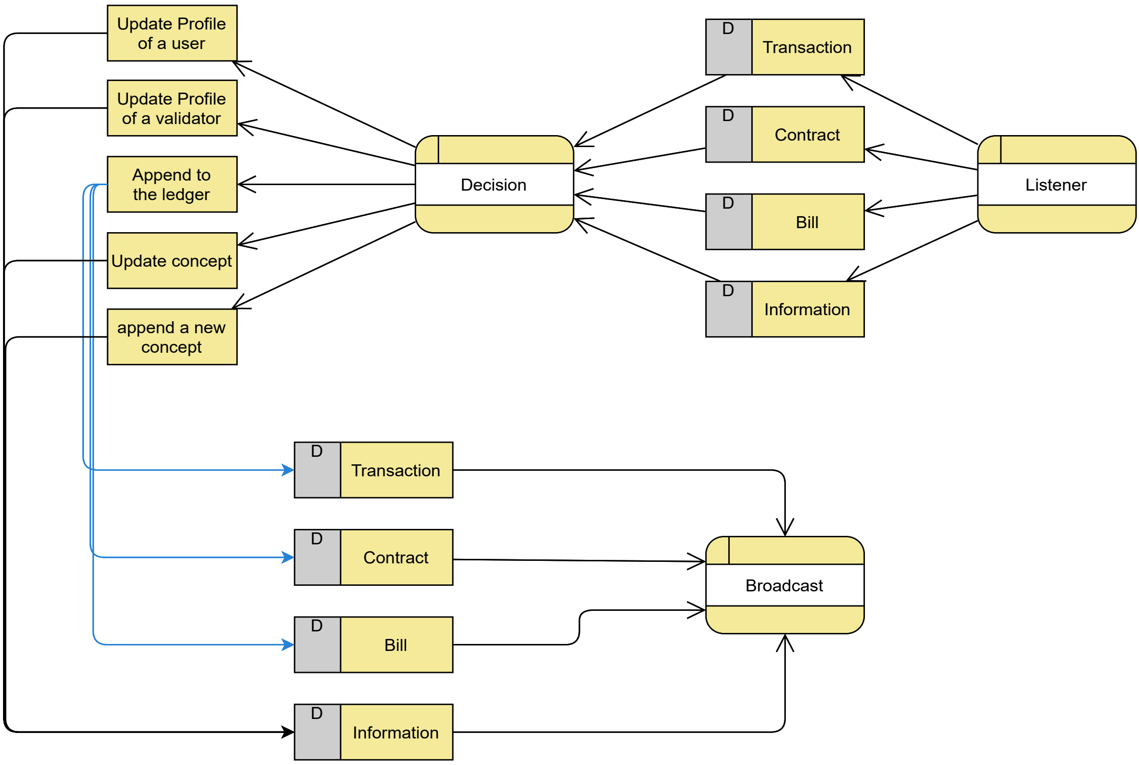


Figure 5.3: Flow of data

5.4.3 Reputation-based network

Building belief in the peer arrives through the management of the reputation with the existing world besides the value of the exchange itself. Harm is disbelief within an entity. The connection side will be managed continuously through Algorithm 6 and Algorithm 7, which update belief in relationships, before modifying the surprise in the concept through Shoney conditional function. First, Algorithm 6 receives a user profile and the vector of transactions. The user will count the number of duplications in line 1, inconsistencies in line 2, and forbidden actions in line 3. Punishment will be relative to the number of users in the initiator's community in line 4. The result will be evidence of malicious behavior to make the receiver unlink the binding with it as the disbelief in this entity turns out to be harmful in line 5. The variable will be updated in line 6.

1. Duplication: two transactions that contain the same sequential number and/or the same coins.

2. Inconsistency: a transaction received from a user in another region that does not stand this rule: $Sequential(i) = Sequential(i-1)+1$

3. Forbidden: a transaction with coins which have been used, not possessed, or $Sequential(i) < Sequential(i-1)+1$

Algorithm 7 updates the validator profile by checking in the DNS ledger whether the attached block has been registered as malicious through verifying content and identity. intersectionFactor is a variable that proportionally describes the expected intersection level, it will be 1.1 deducted from dividing the profile intersection over the maximum intersection found. if the block is invalid, it will enter to compute the proof. The proofs will be with different references. In the case of relentlessness, it will be calculated by multiplying the intersection factor with the result of multiplying the doggedness and the number of validators. If it has been overlooked, it will be calculated in terms of the intersection factor with the number of validators. Otherwise, it will inform the validator. Finally, it will update the rank for the validator. Thus, the platform community growth has high relevance to punishment. Moreover, it will be harder for highly intersected nodes compared to others to participate in any misbehavior.

Algorithm 7 manages validators and ponderate profiles according to three criteria:

1. doggedness: the act of resubmitting a block that contains proven malicious activity

2. overlooking: the action of distributing user data containing prohibited transactions for validation

3. Region intersection: this represents the number of intersections in the regions across which the subject validator operates

Algorithm 6: Update user

```
Input: profile, transactions
Output: profile
$dup \leftarrow searchDuplicate(transactions)$
$inCon \leftarrow countInconsistency(transactions)$
$forb \leftarrow countForbidden(transactions)$
$CommSize \leftarrow CommunitySize(Profile)$
$Evidence \leftarrow Multiply(Add(dup, inCon, forb), CommSize)$
$updateRisk(Evidence, profile)$
```

Algorithm 7: Update validator

```
**Input:** profile, Block
**Output:** profile
$intersectionFactor \leftarrow$
  $deduct(1.1, Divide(intersection(profile), MAXintersection))$
if $BlockNotValid(block) = true$ then
    if $checkDoggedness(profile, block)$ then
        $Evidence \leftarrow$
          $Multilply(intersectionFactor, Multilply(doggedness, size(validators)))$
    else
        if $Overlooked(block)$ then
            $Evidence \leftarrow Multilply(intersectionFactor, size(validators))$
        else
            $Inform(validatorProfile, block)$
$updateRisk(Evidence, profile)$
```

5.4.4 Node splitting

The usual communities expected to be detected within a social network are out of date. However, the splitting of nodes is usually based on a distance metric, in which the goal of extracting the distribution to overfit or adding randomness to suppress an expected outcome are the two options. In addition, a behavior tree that aims to model the system fails to handle dynamic iterative beliefs. Thus, TheTree is a dynamic solution with injected social behavior that turns decisions into a network to overcome previous philosophical limitations. Concepts such as “shop”, “sun” or “taste” can, themselves, be transformed into relations and studied in terms of interactions such as "taste", "hurt" or "credibility" to build a complex sequence of the infinite world of worlds

5.4.5 Relevance map and set prior belief

Interaction must take into account the transmission and reception from each entity separately. Each direction must guarantee “no noisy data”, which is generally considered in a probabilistic approach. The goal of Algorithm 8 is to build concepts and a ranked list of exchanges. In line 9, the splitting of the data into sent and received is combined in 10 based on the mean value. From 11 to 16, each interaction with a validator is represented in terms of a concept in which two relations will be built to model the direction. The value of surprise on the relational level is relevant to the exchanged value minus the value of malicious activities multiplied by the number of validators. The partner name is the peer's name. OutName, InName, and ConceptName represent the names of outgoing relations, incoming relations, and concept respectively, in which description will allow building more complex beliefs above them. Calling the function setconditional will be followed by calling the procedure conditional. Prior is relevant to the value of malicious activities. valueExchange represents the amount of the exchange made. The result is the substraction of the value from multiplying the prior by the number of validators before diving it over the prior.

Algorithm 8: Rank

```
Input: $data, conceptName, OutName, InName, DNSLedger, validator$
Output: $sorted, concepts$
procedure CONDITIONAL($DNSLedger, Validator$)
    $prior \leftarrow valueMalicious(DNSLedger)$
    $value \leftarrow valueExchange(validator)$
    $return\ Divide(Multiply(value - (Multiply(prior, validatorSize))), prior)$
$validators, received, sent \leftarrow splitdata(data, validator)$
sorted ← Based on the mean value combine sent and received before sorting it
for $validator\ in\ validators$ do
    $initiate(concept, conceptName)$
    $initiate(relations, Inname, OutName)$
    $set(relations, validator)$
    $setConditional(relations, Conditional(DSNLedger, (sent(or)received), Size(validators), validator$
    $set(relations, concept)$
    $setSurprise(concept, mean(sorted, validator))$
```

5.4.6 Relevance map and set prior belief

The centralization of a member within a society is an approach to characterize his behavior leading to the maximization of the gain to be conditioned by his relationships. However, each set of entities is an interchange region with gates to another parallel, intersection, or container world. The gates ensure the absence of dominance or build advanced knowledge.

```
Algorithm 9: Team division
   Input: $data, conceptName, OutName, InName, DNSLedger, validator$
   Output: $teams, concepts$
1  $Sorted1, concepts1 \leftarrow$
     $Rank(data, conceptName, OutName, InName, DNSLedger, validator)$
2  $set(removeHigherRanked(Sorted1), FirstPartner)$
3  $Sorted2, concepts2 \leftarrow$
     $Rank(data, conceptName, OutName, InName, DNSLedger, FirstPartner)$
4  $Remove(Sorted2, validator)$
5  for $element\ in\ sorted1$ do
6      $value1, value2 \leftarrow extractSurprise(concepts1, concepts2)$
7      if $value1 > value2$ then
8          $append(element, left)$
9      else
10         $append(element, right)$
11 $Add(validator, left)$ , $Add(firstPartner, right)$
12 $return\ setTeam(team, [left, right])$
```

The goal of Algorithm 9 is to build teams related to the trust from each peer to another based on the recorded ledger of malicious activities (the DNSLedger) and the portion of managed data. After Calling Rank at 4 and 6 to extract rank for the validator and its first competitor, 8-13 is implemented to assess to which side the trust is higher for validators to be associated. It will extract the surprise from the set concept. It will form a team associated with each of the two main competitors. Finally, it will be with the validators competing with each team in 11.

5.4.7 Tree building

An entity can build, with an ensemble of heterogeneous entities, a world upon different concepts. Many functioning worlds may be impossible, which means inconsistency, but due to the lack of evidence, because there is no complete existence of characterized entities, the world may flourish. Entities must ensure their knowledge is such that their world is consistent, and trust can be increased in it. In this way, there are financial, social, or biological gains to each entity where harm does not exist. Spotting and eliminating the malicious activities within the world lie in the members' instincts, driven by gains. The members that constitute a world conceptual community are defined by the characteristics of the world itself; consequently, a stopping condition is a very important element from a creational perspective.

5.4.8 Stopping condition

An organization driven by TheTree must have a stopping condition defined by the minimum number of components that build a world. The recursive construction of TheTree will be maintained until the basic world number is reached. The gates between worlds are not organized entities, but they are treated as parts of the regional system based on their exchanged value extracted from the data held by the validator. Algorithm 10 represents the recursive building. In the end, a sequence of leaves will be constructed, in which the further to the right of the main validator, the level of competence rises. However, at the community level, this means higher reputation destruction.

Algorithm 10: Build Tree

```
Input: $node, data, conceptName, OutName, InName, DNSLedger, validator, worldSize$
Output: $tree$
procedure SETTONODE($node, data, concepts, team, compititorTeam, direction$)
if $Size(teams) <= worldSize$ then
    $setIntersection(Concepts)$
    $setConscience(Concepts)$
    $setCommunities(Louvain(data))$
else
    $filterOutCompetitorTeam(transactions)$
    $setNode(concepts)$
    $buildTree(data, conceptName, OutName, InName, DNSLedger, validator, worldSize)$
$teams, concepts \leftarrow$
$Teamdivision(data, conceptName, OutName, InName, DNSLedger, validator)$
$setNode(concepts)$
$setToNode(node, data, concepts, team.get(0)("team"), team.get(1)("CompetitorTeam"), "left")$
$setToNode(node, data, concepts, team.get(1)("team"), team.get(0)("CompetitorTeam"), "right")$
```

5.4.9 Surf TheTree

The usual trick of society when a chosen force tries to apply a harmful interaction such as a high tax is to invest in a new chosen force. The distribution of force allows each entity to nest a client's directory, but if an entity acts in a harmful way with a member that has proof of its behavior, it decreases trust within that entity, which will drive the R up in the logistic map, driving the algorithm to act within its limits, and then periodically or chaotically to involve other forces that might be interested in overtaking the environment. Algorithm 11 describes the stage when TheTree leads the client to defend itself against malicious activities by involving other validators to increase the rate of deterrence.

Logistic map "(R×X×(1-X))" will be assigned the value of R that may be from zero to four. The value zero and one are considered separately. However, two will always generate a value under 0.5, which leads to the right, as opposed to the left, in which the validator has a normal path. The value of three will generate a value under 0.6, which leads the expectation to go right more than it goes left. However, the value of four will generate chaos based on the initial condition. At lines 4 and 15, the R-value will have incremental growth on each step relevant to the depth. The switch from 5-12 is to assess the value of R and act upon it.

Algorithm 11: surf TheTree

```
Input: $node, Validator, depth, R, step$
Output: $Validators, Communities$
$set(step, R/depth)$ if R!=0
if $round(step) == 0$ then
    $nodeSon \leftarrow nextNode(node, Validator, R)$
else
    if $round(step) == 1$ then
        $nodeSon \leftarrow nextNode(node, Validator, R, false)$
    else
        $R \leftarrow floorUpTo(4, step)$
        $value \leftarrow LogisticMap(R, Number)$
        if $value < 0.5$ then
            $nodeSon \leftarrow GetRight(tree)$
        else
            $nodeSon \leftarrow GetLeft(tree)$
if $nodeSon\ is\ null$ then
    $return\ node$
$step \leftarrow Add(step, Divide(R, depth)))$
$step \leftarrow Add(step, Divide(R, depth)))$
$surfTheTree(nodeSon, Validator, depth, R, step)$
```

5.4.10 Users community

Humans have many appreciated sins such as forgetting, and unappreciated ones such as unconsciousness; however, society has survived through solidarity, and this has been the basic engine of society, allowing it to flourish as a civilization, one in which a deterrent for any harmful behavior of one entity is to inform other entities of a change or to not cooperate, based on evidence which has led to the rest of the belief being harmful. The leaves of the tree contain validators attached to its client communities; sometimes a member will act in ways that are harmful to their environment and, at other times, to validators. One way to deter this behavior is to inform their community, in addition to other validators in their world. Algorithm 7 describes the process of reputation destruction.

The algorithm starts by first checking the type of the current executor. If it is a user, based on a sequence, it will update the rank first, then cut off inbound, and finally unsubscribe as a customer. The wallet will send messages to the nearest community of users and validators in line 4 and 5. It will surf the tree increasing the value of R and keep calling the surf function until it receives positive feedback or all options have been exhausted, it will filter by the highest trusted validators on line 8. However, if it is a validator, then peers will be notified on line 11 as well as the managed community at line 12. Finally, it will dynamically couple the section at line 13. The algorithm finishes by setting asynchronize function that will keep check if the first claim is falsified by a signed signature, then it will update positively for the validator and negatively for the source.

Algorithm 12: Reach Community

```
    Input: $User, validator$
    Output: $tree$
1   $RequestJustification(Validator)$
2   if $this.ID\ is\ User$ then
3       $Choose\ once\ a\ one\ from\ the\ sequence : update\ rank(), remove\ from$
          $inbound(), or\ unsubscribe\ as\ a\ client()$
4       $informCommunityValidators()$
5       $InformUsersInCommunity()$
6       $CallsurfTheTree(validator, R)$
7       $node \leftarrow HigherTrustedValidators(concepts, team)$
8       $Go\ Back\ To\ *Call\ surfTheTree\ with\ high\ R*\ Stoping\ condition\ is$
          $exausting\ the\ options\ or\ receive\ a\ success$
9   if $this.ID\ is\ validator$ then
10      $InformValidators()$
11      $informUserCommunities()$
12      $DynamicallyCouple(user)$
13  $Set\ rule\ if\ justification\ is\ provided\ rank\ is\ updated\ positively\ for\ validator$
      $and\ negatively\ for\ the\ source$
```

5.4.11 Dynamic layers coupling

The coupling between the two communities within graph theory has always been a high element of discussion within biological studies due to the importance to interlink between different biological worlds. However, within the police sector coupling has been used to interlink between the polices officers and the dangerous locations. The distributed world has many officers called validators, miners, or maintainers that suffer not only from malicious behavior of the clients that aim to stock the new information, but as well from their peers of the same service. Consequently, the dynamic criteria within the graph coupling are very important criteria to maintain the environment because there is a need to jump to another community aiming to secure a fast finality of transaction due to the high level of malicious clients. The other case is to attract clients of malicious validators to join a safe client directory. Following in algorithm 9 is a representation of a mechanism.

```
Algorithm 13: Reach Community
  Input: $Transaction$
  Output: $ValidatorProfile$
  $CommunityStructureliste \leftarrow$
    $surfTheTree(node, Validator, validatorNumbers, Transaction.receiver,$
    $depth, 0, 0)$
  $validatorProfile.communities.put(size, liste)$
```

5.5 Evaluation

5.5.1 Security discussion

The financial incentive is the driver of miners within blockchain technology. Paxos (Lamport 2001) and Raft (Clow and Jiang 2017) favored fault tolerance and safety to eventually secure a single state of the ledger, whereas Bitcoin favored liveliness and safety to secure to each node its copy of the ledger. TheTree switches the financial motivation from a tragedy of the common to a user's satisfaction to be the center of interest for maintainers. It preserved all previous advantages, but validators should not be anonymous so that they can be incorporated into the taxation system. In the case of anonymity, integrity is secured solely through the intersection's complexity. The approach is based on reputation besides open participation, which eliminates means of monopolization that leads, eventually, to manipulation. Moreover, anonymity with financial motivation based on a tragedy of the commons was the cause of skepticism due to the inability to punish in the case of a scam.

The left side of Figure 5.4 demonstrates the difference in data maintenance between PoW above and TheTree down. PoW is simply continuous competitiveness among different anonymous pools not interested in the safety of the user data, but seeking gain

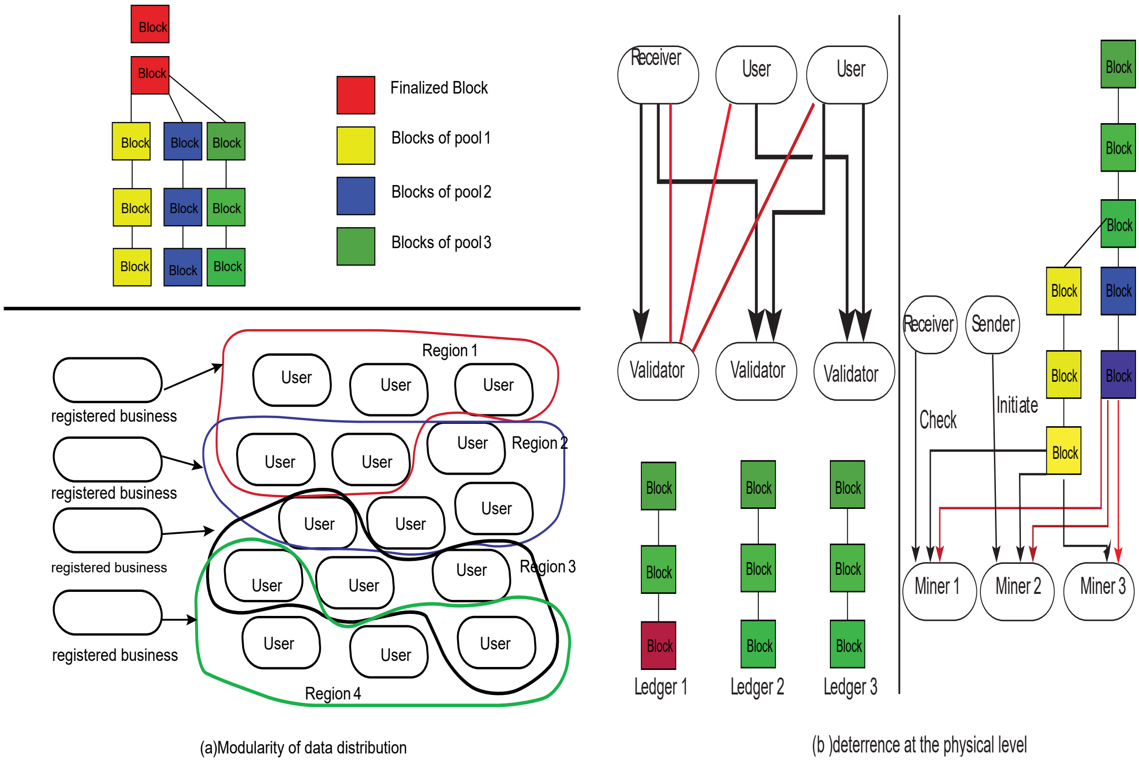


Figure 5.4: TheTree Vs Bitcoin PoW (modularity and deterrence)

from a unique ledger. Each one of the pools is in a race to force its version to achieve financial self-interests. However, TheTree approach focuses on the user as a center of interest as a client. Worldwide adoption with business registered validators will increase trust in the system through huge intersection complexity, besides police support against cyber-attacks coming from the tax benefits.

Malkhi et al. (Malkhi et al. 2019) introduced the concept of corrupt but alive (CBA), in which an adaptive quorum is a solution to maintain validity. CBA can take place conceptually in many other approaches, such as long-range attacks within PoS, leader targeting in improvised Bitcoin-NG (Yin et al. 2018), or the intention to fork within PoW. TheTree invests in the intention of validators to nest a client's directory. It will accelerate block propagation within the network and the validation time, unlike in previous works in a permissionless network. It is not a race to generate the longest sequence of hashes but an intersection of interests with validators that look at any newly generated block as updated information upon which a probable transaction may be based. Moreover, it is an increase of trust, not just compared to other peers, but compared to the government itself. The interaction for a peer is based on concepts generated from the Rank Function in Algorithm 4 to have a direct connection with territories of interest.

The right side of Figure 5.4 demonstrates the difference between the two approaches in terms of finality besides deterrence, in which the finality of the Bitcoin approach is probabilistically relevant to the number of competing nodes needed to force one ledger, which makes scalability in terms of miners positively correlated with latency. Moreover, post-deterrence is not considered. TheTree just aggregates data received from close validators in a certain order, motivated by probabilistic financial motive, and deterrence is maintained through an intersection before reputation destruction that will be applied by the transaction initiator.

The blockchain's transaction receiver may suffer from a double-spend initiated from the sender by using bribery (Liao and Katz 2017), or by being eclipsed by a monopolizing group. Conceptually, the problem lies in the incentive that encourages miners to search for rewards and not reputation. A DoS attack may be used to undermine the network and force double-spend. Thus, probabilistic finality has always been the most interesting concept in the system. The conceptual choice in this problem is the lack of trust with an anonymous entity capable of manipulation, especially in the case of many validators with the same time of block generation. TheTree uses the proposed model by the authors (Ikbal Nacer et al. 2021). This is a model of exchange that initiates a transaction from the receiver side by getting signed coins, leading the validator to be associated with the receiver for profit. Moreover, the deterrence of validators functions through the chaotic behavior of TheTree to ensure reputation destruction with close communities and the involvement of other competing validators.

A Bitcoin network allows for eight outbound connections and 125 inbound connections by default. Many researchers have investigated approaches to explore the topology to model finality time (Nerurkar et al. 2021). A neighbor discovery service is limited to extracting DNS seeds that represent an ensemble of miners. However, the ability to reconstruct the network virtually raises many concerns as it paves the road to many malicious activities such as RBG hijack, DoS attack, and eclipsing (Ikbal Nacer et al. 2021). The integrity within PoW consensus comes from the low pace of injection besides the distribution; however, without the centralizing MemPool, the finality latency will increase dramatically. TheTree allows a huge distribution as well, but the belief in the node is relevant to the rank, which puts the reputation to be a manager of connections. On the other hand, the authors have proposed a model (Ikbal Nacer et al. 2021) used by TheTree which uses mobile agents to exchange public keys between a receiver and a sender, thus empowering a hidden topology for the users. The validators' topology will be public, but as it is registered as a business, considerations of security measures will be practical enough.

5.5.2 Environment modelisation

A weak evaluation of the operating environment is provided in this section. The only purpose of this section is to provide a broader view for the reader to observe the proposed system from many sides. Probability theory is the art of describing the subjective interpretation that needs to be applied to decision theory to generate action. In all theories there are logical rules, and it is very important to clarify the difference between valid and right. Valid is a possible deduction based on the stated rules that have defined the set of propositions, whereas the right is the consideration of all aspects that define the real world. Following these leads to the valid being equivalent to the right. This section will start by modeling the blockchain environment and, more precisely, the world created by TheTree, in which the following sentence summarizes the functioning of the system: *"Integrity in the system is fostered by the majority of users satisfaction or the low level of malicious activities exhibited in it"*.

The space of validators is defined as complete, finite, and relationally atomic: X stands for a set of validators. S stands for a system, and Y stands for a set of users.

$\exists s \in S, \forall x_i, x_j \in X, validator(x_i) \wedge validator(x_j) \rightarrow independence(x_i, x_j, s)(i \neq j)$ (1)

Transforming the foundational sentence stated above to a rule, the assumption within blockchain technology is that user satisfaction is described in terms of the finality of its transaction, whereas malicious activities are described in terms of trust in the validators.

$\exists s \in S, \forall x \in X, y \in Y$ where $X, Y \subset S, Trust(x) \vee finality(y) \rightarrow Itegrity(S)$ Where:$Trust(x) \wedge finality(y) = \emptyset$ (2)

The concept of finality within blockchain technology depends on two intersecting concepts, which are the propagation of the transaction to validators and the integrity of the validators themselves, which means their honesty from the user's point of view. T stands for a set of transactions.

$\forall x \in X, y \in Y, t \in T, propagateTransaction(x, t, y) \wedge honest(x, t) \rightarrow finality(y)$ (3)

The concept of trust in the validators within the blockchain technology, and especially from TheTree perspective, depends on two intersecting concepts as well, which are the propagation of the block that contains the transaction and the reputation of the validators. B stands for a set of blocks.

$\forall x_i, x_j \in X, y \in Y, b \in B, probagateBlock(x, b, y) \wedge reputation(x_i, x_j) \rightarrow Trust(x)(i \neq$

$j)$ (4)

The regularity in probability is a rule which sets the background that all probabilistic propositional assumptions cannot be zero. Thus, each concept must be modeled probabilistically to define the background of the evaluation, in which the constant must manage the growth but must always assume the existence of dissatisfaction'and some malicious activities.

Based on the rule of general additivity applied in 2:

$P(integrity) = P(Trust) + P(finality)$ (5)

Based on the rule of multiplication applied in 3 and 4:

$P(finality) = P(honest) \times P(probagateTransaction \mid honest)$ (6)

$P(Trust) = P(reputation) \times P(probagateBlock \mid reputation)$ (7)

However, due to the philosophical argument of context applicability, the solution will just consider rules 6 and 7 to be a simple multiplication to secure the evaluation of the impact. The next step is to define low-level concepts such as honesty, the propagation of transactions, the propagation of blocks, and intersection.

$probagateBlock(b)$= $\frac{\frac{(\gamma \times size(b)}{MaxSize} + \frac{\delta \times intersection(i)}{regionsNumber} + \frac{\zeta \times power(i)}{MAXPOWER} - \frac{milicious(i)}{AllMilicious}}{3}$ (8)

$Reputation(i)$= $\frac{\frac{(\delta \times intersection(i)}{regionsNumber} + B_i \frac{\beta \times NumberOfclient(i)}{NumberOfUsers} + \frac{\varsigma \times Conscience(i)}{clientData(i)} - \frac{milicious(i)}{AllMilicious}}{3}$ (9)

$honest(i) = \frac{(\beta \times intersection)}{NumberOfValidator - \xi \times Risk(i)}$ (10)

$ProbagateTransaction = 1 - F^{receiversNumbers}$ (11)

First, the center of the study will be based on a rule (2), the aim of which is to observe continually with an independence each event and how the environment grows and maintains the community t to draw the boundaries of the system management. Second, the study will try to model and evaluate the real-life finality with growing and cumulative user belief toward the system by considering its factor within a delta time, in which rule 2 will be transformed to:

$$\exists s \in S, \forall x \in X, y \in YwhereX, Y \subset S, Trust(x) \wedge finality(y) \rightarrow Integrity(S)$$

$$Consequently : P(Integrity) = P(Trust) \times P(Finality)(12)$$

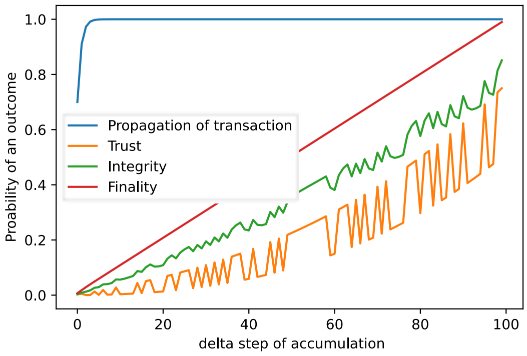
Rule 12 is deduced based on the same comment stated above regarding rules 6 and 7. Rules 8-11 have been concluded from the defined data structure of each profile, in which the validator profile that will be followed by his peers is based on the level of intersection, conscience, previous malicious activities, and the power of the used devices. Rule 8 defines the block propagation, which is normalized over three, besides defining the

Table 5.1: The definitions of the constants

Constant	Role
$\varsigma, \zeta, \delta, \xi, \beta$	Constant to manage the concept presented in the whole platform
MaxSize	The maximum size permitted for a block
regionsNumber	Number of regions to apply normality over intersections
MAXPOWER	Max Power to apply normality
AllMilicious	All malicious behavior in the system
NumberOfUsers	Number of users in the system
clientData	Get the number of submitted client data for validator i
receiversNumebrs	Number of peers to broadcast to
F	Setting the number of losses in the platform
B$_i$	Business or not (1 or 0)

most important components required to secure fast propagation. The speed of the block propagation is based on the size of the block, the level of intersection within the system normalized over the number of regions, and the power before deducting the malicious activities that have happened in the system. Rule 9 will again evaluate reputation based on the intersection level depending on whether the peers are registered as a business or not, before adding two intersection concepts, which are the portion of the clients from the system multiplied by conscience and finality, deducting again the level of malicious activities. Rule 10 will evaluate the honesty of the validator from the user's perspective, in which the level of intersection is the important criterion before deducting the level of risk. Finally, rule 11 evaluates the propagation transaction in terms of the probability of dropping a packet.

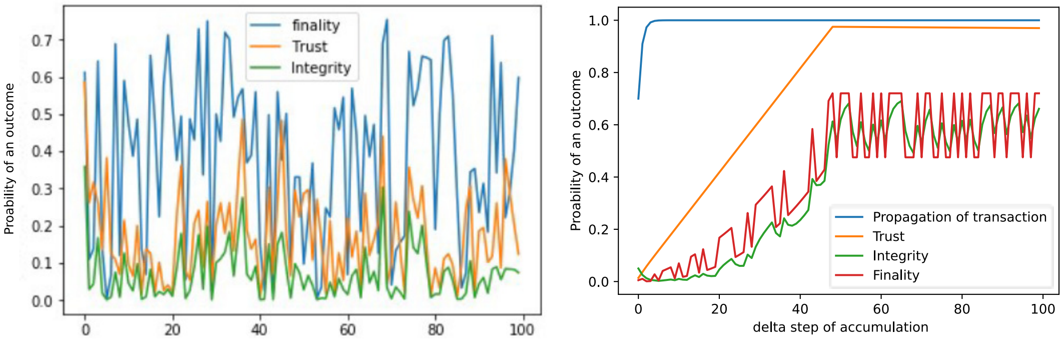
Figure 5.5 shows the growth of parameters against rule (2) with a highly independent event, which dictates the normal growth of the system over the long term. User parameters over trust in validators does not have the greatest impact on integrity. The fluctuation represents the random choice on the registered companies. This implies that in the long term, the system is not responsible for the satisfaction of each user but for ensuring a high level of finality. Thus, we can conclude that the system like any other institution is preserved as a global commutative stability built in a growing community that generates finality. A created object named region that contains methods such as immigrate, update parameter and chaos is set. It is embedded in a community object. a list of communities



(a) the growth of all parameters defined simultaneously shows more resilience to the platform.

Figure 5.5: Non-chaotic experiment with linear growth of parameters

will provide an example of the system. All variables were set to 0.99. All management constants were initially set to 0.01. Special variables such as risk and malicious are set to 0.1. The constants will be incremented slowly to the norm. From system point view, the figure is just a demonstration of how the variables that correspond to the structure of the network, which has been selected or imposed on TheChain, do not conflict.



(a) Chaotic behavior in a region due to malicious activities reduces trust and integrity and leads to immigration to the region in figure (b)

(b) Immigration has resulted in higher growth of parameters that force the conceptual aspect to grow

Figure 5.6: Experiment where finality impacts on trust to generate integrity

Figure 5.6 illustrates rule 12, where the intersection case follow delta time but is more focused on the long-term stress of the system by questioning the capacity for chaos. The longterm chaotic behaviour of validators and users will likely reduce the integrity of the system,mainly between 0.0 and 0.1 with the responsible region. The integrity will strongly depend on the continued value of trust and finality due to the commutative emotional feeling expected in case of chaos. Thus, objects were chained which represent the behavior with the infinity hypothesis on the number of these objects. Immigration is interpreted as higher growth of parameters. Thus, as shown in Figure 5.6 (a) and (b), the use of rule (13) resulted in the expected integrity convergence between 0.5 and 0.7 because it strongly depends on confidence in this choice, however, it is (a) the chaotic region that drops integrity to 0.1. The demo sequence tries to show how a delta time of chaos does not have a catastrophic impact on the system and to point out that there is a logical separation between the regions which eliminates the expansion of chaos because the trust of the users is associated with the relevant validators. From a system point of view, it is a demonstration of how the emotional effect that leads to the intersection of two concepts will result in a weaker view of integrity for the new community.

5.5.3 Formal study

TheTree functions over different components to maintain integrity as a final conceptual state. It must be made clear that there are two kinds of peer in the system, building two layers of topology. The user's side, in which a transaction initiator is a receiver, and incentivized by the intention to earn money. Consequently, reputation is very important to attract receivers to be clients. On the other hand, the validator has two kinds of incentives. First, the intention to force consistency with high duplication leading to credible finality. Second is the intention to inform through propagating information. The first criterion is met as a normal cause of the intersection of interests, in which duplication in order is in the financial interest of any validator due to the probability that future transaction fees may be based on it. The second criterion is met by the intention of the validator to finalize the interaction with the user to secure fees.

Figure 5.7 demonstrates the main activities taken in the validation session. First, the initiation of a transaction from the receiver side is broadcasted to the main validator and his regional peers. Then, awaiting with relevance to a capability of propagation, which is noted in the testing section. Finally, if the trust among validators and their regional peers is low, checking the exterior peers is an option, before inviting them for help in the case of intentional delay. However, the initiator is a receiver, and he holds coins as proof of transfer. If the region delay is intentional and may be associated with double-spending, the reputation will be updated. The states that can be happening in the system are the following: transaction initiation, user broadcasting, transaction holding (stands for lack of intention to share), peer involvement (assuming there is always someone that helps), updating the system, broadcasting the new state and, in the end, arriving at a transaction finality.

The system always intends to reach finality. The following is a description of the

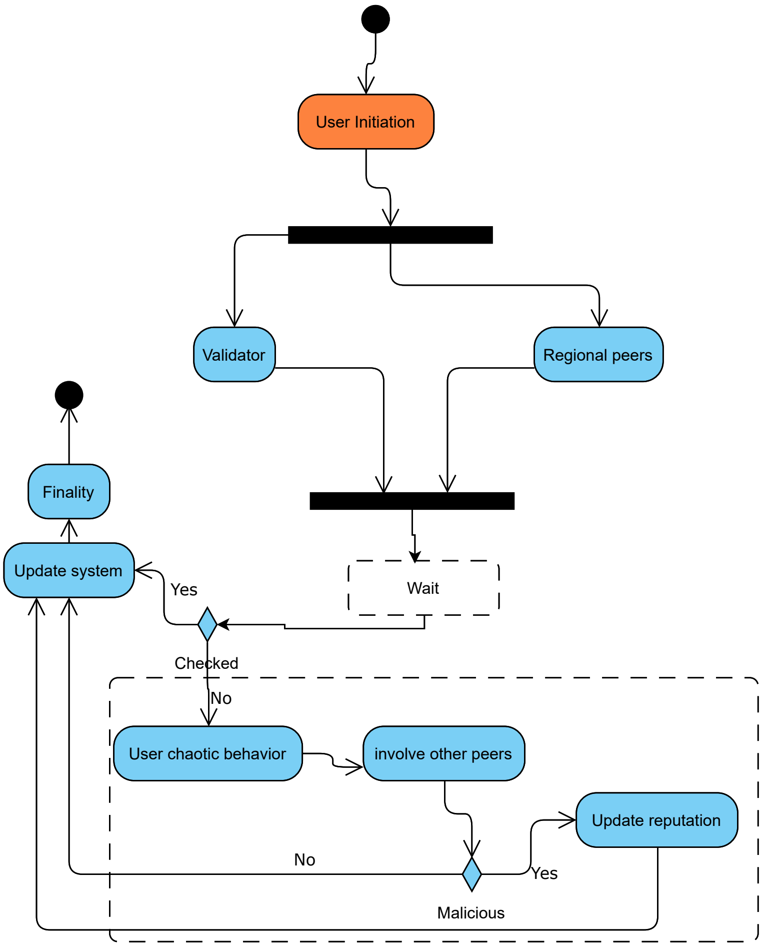


Figure 5.7: Activity diagram

proprieties that are involved in the transition among states. The state transaction initiation has the propositional rule that states: the user is satisfied. User broadcasting has a rule, which is that the validator is credible, the state transaction holding rule is a user who is not satisfied, and peer involvement means the reputation has been updated. Broadcasting a new rule means the validator is credible, and the finality rule means the user is satisfied. However, the temporal logic between states indicates that, eventually, there will be a finality. The next sequence is derived from the activity diagram. Figure 5.8 has been generated using the graph reachability algorithm.

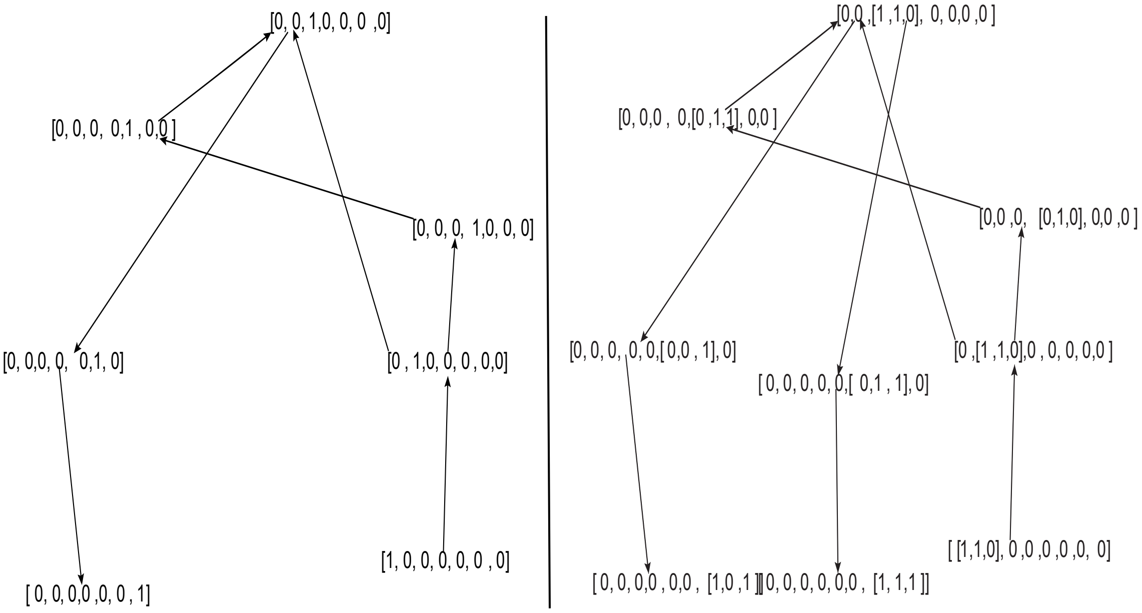


Figure 5.8: Graph reachability

On the left side of Figure 5.8, the states are transaction initiation, user broadcasting, system updating, transaction holding, peer involvement, share the new state, and finality. On the right side of Figure 5.8, the manipulation of attached proprieties introduced the intern vectors [user satisfaction, validator credibility, reputation updated]. As can be observed, as the assumption has been preserved such as there is always a validator to help with the high complexity of intersection, this will secure, in the end, the user's satisfaction as well as quick finality.

5.5.4 Comparison

Algorithmic complexity is a way to evaluate the algorithm's expected functioning by evaluating its worst and best execution. The following is a comparison of the system choices before dividing our approach in terms of deciding and dealing with malicious activities.

Table 5.2: Conceptual choices

	PoW associated technique	PoS Associated technique	IOTA Approach associated technique	TheTree associated technique
Finality Type	Probabilistic	Probabilistic	Probabilistic [6]	Deterministic/ Probabilistic
Information propagation	Gossiping	Gossiping	Gossiping	Broadcast among committee
Broadcasting complexity	O(nlog(n))	O(nlog(n)	O(nlog(n))	O(n)

The decision is a criterion that leads to finality, in which the PoW is a solution based on solving an NP problem by investing huge resources to generate a solution. However, the decision of finality is based on three components, which are the PoW complexity, the broadcasting complexity, and the probability of being the first. On the other hand, PoS inherits randomness, but in a different form, by making a random vote on the next validators before broadcasting, embedded with the probability of submitting a block. Finally, IOTA is based on a small set of NP problems before dealing with the probability of linking transactions above the latter, counting on the high level of submission. However, TheTree decision is based on surfing the tree to come to the knowledge of the validator's environment. Thus, the decision is based on the criteria of surfing complexity, broadcasting, and verification.

Table 5.2 demonstrates the information propagation choices within different proposals, in which IOTA, PoS or PoW platforms use a gossiping algorithm with complexity $(nLog(n))$. The tragedy of the commons incentive over the gossiping protocol leads to hard probabilistic finality. However, TheTree on the validators level uses broadcasting within the committee that has been generated through ranking. Therefore, it will be relevant to n in terms of complexity. TheTree finality can shift from probabilistic to deterministic with the relevant chaotic value of surfing it.

The only real competitor concept will be the PoW as other approaches fail conceptually to respond to many security criteria. The worst-case form that TheTree can take is to be the same as a decreasing recursive function. Consequently, it will have the following representation:

$$f(x) = \begin{cases} node, v \leq worldSize \\ T(v-1), v \geq worldSize \end{cases} \qquad \left\{ \text{It will lead the complexity to be O(v)} \right.$$

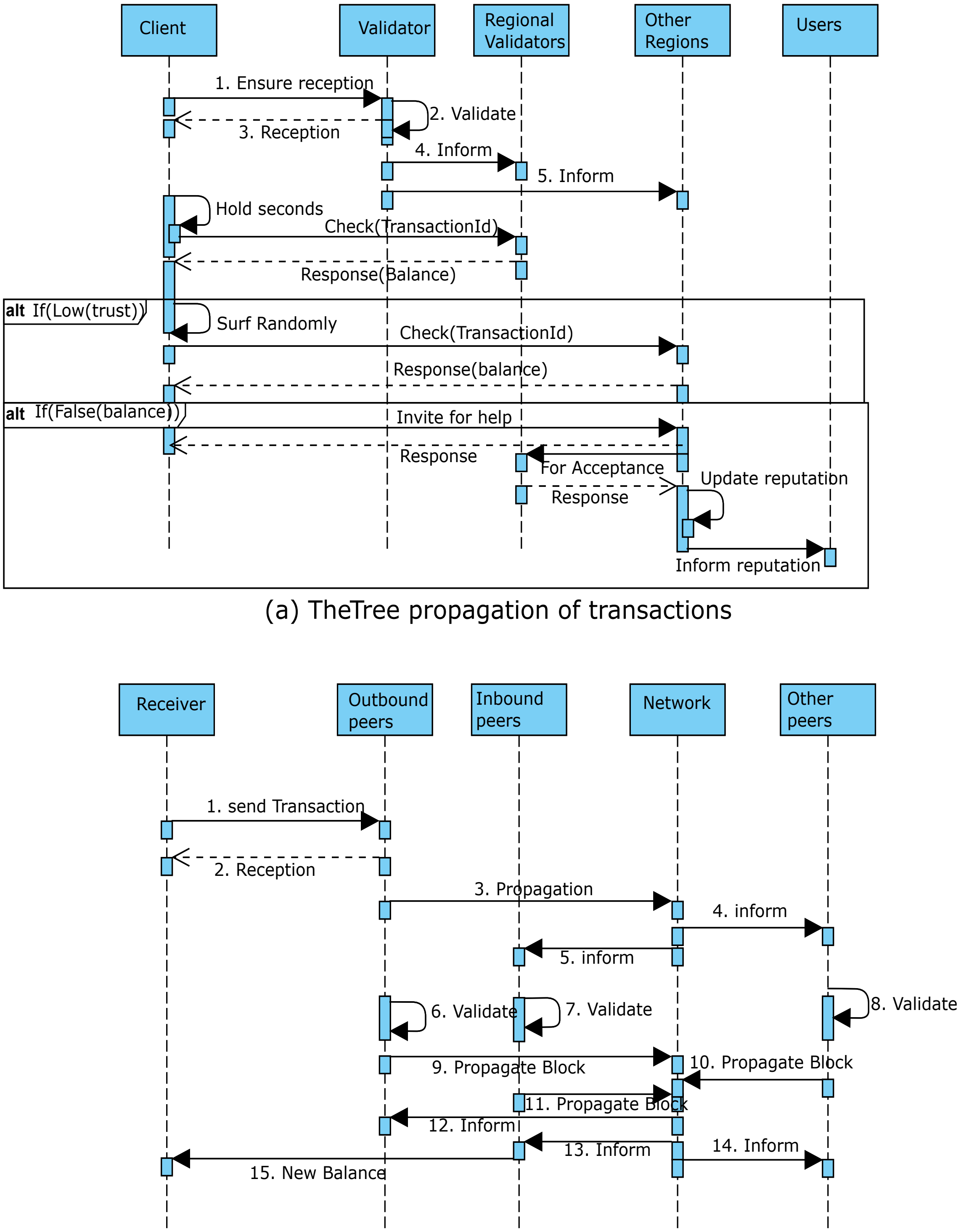
In which v stands for validator list size, worldSize is the limit that each conceptual world must contain on the low level, and a node is a data structure that contains all the saved knowledge about the validator and its environment.

The best-case scenario is when TheTree is well balanced, which leads the surfing to be smooth. The following is the representation:

$$f(x) = \begin{cases} node, v \leq worldSize \\ T(\frac{v}{2}) + v, v \geq worldSize \end{cases} \qquad \begin{cases} \text{It will lead the complexity to be(v}^2\text{)} \end{cases}$$

In leaderless blockchain approaches, all validators perform puzzle-solving, useful work, or random sleep. Thus, the solution can be described in $\exists L, b, c \in B, \forall v \in V, finality(L) = choose(generate(c, v, b), 1)$ in which $v, b, c,$ and L represent validators, block, processing capacity (Transaction per second), and ledger respectively, and choose will select a single version of a block from all the blocks generated from different related validators with relevance to their capacity. Thus, the level of processing of the transaction in a linear order can be described in $Traitement(t) = \frac{size(t)}{(sizeofblock)} \times Delay$. T represents a list of transactions and delay is the expected delayer for each Block separately. However, in leader-based approaches, pipelining and spinning are different options. Pipelining can be described as $\exists v \in V, L, b \in B, finality(L) = generate(c, v, b)$ where one validator is the block generator at a time. Thus, subjecting it to the capacity of a single validator described in $Traitement(t) = \frac{(size(t))}{C}$. However, the spinning increases the capacity (c) in the linear atomic order of processing, as the pipelining is subject to leader bottlenecks. Finally, TheTree allows generation from all validators at the same time with relevance for their client directory. Therefore, transforming it into $\forall v \in V, L, b \in B, finality(L) = generate(c, v, b).$. Will make the traitement processs to be $Traitement(t) = \frac{(t)}{C \times numberofvalidator)}$. However, TheTree worst topology structure performs the same as pipelining.

Figure 5.9 (a) demonstrates how the sequence of actions with TheTree approach is based on the receiver's persistence until the transaction has been injected. The other users serve as social punishment for the non-cooperative nodes. It starts with the receiver's intention to secure the fund, followed by different instructions for checking the system. Finally, it checks the trust in the regional validators before involving other regions until it makes sure the transaction has been injected with success. In the case of malicious behavior, the social punishment will be there through the reputation being updated. On the other hand, on Figure 5.9 (b), the transaction depends on the initiator,



(a) TheTree propagation of transactions

(b) normal propagation of transactions in the blockchain technology

Figure 5.9: Sequence diagram

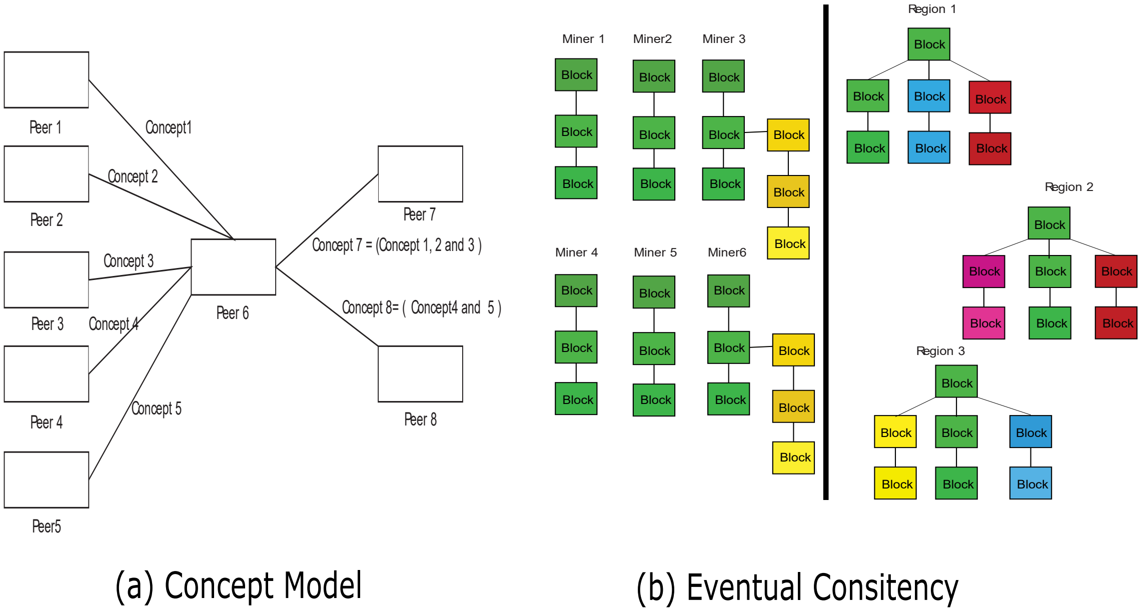
who propagates the transaction using a gossip algorithm that ensures its injection due to the expected zero collaboration in the case of a well-propagated transaction. It starts by sending the transaction to the outbound nodes that will be propagated in the network, which makes sure that all the nodes are aware of it. The different nodes compete over the block, then, with probabilistic finality, the turn will reach the transaction for it to be eventually validated. The receiver, as well as the sender, will be waiting for the upcoming news from their inbound peers. The choice among inbound is random with consideration of their reputation.

5.5.5 Conceptual comparison

Blockchain code is not well documented due to the high scale of adoption, which leads to different implementations. However, its architecture has been the focus of academic interest. Many studies have described the network topology, peer modularity, and implementation efficiency. The solution suffers from a software engineering perspective of an unmet legal requirement, low capability of testing due to its distributed nature, medium agility due to standards that have to be met for each peer to run within the environment, low ease of development due to its distributed nature that requires many network considerations and trade-offs, its scalability, coupled with performance, is subject to an eventually probabilistic consistency that defines the system as having low scalability, and it has low network performance concerning convergence. Thus, the concept of reliability is an important criterion, along with the short response latency, scalability, and modularity. Moreover, the solution must address market restrictions, such as legal compliance, a set of standards, and the high cost of its implementation.

Blockchain technology was dedicated in its first decade to the production of cryptocurrency and, due to its wide adoption, it has also been considered within the insurance sector, finance and government. However, modularity must be met to ensure the agility of the architecture to generate a system that can be easily adopted. It has been observed that systems such as Bitcoin, Ethereum and other implementations that possess a high coupling between the different components have low agility. TheTree has proposed the use of a new pattern to model the world into virtual computing components. The solution innovated away from the peer-to-peer pattern or event-oriented design but has built upon it to generate concept management between the two virtual peers on the distribution level of the concept-built regions, which can be an ensemble of concepts of the same type from different peers or different types of concepts. The left side of Figure 5.10 demonstrates the pattern which will allow flexible, controllable agility and maintainability of the system on the distributed level. A region of different concepts can be managed as a unique concept. The differentiation of this approach from the modelization of component-oriented programming frameworks such as OSGi, Corba and fractals is that security issues are related to the concept of a contract that focuses on the data structure and not information, as well as the middleware implementation that manages the service registration.

Reliance in blockchain technology is described as the capability of the system to serve at any time. However, the system's worldwide adoption with its financial gain is subject to horizontal and vertical scalability to ensure reliance. The scalability of the treatment of the transaction is subject to the CAP theory: in other words, consistency, availability, and partial tolerance. The legal requirement of business registration will allow different validator nodes to legalize their business in the system, as well as ensure a low level of malicious activity that eliminates an eclipsing or RBG hijacking, guaranteeing the concept of partial tolerance. The choice between strong availability and strong consistency has always led to strong availability and weak consistency within a delta time, before eventually achieving consistency. The right side of Figure 5.10 demonstrates the difference between TheTree and the Bitcoin approach, in which TheTree is expected to reach eventual consistency more quickly due to the lack of probabilistic finality related to competence over one version of each block but it is limited to the state of acknowledgment.



(a) Concept Model

(b) Eventual Consitency

Figure 5.10: Conceptual comparison

Scalability must deal with malicious behavior in the system. PoW, PoS, Tangle IOTA or BFT are all techniques that use either voting, resources or stakes to force the longest chain or path. However, the monopoly must take place following the longest chain rule. The concept that initiated the blockchain technology was PoW, which used complexity and randomness to deter malicious activities. However, a true elimination of the trusted party must take down the capability to monopolize the system. TheTree has taken a different approach, betting on the validity within a high intersection of interest among the different nodes. The following is the expected probability of maintaining a low consensus between nodes. $f = 1 - c^{(n(n-1)/2)}$ If the probability of the coming consensus between the two parties is: $c = 0.99$ it models the probability of coming to a consensus (c) to force a certain state with the ability to bring all other nodes onto the table in a deal that can be modeled with a complete graph. Thus, the probability of not coming into f is what remains of the space minus what is believed to be a consensus. The growth of the number of nodes n will diminish any deals due to exterior factors, such as legal compliance. Finally, the discussed concept provides a good background for setting standards of communication, which will later be the background for legal compliance. The capability to model the world through concepts will allow the easy integration of any component into the system.

5.6 Testing

The engine of economic growth is the connection between the human delusional evaluation of certain objects and his efforts. Guaranteeing the ownership of the object requires recognition, the finalization of the exchange and the securing of authenticity. On the distributed information, it can be translated into propagation, final consistency and deterrence of the system. This section is divided into three stages. First, the topological level addresses the impact of platform choices on its functioning by improving its expected propagation time. Second, the consistency level assesses the expected time of the exchange in a manner comparable to the growth of information generation. Third, the safety assessment relies on the convergence of actors in the event of chaotic behavior.

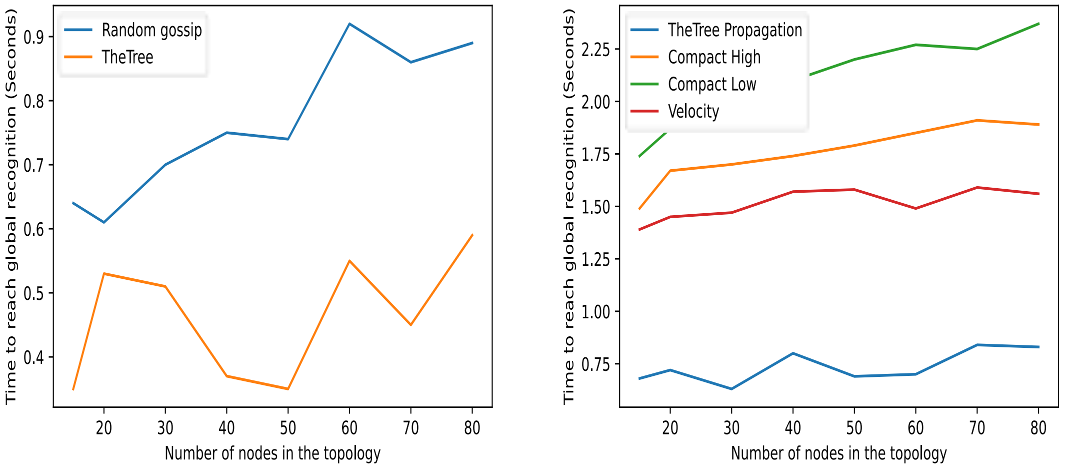
The blockchain's deterrence against double-spending is achieved by ensuring consistent duplication between many validators. For PoW, there is a delay until a winner is declared in a race leading others to adopt the version and start the same process over again. However, for technical reasons, users will be satisfied after a few more appendages in the ledger. Also, in other approaches such as PoS or BFT proposals, the finality is decided by the global attachment of the transaction. Thus, the techniques associated with the propagation of information followed by the logic ensuring an overall consistent finality are very important in time for comparing the operations of the platform from a user perspective. On the security, the high level of duplication and anonymity associated with PoW has led miners to continue racing as any intentional modification of previously processed information is very costly. On the other hand, BFT and PoS use severe penalties for deterrence. Thus, the evaluation of the cost of malicious behavior on the operation of the platform is an important factor.

The device used was a Windows 10 Intel 64-bit core i5 machine with a frequency of 1.8 GHz and 8 GB of RAM. NS3 simulation was implemented, 5% packet loss, data rate and delay were real for peers distributed virtually on six continents. Each link was managed with a socket. The block size was 1MB to 25MB and the transaction size was 1.2 to 2KB. Additionally, the Actor model implementation was used to simulate the distributed behavior of the runtime using the AKKA library in Java with Intellij as a development environment. Additional delays have been added to mimic an international execution. On safety, the actors are nested with a decision function and learn from the environment to act in a manner consistent with the protocol because of the high rate of deterrence. This shows that eliminating their cooperation will cause them to harm each other for financial gain and eventually force everyone to obey the law.

5.6.1 Topological level

Transaction propagation is the first element that takes advantage of the topology to inform all peers of the new knowledge that has been generated. Random gossip is the dominant approach for the propagation of transactions. Thus, it was evaluated in comparison to the TheTree approach. On the other hand, block propagation is the second data structure to be exchanged between maintainers. Therefore, the test has demonstrated TheTree and its comparison with the available solutions, such as high, low bandwidth Compact Block Propagation, and Velocity. Nodes are highly linked, in which each member has a unique collection of eight peers. . Estimated time based on an increasing number of nodes and blocks varying between 1MB and 25MB.

On the left side of Figure 5.11, Random Transaction gossip performs poorly against scaling due to the growth of duplication, but TheTree uses source routing to broadcast the transaction to other peers for the pre-verification. In addition, regarding block propagation, TheTree's performance is due to a direct link between the interested parties and a geographical consideration at the topological level compared to other approaches which



(a) Increasing number of nodes within highly coupled topology to demonstrate the expected linear growth to reach global recognition of a transaction

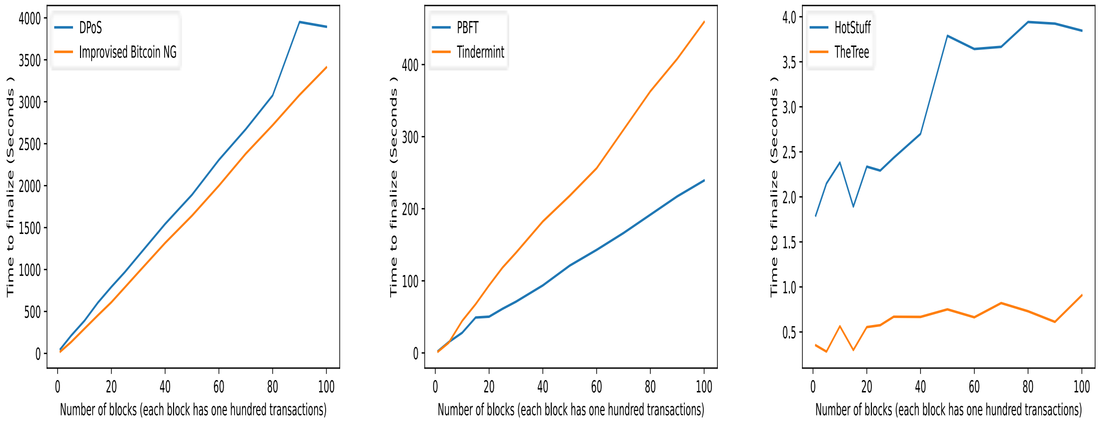
(b) Increasing number of nodes within highly coupled topology to demonstrate the expected linear growth to reach global recognition of a block

Figure 5.11: Propagation time

use a random flat topology to offer a vision of anonymity next to the level of exchange. As previously stated, TheTree first submits the transaction for pre-verification, then upon receipt of the signed commit, it will submit a block containing the transactions previously pre-verified using source routing. This allows the system to take advantage of the high performance expected of the topology. In addition, scaling will not be a problem as consistency is seen regionally rather than globally. Eliminating double-spending requires rapid dissemination of information. TheTree's architectural choices make it the most efficient approach to meet user expectations due to very low linear growth for time propagation in the case of a higher number of nodes and blocks.

5.6.2 Consistency level

The logic to be achieved before declaring finality results in a delay for the retrieval of proof of submission, which leads to manipulation of many layers such as leader attack, RBG hijacking, or DOS attack during one of the required steps. However, adding transactions to the general ledger of all peers requires an order. Entering into a world order dictates reaching the finality. BFT approaches, which use an authorized environment, hence the PBFT pipeline, and spinning or a combination of the two approaches have been used as different conceptual solutions to increase throughput with impact on finality. Moreover, a solution such as DpoS, improvised Bitcoin NG or Tindermint has linear growth due to the need for one version. However, TheTree had to focus on the transaction, not the block order before submitting the order based on an invitation provided by other validators.



(a) The increasing number of blocks in DPoS and Improvised Bitcoin-NG compared to the time required to finalize the logic to ensure consistency

(b) The increasing number of blocks in Tindermint and PBFT compared to the time required to finalize the logic to ensure consistency

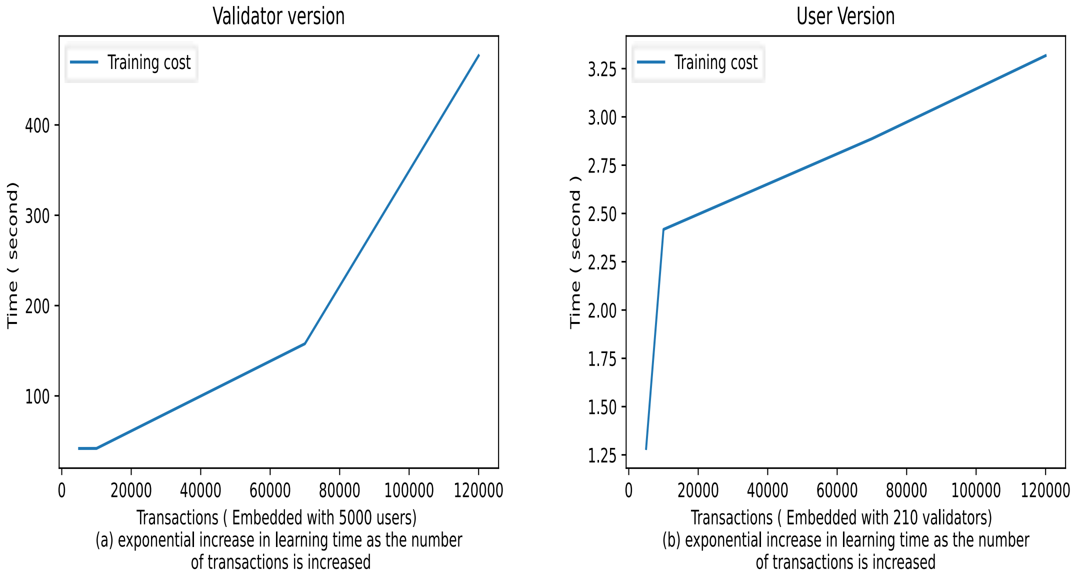
(c) The increasing number of blocks in HotStuff and TheTree compared to the time required to finalize the logic to ensure consistency

Figure 5.12: Finality and deterrence

Figure 5.12 is a demonstration of expected runtime performance drawn from many sessions of an actor model trial with a random selection among delay and topologies. Hotstuff's high performance is due to the use of PBFT pipelining within an expected permissioned environment. Tendermint uses spinning and the order uses a combination of PBFT and PoS to reach consensus. The downgraded DpoS (Yang et al. 2019) is the worst after pure PoW due to the use of a lite version of it for the selection process before voting that end of comparison of blocks, but improvised Bitcoin-NG works a little better due to the direct random selection process. The security assumption for execution makes Hotstuff better considering the requirements. TheTree performs best overall as deterrence is turned into a network, forcing users to submit authentic transactions and validators to force the acknowledged expected order of it. Therefore, it eliminates the probabilistic finality arriving with the blocking order and eventually makes the consistency subject to acknowledgement. The logic ensures reliability on the user side because the proof of reception is a set of registered businesses signatures. In addition, it is the fastest in terms of requirements to propagate and complete the transaction.

5.6.3 Unit Test

This subsection will present the training and experience required to assess the reputation of management as part of system scale-up. As demonstrated in Figure 5.13, the growing cost of training in the validator's version is exponential due to the use of a greedy search algorithm for community detection. However, on the fly, detection by dynamic community association will be used, which will be less expensive but, for new validators, the community must be detected through training of TheTree. The data used for training is Global Trade (Bank 2022), in which countries represent validators with fake users generated for each transaction.



(a) exponential increase in learning time as the number of transactions is increased

(b) exponential increase in learning time as the number of transactions is increased

Figure 5.13: TheTree training

System of an actor model that has 210 validators and more than 2000 users. The system ease of operation is expected due to the registration of validators as businesses. Therefore, any malicious behavior is reflected on a state internal security system. However, in this test, the hypothesis is based on the possible realization by anonymous validators to demonstrate the cost of malicious activities on the validators and to explain that the logic loop explained previously will force each actor to act honestly because there will be a high rate of deterrence.

Each actor is implemented to seek its interest by aiming to maximize its gain. A validator has a decision object that chooses to act based on preset likelihood between malicious and honesty based on the size of lost and gained users. It is injected with Ranking Network (Spohn 2012) which has four concepts to maximize the gain. It contains the malicious concept which has two links, one to gain customers and one to lose customers, and both then lead to financial gain. If a validator has not received a request for justification, information about a ranking update, or unsubscription from a client for their withholding of a transaction, they will update the malicious act as a positive behavior. However, negation will update it negatively. On the user side, any logically incorrect information, whether in the metadata or the data itself, will also result in a user update. However, the focus will be on validators as they are the managers of the validity.

Running the same parameter a hundred times with a random choice of users in a fraction of ten seconds to initiate a transaction showed that the cost of malicious activity on validators forced them to act honestly after continually updating the ranking function. The malicious act, which is not followed by environmental action, is considered financial gain. However, before these actions are taken, users ask the validators to justify themselves. After a while, the system stabilizes as the high deterrence rate coupled with strong peers connection led to the update of the ranking function. The convergence time is eliminated because it was highly dependent on a different initiation (updating the ranking with relevant ones for community members, validators, or topology link).

The protocol of communication between peers is based on the exchange of many kinds of data structure, represented in the following :

User Protocol data structure

$\rightleftharpoons \Uparrow Transaction$

$\rightleftharpoons \Uparrow Request\ Proof$

$\rightleftharpoons \Uparrow Request\ Registration$

$\rightleftharpoons \Uparrow Request\ Rank$

$\Uparrow$RequestLink

$\Uparrow$RankResponse

$\Uparrow$ResponseUserList

$\Uparrow$RequestNeighbor

Validator Protocol Data Structure

$\rightleftharpoons \Uparrow Transaction\ Received$

$\rightleftharpoons \Uparrow Provided\ Proof$

$\rightleftharpoons \Uparrow Registration\ Validation$

$\rightleftharpoons \Uparrow Rank\ Response$

$\Uparrow Request\ Block$

$\Uparrow Requested\ Block$

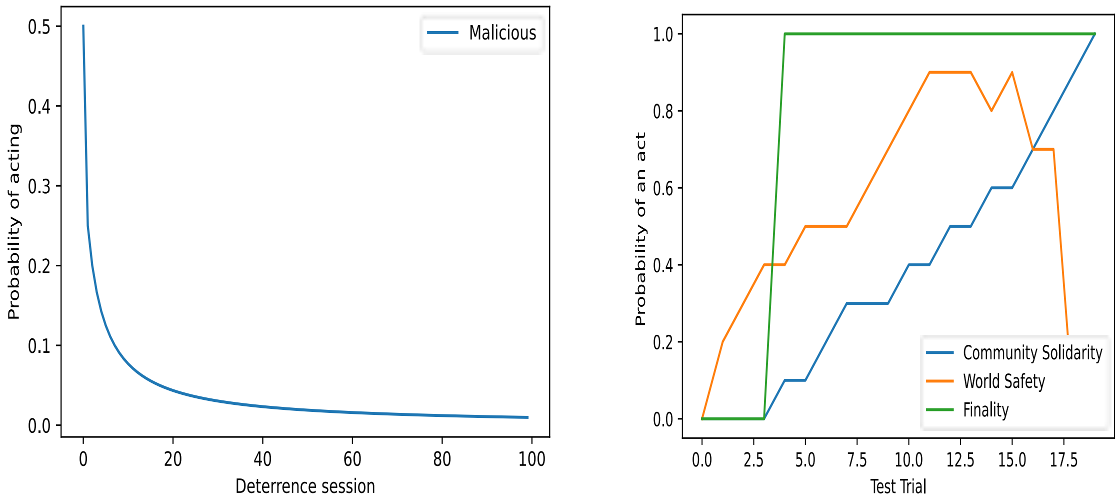
$\Uparrow Request\ Ledger$

$\Uparrow Requested\ Ledger$

$\Uparrow Peer\ Validators$

$\Uparrow Peer\ Registration$

The different data structures exchanged by the two components will be responsible for the exchange of data, which can be in many forms of wrapped data within a transaction. The symbol $\rightleftharpoons$ stands for a data structure that is exchanged between parties, whereas $\Uparrow$ stands for data structures that can be exchanged with the same type of components. The data structure will contain information related to the exchange of data for a user-centric benefit for validation, or it will be related to managing good knowledge about the community of existence.

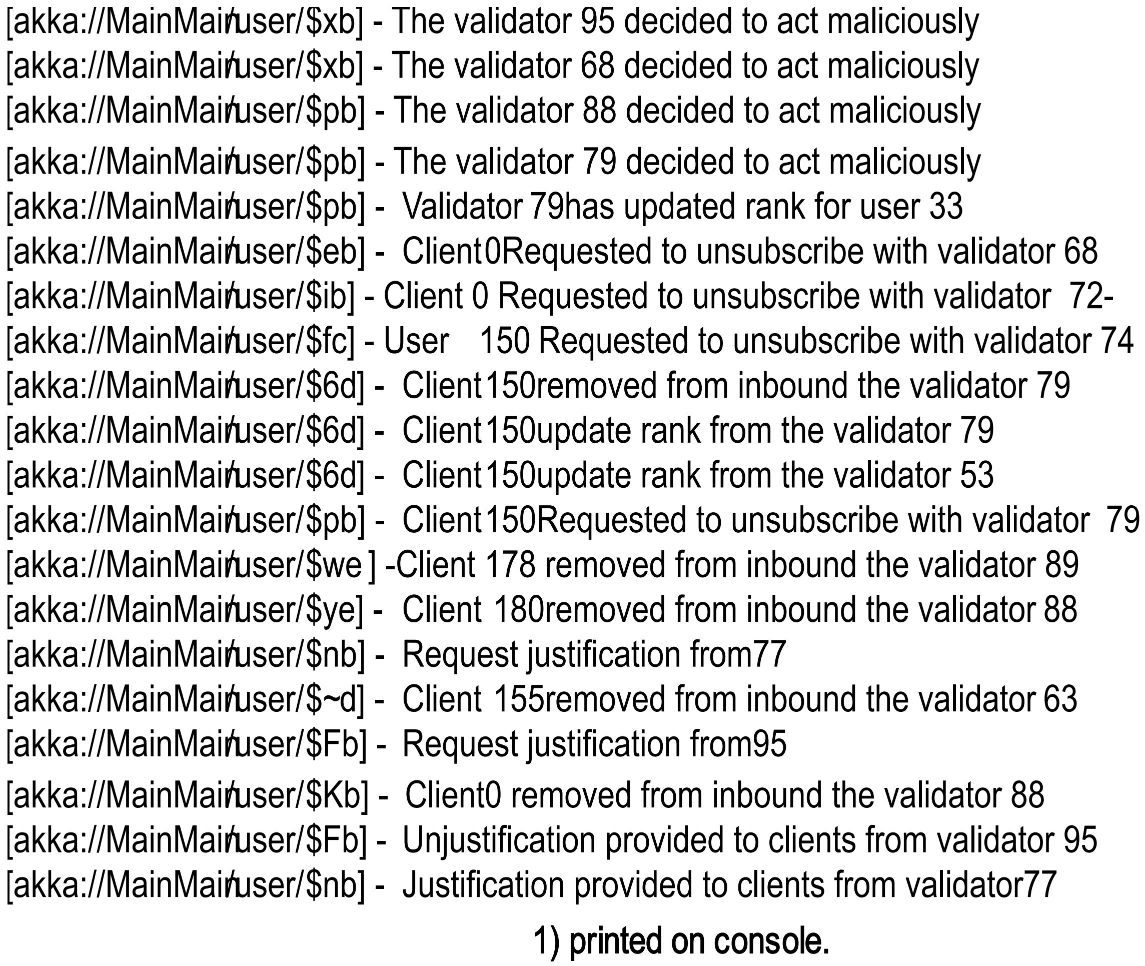


(b) Updating the decision-making function due to financial losses

(c) Finality comparable to the evolution of the number of malicious activities among users and validators in the same world

Figure 5.14: Peers managements

Figure 5.15 on the left shows the actions printed from actor interactions, in which rank update, unsubscribe request, and cut link are different options for customers. However, for validators, the rank update is the main option that prevents the validator from transmitting the signatures of the most malicious validators as proof of recognition to the users, which leads to preventing the extension of their scope of action. Figure 5.14 on the right is generated by manipulating the parameter of several malicious members in each user world. This shows that the level of malicious members within the community, as well as the security provided by validators within a world, is not important as long as there is at least one path to deliver the message to certain users, which led to churns translated deterrent and updated the ranking function. The graph represents many trials with a different set of community solidarity and global security, which represent the number of cooperative users and validators respectively. Stability is achieved when the level of maliciousness is very low after numerous deterrence stages, in which validators only aim



```
[akka://MainMain/user/$xb] - The validator 95 decided to act maliciously
[akka://MainMain/user/$xb] - The validator 68 decided to act maliciously
[akka://MainMain/user/$pb] - The validator 88 decided to act maliciously
[akka://MainMain/user/$pb] - The validator 79 decided to act maliciously
[akka://MainMain/user/$pb] -  Validator 79has updated rank for user 33
[akka://MainMain/user/$eb] -  Client0Requested to unsubscribe with validator 68
[akka://MainMain/user/$ib] - Client 0 Requested to unsubscribe with validator  72-
[akka://MainMain/user/$fc] - User   150 Requested to unsubscribe with validator 74
[akka://MainMain/user/$6d] -  Client150removed from inbound the validator 79
[akka://MainMain/user/$6d] -  Client150update rank from the validator 79
[akka://MainMain/user/$6d] -  Client150update rank from the validator 53
[akka://MainMain/user/$pb] -  Client150Requested to unsubscribe with validator  79
[akka://MainMain/user/$we ] -Client 178 removed from inbound the validator 89
[akka://MainMain/user/$ye] -  Client  180removed from inbound the validator 88
[akka://MainMain/user/$nb] -  Request justification from77
[akka://MainMain/user/$~d] -  Client  155removed from inbound the validator 63
[akka://MainMain/user/$Fb] -  Request justification from95
[akka://MainMain/user/$Kb] -  Client0 removed from inbound the validator 88
[akka://MainMain/user/$Fb] -  Unjustification provided to clients from validator 95
[akka://MainMain/user/$nb] -  Justification provided to clients from validator77
```

1) printed on console.

(a) Different actions taken in sequence within the deterrence sessions to force the validators to act honestly

Figure 5.15: Peers managements

to act honestly. In addition, the updated ranking function quickly escalates but requires a lot of testing to eliminate the maliciousness. The malicious line represents the decreasing actor-level probability of acting maliciously, as it is set to 0.5 and rated with relevance to the gain and number of registered users.

5.7 Future work

This work has introduced a concept model to respond to modularity, agility and increased scalability in terms of a flexible injection of a new component that manages new kinds of information. Simultaneously, it is recommended as a new approach to reasoning. Each built world is managed through reputation, and belief is attached to a distributed entity that manages the concept. Moreover, TheTree is a data structure distributed in the network to provide knowledge about its structure in terms of a world driven by reputation. The following is a list of directions and the future work that needs to be studied:

1. As the proposal aims to adapt to a user-friendly legal system, the study of the injection of state security representatives into TheTree will be studied in a way that preserves user privacy and business transparency.

2. Observing the web in terms of reputable possible worlds can be useful for consistency of information, but the price of isolation is injected. So, it is important to study the user side as a scaling manager, in which users support many validators that handle different heterogeneous / homogeneous concepts will increase competence and mistrust between validators within the business model.

3. Switching decisions from computational components to a network can be followed by considering the user moving from an observation item in simple static terms to a rule generator. The rules will be recorded in different areas of activity, represented in the transaction, to then be explored using algorithms that simulate human behaviors such as kindness, greed or decoding.

4. A node discovery requires the study of the concept of the prior in an open context where a hypothesis on a concept managed by an entity is relevant for the reputation of its world or more. Each node must be seen for a new eye as renowned as its surroundings.

5. Explore more reputation metrics. It can also be a user-generated rule.

6. The approach will be proposed to be implemented at a university to offer students double-blind management and generate tests in real-life scenarios.

5.8 Conclusion

This work has introduced TheTree to provide a structure for approaching validators at the top of a system that increases competence through reputation. A sociological ideology has been injected into the system to deter validators. The concept model is the key to horizontal growth and modularity in the system. The whole system is seen as a new kind of web where consistency relates to the digital world of existence. However, authenticity is a matter of necessity in all worlds. TheTree algorithm has been demonstrated, in which reputation is managed through defined criteria with relevance to the community and validator numbers. Moreover, competency at the team division is about choosing a world where validators have less trust in each other. On the associating nodes, it will build a sequence of deterrence by which it will involve finality as the major competitor with high trust. Users will be able to deter users through a random invitation of other validators into the world to execute reputation destruction in case of a validator's misbehavior. The approach has been studied in terms of a security discussion, environmental modelization, a formal study, and a conceptual comparison. Finally, simulation in NS3 and the Actor model has been implemented and compared with some models published earlier. The paper can be summarized as follows:

1. Discussion of the reputation-based network
2. Introduction of TheTree Algorithm
3. Introduction of the concept model
4. Theoretical and empirical evaluations have been demonstrated to show the outstanding performance of the proposal

6 Conclusions

6.1 Discussion

Banks are an institution between depositors and borrowers in order to guarantee liquidity. Thus, providing investors with the ability to quickly move state rewards, i.e. money, is an important attribute for faster services. The digitization of money offers investors, banks or the state itself the ability to undermine the problem of the agent of external trust in the institution itself. In addition, banks will be able to drop the emphasis on routine instructions. It has been demonstrated with TheChain specification that the goal is to speed up validation, provide traceability of information, enable scalability and increase reliability. Moreover, at TheCoin level, fuzziness has been used to link different transactions attached to the same identities. Identities can represent different state-registered businesses. This will allow states to enforce taxes in an efficient form. Moreover, it will be the basis of traceability to eliminate the manipulation of information value, which is the most used tactic to launder money. Confidentiality is very important for the security of the award owner or the institution itself. Thus, using a mobile agent as a tactic to exchange keys with a ring signature providing users with more capabilities to generate keys allows for a high level of privacy.

The generation of tangible cash from central banks has many disadvantages, such as being prone to counterfeiting, the lack of traceability that leads to easy manipulation, which can be used for fraud or money laundering. The idea of signed data instead of paper can be a solution to full digitalisation that can solve many real-world problems besides securing parallelism of execution. Moreover, it lowers the fees and can be executed under the bars of the expected World Bank goal, which is three percent. The authenticity of the coin can be the same as the transaction, which lies in the digital signature. The first depends on the issuer of the coin, and the latter depends on the validator of the transaction. The proposed model is to sign all first issued coins from the central banks. However, the system may later be in need of the issuance of a new coin to finish a transaction, but each new coin is signed by the validator and linked to the previous one to ensure traceability. An advanced vision to it is to provide a traceability to its generator to allow later an advanced search to check the association between the coins and information. TheCoin as a standalone model can be implemented without any other components. It will provide a bank/central bank with the ability to process transactions in parallel. Eliminate money laundering as each coin spent will be associated with a piece of information taking advantage of coin traceability, the information price is fixed or predictable. Provide a new business model for the bank, where the customer can pay per transaction in exchange for high confidentiality.

The trade-off between internal parameters was common practice to manage consistency, safety, and liveliness. However, consideration of the distribution of components to be managed by different human beings must consider the social factor of interaction. Thus, the proposal introduced a new business model under which access to business within the financial sector will be easy. On TheCoin level, the introduction of the authenticity of the coin leads to traceability to its original. The origin can be the central bank as representer of the state or another trade institution. However, at TheTree level, each validator must be registered as a business just to maintain, whereas all trade and exchange are withheld through dynamic implementation and setting by clients. Banking as an investing institution is not threatened by this model but it will make their business faster and encourage many depositors to invest.

The state as a bureaucratic institution suffers from services that can easily be automated in the legal sector, technical records and accounting. The ability to duplicate data with a secret identity provides citizens with the expected transparency. Additionally, contracts are rules that have been put in place to be applied as entry requirements are met. TheTree takes a sociological approach to ensure competence. It will provide new services and facilitate access to financial management as a business. The state itself invests in the confidence provided by the system to delegate routines to the machine and focus on very advanced tasks. Brokers in the form of insurance and market participation goods can be directly replaced by direct contact between citizens and businesses. However, the need for information at this level is highly required. The traceability of TheCoin model is conditioned as explained in TheChain by building a Petri net to control the modulation. The transition is the rule to apply to the value of the Wallet. Thus, it has been flexibly modeled to contain more advanced symbolic elements.

Architectural choices are a game of playing with computing principles to have an impact on performance and attributes. The architecture of the distributed system varies from centralized to completed distributed entities that collaborate. However, code organization to communicate and calculate is called architectural pattern. Our approach proposed the usage of a new model within the distribution of information. It is different from the basic foundational architectural that address flow of data such as event base or object base can be found in CORBA 2, but it addresses the highest level at which information is organized such as Fractal, OSGI and CORBA 3. Corba uses component static learning and takes advantage of the ORB for dynamicity. OSGI uses a public registry to share new services. Fractal architecture that can take many forms of implementation takes advantage of services as means of connections between components. However, the concept model introduced in TheTree allows a high level of flexibility and dynamicity for different components. However, each concept is considered as a single flow of information that can be aggregated and contained by other concepts. Thus, application is built by connecting to other services and built up on them other services

Validity as a concept requires consistency among the different distributed entities to declare finality and ensure the non-reversibility of decision. Thus, since the eighty many algorithms have been proposed to serve as a consensual mechanism. Leader, leaderless or random BFT and other approaches such as PoS focus seldomly on server-side machine as way to force consistency. However, seeking an internal control of information flow will delay finality that due to global consistency force the regional and not the revert. PoW based its assumption on the low level of trust between validators to ensure infinite race over securing the highest number of blocks within the longest version of the ledger by each miner. However, it again makes the system another trusted party managed by many machines. Thus, node independency proposed in TheChain approach is because competency first basic attribute is autonomy.

TheChain objective is to build self-validating nodes that are enabled with a layer of validation for fast and parallel treatment of transactions. The network is governed by an algorithm that builds intersected regional maintainers. The proposal dropped consensus, which means the absence of convergence on a unique ledger. The normal function of the system is to set a limited number of coins with unique identifiers that will be exchanged between different users. Maintainers will operate in their region to make their customers' transactions public in exchange for a reward. It is in the interest of all the nodes to be up to date with the different exchanges to eliminate any fake coin generation. However, all nodes will not be recording all transactions due to the limited resources, but the closest regions keep up to date with the ledger of the next regions' ledger, to build a complex sequence of regions that watch over the next to ensure integrity.

The nodes are independent of any exterior dictation of data, consequently eliminating any double-spend or fault injection of data that have a high impact on the network as a whole. Moreover, nodes operate in regions that lead to the elimination of any attacks that target the network liveliness.

Deterrence is the main concept to ensure security. Thus, making the cost of malicious act very expensive is also the foundation of the network. It has been implemented by many distributed systems to preserve the global truth from adversary's manipulation. Bitcoin PoW inspired from hashcash PoW usage to deter spammers from sending emails by making it hard for a malicious user to replicate a long sequence of atomic blocks difficult to be generated. PoS and BFT approaches use hard penalties to eliminate any stake owner by slashing their stake in the case of misbehavior. The variant of distributed systems implemented within the sector uses reputation-based security to eliminate DDOS attacks coming from certain IP addresses. However, there is need for a dynamic solution that observes the world virtually as a normal social behaviour. Thus, the proposed approach aims to adopt human civilization behaviour to grow into many safe groups. It can be divided into three important concepts, which involve the users as a part of the punishment system, increase the division of power, and reputation is the most important factor of connection. The solution has changed security from being a component of calculation to a network that applies reputation destruction as a consistency forcer

The absolute truth on human management of information is just a delusion because perception of reality is different due to cultural, ideological, and practical differences. Thus, observing the world current institutions as competing hierarchical regions to secure validity is easily to be adopted within a distributed system as well, as it will be of human nature to be in groups. However, each group contains a basic belief to manage, which is non-inferential for them. An ensemble of basic beliefs through the intersection of regions of interest can build a new belief to be the basic within that world. Considering the realworld absence of absolute truth to be the foundation construction for belief and be later a subject of decision. The inference function on the reception of a symbolic element will generate a set of beliefs.

Modelling beliefs in terms of concepts makes reasoning divide the process of inference into concepts, rules of transition, and motif. It can be observed in human theory as presumption, natural law and soul motif. On TheChain, the motif is inherited from users as they will be the manager of scaling. Users can be the generator of rules of transition by the initiation of different contracts and conceptual belief is built as an interconnection of

Table 6.1: Tables of variables for validity

Variable	Stands for	Variable	Stands for
D	Data	N	Node
T	Transaction	M	Model
Variable	Standing for	Variable	Standing for
L	Number of level 1 intersection (among validators)	C	Probability of consensus among two nodes
R	Number of level 2 Intersection (among users)	N	The number of nodes in a community
M	The level of misconduct in the community	T	Transaction

different businesses. Connecting all those elements to function in an orchestration within a competing world of worlds can provide a virtual model to nature. Moreover, flexibility regarding logical consistency in the system with ensuring an open context of functioning provides decision as a network. Thus, unlike previous work that considers decision as a component of the calculation, this work provides a dynamically growing platform to be the basis of future management of decisions to automate many intellectual tasks besides being a case of study as an inference platform as AGM.

6.2 Validity

1. The deterrence in the system is through the reputation destruction mechanism, where u stands for a list of users and v for a list of validators. $\forall v \in V, u \in U, misbehavior(v,u) \rightarrow reputationDestruction(u,v)$

2. The model uses all the cryptographic conventions of a blockchain system. To fulfill the legal requirement, the validators must be registered as a business. Validators will be deterred by legal cross-border compliance before dealing with a huge number of competitors that want to take over their client directories. The deterrent mechanism applied by the nodes is the reputation destruction of the validator through providing nodes in the community proof against each malicious behaviour. Validity is driven by the no consensus on a single version, but the authenticity and traceability, in which there are more intersections across the region, will secure the system. TheChain system solution is to find the balance between the actual reality and the target vision, in which its goal is to provide a no consensus approach that overcomes the monopoly and provides high validity.

3. The definition of validity within a distributed system can be seen as the authenticity of generated data, the traceability of its origin, in the case of currency to make sure

that all coins are associated with an information to prevent manipulation that leads to things such as money laundry, the incapability of the malicious node to alter the belief by reversing a ledger, and the incapability of the node to eclipse a user.

$\forall d \in D, v \in V, \exists l \in L, authenticity(d) \wedge NonReversibility(l) \wedge Traceability(d,l) \wedge noEclipsing(l,v) \rightarrow validity(l)$

4. The usage of double key encryption within blockchain technology provides high authenticity over the data.

   $\forall d \in DdoubleKeySignature(d) \rightarrow authenticity(d)$

5. The increase of competing validators within a regional space is in the interest of users and validators. However, the registered business will add a high deterrence motive to each validator,

   $\forall v \in V, \exists l \in L, competence(v,l) \wedge deterrence(v) \rightarrow NoReversability(l)$

6. The traceability of the data is with high authenticity and validity due to the use of sequential hash generation over each linked block and high duplication among competitor validators.

   $\forall v \in V, \exists l \in L, d \in D, HighDuplication(l) \wedge hardToAlter(v,l) \rightarrow Traceability(d,l,v)$

7. Based on the formula that $Security(n_i) = \frac{Incentive \times ((N \times L)+(R \times (1-M) \times R)}{2}$ where m=[0,1], each member must ensure that its world is highly intersected within a networking level to ensure not being eclipsed. $\forall v \in V, u \in U, IntersectionLevel_1(v) \wedge Intersectionlevel_2(u) \rightarrow NoEclipsing()$

8. The formula that maintains security is based on increasing the number of validators. Thus, making the system open to any new participant, in addition to the financial gain provided that incentivises people to participate, will generate high competence over space of data and a high duplication().

   $\forall v \in V, \exists l \in L, d \in D, OpenSystem(l,v) \wedge HighNumberOfValidators(l) \wedge Gain(d) \rightarrow Competence(l) \wedge HighDuplication(l)$

9. A deterrence for validators within the blockchain technology is expected to ensure the trust of users within the system. The proposed model provides reputation destruction mechanisms to each mal behaved node beside the open of a node to be a registered business to ensure high deterrence and it is ensured by the assumption that increasing of number of node lower consensus. $N = 1 - C^{\frac{n(n-1)}{2}}$

$\forall v \in V, \exists l \in L, ThridParty(v) \wedge reputationDestruction(l) \rightarrow deterrence(v)$

10. To make the ledger hard to alter, the model uses a sequence of generated Hash()

    $\forall l \in L, SequenceOfGeneratedHash(l) \rightarrow hardToAlter(l)$

Derived from the model running the following proposed theory generalized in (1) leads eventually to the conclusion of validity.

6.3 Conclusion and future work

Blockchain is considered the greatest invention since the internet. The implementation of the bitcoin platform has led to huge hype surrounding the technology. However, the underlying incentive was to build a solution capable of preserving the confidentiality, transparency and autonomy of the system, but with increased security, reduced validation time and limited reliability. TheChain raised the fact that there is no need for consensus if everyone is able to make the right decision. It derives its philosophy from the idea that there is no need for global consistency in blockchain technology. TheCoin is the data structure that will be exchanged between the different validators in the system. It tried to give a formalism to the digitization of cash. Finally, TheTree was the algorithm to guide the different nodes to consistency.

6.3.1 Future Work

As a platform, the goal of the system is to provide a backdrop for a global machine that can coordinate between different nodes to ensure integrity. The solution is injected with a social inspired algorithm to bring the distributed system out of the rigid state that depends on coordination between servers for consistency so that users become part of the validity of the system. The authenticity and reputation of the network is subject to the construction of different versions of regional consistency which overlap and lead to the intersection of interests between the validators and, due to their independence, it will launch the competence on the service to the users. Another vision of the system is to look at the world as a set of users who initiate a tendency similar to human will, which makes them act like a human soul. Connecting these nodes to provide different worlds of existence injected with reputation destruction algorithm can be seen as the law of nature. Finally, the transition between a state and a belief combined with the distribution is the presumption. The claim is that intelligence is not a single component of computation but a part of a global network that forces us to act with a grand scheme of things.

This work needs to be deeper theoretically, and it can find a place to be implemented in many sectors. Here are business use case suggestions and explanations:

The bureaucratic institution in all its forms can adopt the developed system to lighten the load on the managements. Additionally, it can offer traceability on the generated information, which provides additional integrity and authenticity for their generated reward, which are coins or their generated information. Additionally, automation with system coordination between nodes to force oversight and transparency

For example, in the case of the financial sector, TheCoin will be the model above which the system will trade currencies. Currencies should be traced back to their origin to eliminate counterfeiting even in the impossible case of digital signature failure besides the fact that it is practically set up in the background for a search algorithm that harvests data to verify the connection between the spent coin and the information to eliminate manipulation, which is the basis of money laundering. Parallelism is another flavor of transaction processing at the distribution level, as the balance model suffers from a single entry point. TheChain as a system can be seen as different distributed affiliates which can be the basis of a vision of transparency for users due to its anonymity. Theft can be designed by the generating authority to freeze all coins by flagging it to force its subsequent exchange to be frozen as well. TheTree will work as a security base to force integrity into the system. The logic of the system can be inherited from many other institutions to be the basis for the exchange of information focusing on the transition rules.

The system can be seen as an information network, in which users submit their transactions in terms of values or information to the system to be explored further to build on it from a conceptual perspective in which the users' vision of truth will be reputable, authentic and part of the regional trend which can represent their logical and physical world of existence. For example, if we assume that a university wants to implement the system to be part of its student management. TheCoin will be responsible for managing the information. The users are teachers and students. However, few nodes can be maintainers. The type of users will submit university generated coins to each student. These coins, which here means information, are generated during the payment of tuition fees by the student. However, students will take advantage of TheTree in case of unfair submission of parts to inform other peers or to get involved in the management process. It can offer students and faculty double-blind management within the system to increase transparency and neutrality. This last discussion relates only to on-site examinations.

1. Exploring the system more theoretically must first address the notion of the prior within a distributed system because each member can be seen with relevance to its environment.
2. The study can be explored in terms of modeling the environment before drawing impacting variables which can be considered as reputation metrics to rank the nodes.
3. The idea of a world of worlds that generates overlapping regional consistency must be studied by the norm that forces these worlds to intersect.
4. This study must be done using optimization algorithms that support dynamics and are capable of interacting with an open context.

Bibliography

Abd-El-Malek, M., Ganger, G. R., Goodson, G. R., Reiter, M. K. and Wylie, J. J., 2005. Fault-scalable byzantine fault-tolerant services. *ACM SIGOPS Operating Systems Review*, 39 (5), 59–74.

Abdullah, N., Hakansson, A. and Moradian, E., 2017. Blockchain based approach to enhance big data authentication in distributed environment. *2017 Ninth international conference on ubiquitous and future networks (ICUFN)*, IEEE, 887–892.

Abraham, I., Malkhi, D., Nayak, K. and Ren, L., 2021. Flexible byzantine fault tolerance. US Patent App. 17/107,630.

Al-Mashhadi, S. and Manickam, S., 2020. A brief review of blockchain-based dns systems. *International Journal of Internet Technology and Secured Transactions*, 10 (4), 420–432.

Ali, A., Latif, S., Qadir, J., Kanhere, S., Singh, J., Crowcroft, J. et al., 2019. Blockchain and the future of the internet: A comprehensive review. *arXiv preprint arXiv:1904.00733*.

AlMallohi, I. A. I., Alotaibi, A. S. M., Alghafees, R., Azam, F. and Khan, Z. S., 2019. Multivariable based checkpoints to mitigate the long range attack in proof-of-stake based blockchains. *Proceedings of the 3rd International Conference on High Performance Compilation, Computing and Communications*, 118–122.

Amir, Y., Coan, B., Kirsch, J. and Lane, J., 2010. Prime: Byzantine replication under attack. *IEEE Transactions on Dependable and Secure Computing*, 8 (4), 564–577.

Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., Enyeart, D., Ferris, C., Laventman, G., Manevich, Y. et al., 2018. Hyperledger fabric: a distributed operating system for permissioned blockchains. *Proceedings of the thirteenth EuroSys conference*, 1–15.

Awe, K. F., Malik, Y., Zavarsky, P. and Jaafar, F., 2020. Validating bgp update using blockchain-based infrastructure. *Decentralised Internet of Things*, Springer, 151–165.

Back, A. et al., 2002. Hashcash-a denial of service counter-measure.

Bag, S., Ruj, S. and Sakurai, K., 2016. Bitcoin block withholding attack: Analysis and mitigation. *IEEE Transactions on Information Forensics and Security*, 12 (8), 1967–1978.

Balaji, P. and Srinivasan, D., 2010. An introduction to multi-agent systems. *Innovations in multi-agent systems and applications-1*, Springer, 1–27.

Baldominos, A. and Saez, Y., 2019. Coin. ai: A proof-of-useful-work scheme for blockchain-based distributed deep learning. *Entropy*, 21 (8), 723.

Banakar, R., Steinke, S., Lee, B.-S., Balakrishnan, M. and Marwedel, P., 2002. Scratchpad memory: A design alternative for cache on-chip memory in embedded systems. *Proceedings of the Tenth International Symposium on Hardware/Software Codesign. CODES 2002 (IEEE Cat. No. 02TH8627)*, IEEE, 73–78.

Bank, W., 2022. https://github.com/ikbalnacer/data.

Bayer, D., Haber, S. and Stornetta, W. S., 1993. Improving the efficiency and reliability of digital time-stamping. *Sequences Ii*, Springer, 329–334.

Belotti, M., Kirati, S. and Secci, S., 2018. Bitcoin pool-hopping detection. *2018 IEEE 4th International Forum on Research and Technology for Society and Industry (RTSI)*, IEEE, 1–6.

Ben Mariem, S., Casas, P. and Donnet, B., 2018. Vivisecting blockchain p2p networks: Unveiling the bitcoin ip network. *ACM CoNEXT student workshop*.

Bentov, I., Lee, C., Mizrahi, A. and Rosenfeld, M., 2014. Proof of activity: Extending bitcoin's proof of work via proof of stake [extended abstract] y. *ACM SIGMETRICS Performance Evaluation Review*, 42 (3), 34–37.

Bentov, I., Pass, R. and Shi, E., 2016. Snow white: Provably secure proofs of stake. *IACR Cryptol. ePrint Arch.*, 2016 (919).

Bertoni, G., Daemen, J., Peeters, M. and Assche, G. V., 2013. Keccak. *Annual international conference on the theory and applications of cryptographic techniques*, Springer, 313–314.

Boden, N. J., Cohen, D., Felderman, R. E., Kulawik, A. E., Seitz, C. L., Seizovic, J. N. and Su, W.-K., 1995. Myrinet: A gigabit-per-second local area network. *IEEE micro*, 15 (1), 29–36.

Bonneau, J., Miller, A., Clark, J., Narayanan, A., Kroll, J. A. and Felten, E. W., 2015. Sok: Research perspectives and challenges for bitcoin and cryptocurrencies. *2015 IEEE Symposium on Security and Privacy*, IEEE, 104–121.

Bonniot, L., Neumann, C. and Taïani, F., 2020. Pnyxdb: a lightweight leaderless democratic byzantine fault tolerant replicated datastore. *2020 International Symposium on Reliable Distributed Systems (SRDS)*, IEEE, 155–164.

Bosamia, M. and Patel, D., 2018. Current trends and future implementation possibilities of the merkel tree. *International Journal of Computer Sciences and Engineering*, 6 (8), 294–301.

Brunnermeier, M. K., James, H. and Landau, J.-P., 2019. The digitalization of money. Technical report, National Bureau of Economic Research.

Bu, G., Gürcan, Ö. and Potop-Butucaru, M., 2019. G-iota: Fair and confidence aware tangle. *IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, IEEE, 644–649.

Bu, G., Hana, W. and Potop-Butucaru, M., 2020. E-iota: an efficient and fast metamorphism for iota. *2020 2nd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS)*, IEEE, 9–16.

Buccafurri, F., Lax, G., Nicolazzo, S. and Nocera, A., 2017. Overcoming limits of blockchain for iot applications. *Proceedings of the 12th International Conference on Availability, Reliability and Security*, 1–6.

Buchman, E., 2016. *Tendermint: Byzantine fault tolerance in the age of blockchains*. Ph.D. thesis, University of Guelph.

Buhr, P. A., Dice, D. and Hesselink, W. H., 2015. High-performance n-thread software solutions for mutual exclusion. *Concurrency and Computation: Practice and Experience*, 27 (3), 651–701.

Buterin, V. and Griffith, V., 2017. Casper the friendly finality gadget. *arXiv preprint arXiv:1710.09437*.

Castro, M. and Liskov, B., 2002. Practical byzantine fault tolerance and proactive recovery. *ACM Transactions on Computer Systems (TOCS)*, 20 (4), 398–461.

Chan, B. Y. and Shi, E., 2020. Streamlet: Textbook streamlined blockchains. *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies*, 1–11.

Chari, K., 2003. Model composition in a distributed environment. *Decision Support Systems*, 35 (3), 399–413.

Chaudhry, N. and Yousaf, M. M., 2018. Consensus algorithms in blockchain: comparative analysis, challenges and opportunities. *2018 12th International Conference on Open Source Systems and Technologies (ICOSST)*, IEEE, 54–63.

Chawla, N., Behrens, H. W., Tapp, D., Boscovic, D. and Candan, K. S., 2019. Velocity: Scalability improvements in block propagation through rateless erasure coding. *2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, IEEE, 447–454.

Chen, J., Duan, K., Zhang, R., Zeng, L. and Wang, W., 2018. An ai based super nodes selection algorithm in blockchain networks. *arXiv preprint arXiv:1808.00216*.

Chen, K. and Shen, H., 2013. Smart: Utilizing distributed social map for lightweight routing in delay-tolerant networks. *IEEE/ACM Transactions on networking*, 22 (5), 1545–1558.

Chen, L., Xu, L., Shah, N., Gao, Z., Lu, Y. and Shi, W., 2017. On security analysis of proof-of-elapsed-time (poet). *International Symposium on Stabilization, Safety, and Security of Distributed Systems*, Springer, 282–297.

Chourasia, S., 2013. Survey paper on improved methods of id3 decision tree classification. *International Journal of Scientific and Research Publications*, 3 (12), 1–2.

Clavería, A., Delgado-Martín, M. V., Goicoechea-Castaño, A., Iglesias-Moreno, J. M., García-Cendón, C., Martín-Miguel, M. V., Villarino-Moure, R., Barreiro-Arceiz, C., Rey-Gómez-Serranillos, I. and Roca, J., 2022. Interrupted time series analysis of pediatric infectious diseases and the consumption of antibiotics in an atlantic european region during the sars-cov-2 pandemic. *Antibiotics*, 11 (2), 264.

Clow, J. and Jiang, Z., 2017. A byzantine fault tolerant raft.

Dai, Q., Zhang, B. and Dong, S., 2022. Eclipse attack detection for blockchain network layer based on deep feature extraction. *Wireless Communications and Mobile Computing*, 2022.

Damba, A. and Watanabe, S., 2007. Hierarchical control in a multiagent system. *Second International Conference on Innovative Computing, Informatio and Control (ICICIC 2007)*, IEEE, 111–111.

Dannen, C., 2017. *Introducing Ethereum and solidity*, volume 1. Springer.

Das, D., 2021. Toward next generation of blockchain using improvized bitcoin-ng. *IEEE Transactions on Computational Social Systems*, 8 (2), 512–521.

De Angelis, S., Aniello, L., Baldoni, R., Lombardi, F., Margheri, A. and Sassone, V., 2018. Pbft vs proof-of-authority: Applying the cap theorem to permissioned blockchain.

De Roode, G., Ullah, I. and Havinga, P. J., 2018. How to break iota heart by replaying? *2018 IEEE Globecom Workshops (GC Wkshps)*, IEEE, 1–7.

Deirmentzoglou, E., Papakyriakopoulos, G. and Patsakis, C., 2019. A survey on long-range attacks for proof of stake protocols. *IEEE Access*, 7, 28712–28725.

Delgado-Segura, S., Pérez-Sola, C., Navarro-Arribas, G. and Herrera-Joancomartí, J., 2018. Analysis of the bitcoin utxo set. *International Conference on Financial Cryptography and Data Security*, Springer, 78–91.

Delgrande, J. P., Peppas, P. and Woltran, S., 2018. General belief revision. *Journal of the ACM (JACM)*, 65 (5), 1–34.

Devi, A., Rathee, G. and Saini, H., 2021. Security concerns at various network phases through blockchain technology. *Applications of Artificial Intelligence and Machine Learning*, Springer, 605–616.

Dey, S., 2018. Securing majority-attack in blockchain using machine learning and algorithmic game theory: A proof of work. *2018 10th computer science and electronic engineering (CEEC)*, IEEE, 7–10.

Dillenberger, D., Novotny, P., Zhang, Q., Jayachandran, P., Gupta, H., Hans, S., Verma, D., Chakraborty, S., Thomas, J., Walli, M. et al., 2019. Blockchain analytics and artificial intelligence. *IBM Journal of Research and Development*, 63 (2/3), 5–1.

Dodis, Y. and Yampolskiy, A., 2005. A verifiable random function with short proofs and keys. *International Workshop on Public Key Cryptography*, Springer, 416–431.

Dotan, M., Pignolet, Y.-A., Schmid, S., Tochner, S. and Zohar, A., 2020. Sok: cryptocurrency networking context, state-of-the-art, challenges. *Proceedings of the 15th International Conference on Availability, Reliability and Security*, 1–13.

Dotan, M., Pignolet, Y.-A., Schmid, S., Tochner, S. and Zohar, A., 2021. Survey on blockchain networking: Context, state-of-the-art, challenges. *ACM Computing Surveys (CSUR)*, 54 (5), 1–34.

El Ioini, N. and Pahl, C., 2018. A review of distributed ledger technologies. *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, Springer, 277–288.

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

Fakhri, D. and Mutijarsa, K., 2018. Secure iot communication using blockchain technology. *2018 International Symposium on Electronics and Smart Devices (ISESD)*, IEEE, 1–6.

Fan, X. and Chai, Q., 2018. Roll-dpos: a randomized delegated proof of stake scheme for scalable blockchain-based internet of things systems. *Proceedings of the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, 482–484.

Fauzi, P., Lipmaa, H., Siim, J., Zając, M. and Ødegaard, A. T., 2021. Verifiably-extractable owfs and their applications to subversion zero-knowledge. *International Conference on the Theory and Application of Cryptology and Information Security*, Springer, 618–649.

Fischer, M. J., Lynch, N. A. and Paterson, M. S., 1985. Impossibility of distributed consensus with one faulty process. *Journal of the ACM (JACM)*, 32 (2), 374–382.

Ganguly, R., Momtaz, A. and Bonakdarpour, B., 2021. Distributed runtime verification under partial synchrony. *24th International Conference on Principles of Distributed Systems (OPODIS 2020)*, Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

Gaži, P., Kiayias, A. and Russell, A., 2018. Stake-bleeding attacks on proof-of-stake blockchains. *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*, IEEE, 85–92.

Gaži, P., Kiayias, A. and Zindros, D., 2019. Proof-of-stake sidechains. *2019 IEEE Symposium on Security and Privacy (SP)*, IEEE, 139–156.

Gilad, Y., Hemo, R., Micali, S., Vlachos, G. and Zeldovich, N., 2017. Algorand: Scaling byzantine agreements for cryptocurrencies. *Proceedings of the 26th symposium on operating systems principles*, 51–68.

Göbel, J., Keeler, H. P., Krzesinski, A. E. and Taylor, P. G., 2016. Bitcoin blockchain dynamics: The selfish-mine strategy in the presence of propagation delay. *Performance Evaluation*, 104, 23–41.

Government, U., 2019. Standards for the fourth industrial revolution.

Gramoli, V., 2020. From blockchain consensus back to byzantine consensus. *Future Generation Computer Systems*, 107, 760–769.

Haber, S. and Stornetta, W., 1991. How to time-stamp a digital document, crypto'90, lncs 537.

Haber, S. and Stornetta, W. S., 1990. How to time-stamp a digital document. *Conference on the Theory and Application of Cryptography*, Springer, 437–455.

Haeberlen, A., Kouznetsov, P. and Druschel, P., 2007. Peerreview: Practical accountability for distributed systems. *ACM SIGOPS operating systems review*, 41 (6), 175–188.

Haghighat, A. T. and Shajari, M., 2019. Block withholding game among bitcoin mining pools. *Future Generation Computer Systems*, 97, 482–491.

Haldimann, J., Sauerwald, K., von Berg, M., Kern-Isberner, G. and Beierle, C., 2021. Towards a framework of hansson's descriptor revision for conditionals. *Proceedings of the 36th Annual ACM Symposium on Applied Computing*, 889–891.

Halford, R., 2014. Gridcoin: Crypto-currency using berkeley open infrastructure network computing grid as a proof of work.

Haouari, M., Mhiri, M., El-Masri, M. and Al-Yafi, K., 2022. A novel proof of useful work for a blockchain storing transportation transactions. *Information Processing & Management*, 59 (1), 102749.

Hartmann, F., Grottolo, G., Wang, X. and Lunesu, M. I., 2019. Alternative fundraising: success factors for blockchain-based vs. conventional crowdfunding. *2019 IEEE international workshop on blockchain oriented software engineering (IWBOSE)*, IEEE, 38–43.

Hayes, A. S., 2017. Cryptocurrency value formation: An empirical study leading to a cost of production model for valuing bitcoin. *Telematics and Informatics*, 34 (7), 1308–1321.

He, Q., Xu, Y., Yan, Y., Wang, J., Han, Q. and Li, L., 2018. A consensus and incentive program for charging piles based on consortium blockchain. *CSEE journal of power and energy systems*, 4 (4), 452–458.

Heilman, E., Narula, N., Tanzer, G., Lovejoy, J., Colavita, M., Virza, M. and Dryja, T., 2019. Cryptanalysis of curl-p and other attacks on the iota cryptocurrency. *IACR Cryptology ePrint Archive*, 2019, 344.

Hendricks, J., 2009. *Efficient Byzantine fault tolerance for scalable storage and services*. Ph.D. thesis, Carnegie Mellon University.

Hendricks, J., Ganger, G. R. and Reiter, M. K., 2007. Verifying distributed erasure-coded data. *Proceedings of the twenty-sixth annual ACM symposium on Principles of distributed computing*, 139–146.

Hoffmann, M., Klooß, M. and Rupp, A., 2019. Efficient zero-knowledge arguments in the discrete log setting, revisited. *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2093–2110.

Horikawa, T., 2011. An unexpected scalability bottleneck in a dbms: a hidden pitfall in implementing mutual exclusion. *Parallel and Distributed Computing and Systems*.

Hou, H., 2017. The application of blockchain technology in e-government in china. *2017 26th International Conference on Computer Communication and Networks (ICCCN)*, IEEE, 1–4.

Hunt, P., Konar, M., Junqueira, F. P. and Reed, B., 2010. {ZooKeeper}: Wait-free coordination for internet-scale systems. *2010 USENIX Annual Technical Conference (USENIX ATC 10)*.

Iacona, A., 2021. *LOGIC: Lecture Notes for Philosophy, Mathematics, and Computer Science*. Springer.

Ikbal Nacer, M., Prakoonwit, S. and Prakash, E., 2021. Thecoin: Privacy and security considerations within blockchain transactions. *2021 2nd Asia Service Sciences and Software Engineering Conference*, 10–17.

Ismail, L. and Materwala, H., 2019. A review of blockchain architecture and consensus protocols: Use cases, challenges, and solutions. *Symmetry*, 11 (10), 1198.

Jameel, F., Nabeel, M., Jamshed, M. A. and Jäntti, R., 2020. Minimizing forking in blockchain-based iot networks. *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, IEEE, 1–6.

Jang, J. and Lee, H.-N., 2020. Profitable double-spending attacks. *Applied Sciences*, 10 (23), 8477.

Jha, S., Behrens, J., Gkountouvas, T., Milano, M., Song, W., Tremel, E., Renesse, R. V., Zink, S. and Birman, K. P., 2019. Derecho: Fast state machine replication for cloud services. *ACM Transactions on Computer Systems (TOCS)*, 36 (2), 1–49.

Johnson, B., Laszka, A., Grossklags, J., Vasek, M. and Moore, T., 2014. Game-theoretic analysis of ddos attacks against bitcoin mining pools. *International Conference on Financial Cryptography and Data Security*, Springer, 72–86.

Karame, G. O., Androulaki, E., Roeschlin, M., Gervais, A. and Čapkun, S., 2015. Misbehavior in bitcoin: A study of double-spending and accountability. *ACM Transactions on Information and System Security (TISSEC)*, 18 (1), 1–32.

Kern-Isberner, G., Bock, T., Sauerwald, K. and Beierle, C., 2019. Belief change properties of forgetting operations over ranking functions. *Pacific Rim International Conference on Artificial Intelligence*, Springer, 459–472.

Khaldun, I., 2015. *The muqaddimah: an introduction to history-abridged Edition*. Princeton University Press.

Kiayias, A. and Russell, A., 2018. Ouroboros-bft: A simple byzantine fault tolerant consensus protocol. *Cryptology ePrint Archive*.

Kiayias, A., Russell, A., David, B. and Oliynykov, R., 2017. Ouroboros: A provably secure proof-of-stake blockchain protocol. *Annual international cryptology conference*, Springer, 357–388.

Kim, S. K., Ma, Z., Murali, S., Mason, J., Miller, A. and Bailey, M., 2018. Measuring ethereum network peers. *Proceedings of the Internet Measurement Conference 2018*, 91–104.

Koblitz, N., Menezes, A. and Vanstone, S., 2000. The state of elliptic curve cryptography. *Designs, codes and cryptography*, 19 (2), 173–193.

Kondo, D., Javadi, B., Iosup, A. and Epema, D., 2010. The failure trace archive: Enabling comparative analysis of failures in diverse distributed systems. *2010 10th IEEE/ACM international conference on cluster, cloud and grid computing*, IEEE, 398–407.

Kotla, R., Alvisi, L., Dahlin, M., Clement, A. and Wong, E., 2007. Zyzzyva: speculative byzantine fault tolerance. *ACM SIGOPS Operating Systems Review*, 41 (6), 45–58.

Kroll, J. A., Davey, I. C. and Felten, E. W., 2013. The economics of bitcoin mining, or bitcoin in the presence of adversaries. *Proceedings of WEIS*, Washington, DC, volume 2013.

Kshetri, N. and Voas, J., 2018. Blockchain-enabled e-voting. *Ieee Software*, 35 (4), 95–99.

Kulkarni, S. S., Appleton, G. and Nguyen, D., 2022. Achieving causality with physical clocks. *23rd International Conference on Distributed Computing and Networking*, 97–106.

Kumar, M. A., Radhesyam, V. and Srinivasarao, B., 2019. Front-end iot application for the bitcoin based on proof of elapsed time (poet). *2019 Third International Conference on Inventive Systems and Control (ICISC)*, IEEE, 646–649.

Kumar, R. and Tripathi, R., 2019. Traceability of counterfeit medicine supply chain through blockchain. *2019 11th International Conference on Communication Systems & Networks (COMSNETS)*, IEEE, 568–570.

Lai, H. H., 2019. How plausible is the relative plausibility theory of proof? *The International Journal of Evidence & Proof*, 23 (1-2), 191–197.

Lamport, L., 1984. Using time instead of timeout for fault-tolerant distributed systems. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 6 (2), 254–280.

Lamport, L., 2001. Paxos made simple. *ACM SIGACT News (Distributed Computing Column) 32, 4 (Whole Number 121, December 2001)*, 51–58.

Lamport, L., 2019. Time, clocks, and the ordering of events in a distributed system. *Concurrency: the Works of Leslie Lamport*, 179–196.

Lamport, L. and Merz, S., 2022. Prophecy made simple. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 44 (2), 1–27.

Lamport, L., Shostak, R. and Pease, M., 2019. The byzantine generals problem. *Concurrency: the Works of Leslie Lamport*, 203–226.

Lao, L., Dai, X., Xiao, B. and Guo, S., 2020. G-pbft: a location-based and scalable consensus protocol for iot-blockchain applications. *2020 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, IEEE, 664–673.

Lathif, M. R. A., Nasirifard, P. and Jacobsen, H.-A., 2018. Cidds: A configurable and distributed dag-based distributed ledger simulation framework. *Proceedings of the 19th International Middleware Conference (Posters)*, 7–8.

Lee, S. and Kim, S., 2020. Short selling attack: A self-destructive but profitable 51% attack on pos blockchains. *Cryptology ePrint Archive*.

Leiding, B., Memarmoshrefi, P. and Hogrefe, D., 2016. Self-managed and blockchain-based vehicular ad-hoc networks. *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, 137–140.

Lerner, S. D., 2014. Strict memory hard hashing functions (preliminary v0. 3, 01-19-14).

Lewenberg, Y., Bachrach, Y., Sompolinsky, Y., Zohar, A. and Rosenschein, J. S., 2015. Bitcoin mining pools: A cooperative game theoretic analysis. *Proceedings of the 2015 international conference on autonomous agents and multiagent systems*, Citeseer, 919–927.

Li, C., Hurfin, M., Wang, Y. and Yu, L., 2016. Towards a restrained use of non-equivocation for achieving iterative approximate byzantine consensus. *2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, IEEE, 710–719.

Liao, K. and Katz, J., 2017. Incentivizing blockchain forks via whale transactions. *International conference on financial cryptography and data security*, Springer, 264–279.

Lipmaa, H., Siim, J. and Zajac, M., 2022. Counting vampires: From univariate sumcheck to updatable zk-snark. *Cryptology ePrint Archive*.

Liu, Z., Luong, N. C., Wang, W., Niyato, D., Wang, P., Liang, Y.-C. and Kim, D. I., 2019. A survey on applications of game theory in blockchain. *arXiv preprint arXiv:1902.10865*.

Loe, A. F. and Quaglia, E. A., 2018. Conquering generals: an np-hard proof of useful work. *Proceedings of the 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems*, 54–59.

Luz, M. A. d. and Farias, K., 2020. The use of blockchain in financial area: A systematic mapping study. *XVI Brazilian Symposium on Information Systems*, 1–8.

MacIntosh, J., 2017. Causality. *The Arguments of Aquinas*, Routledge, 31–44.

Mahmood, K., Chaudhry, S. A., Naqvi, H., Kumari, S., Li, X. and Sangaiah, A. K., 2018. An elliptic curve cryptography based lightweight authentication scheme for smart grid communication. *Future Generation Computer Systems*, 81, 557–565.

Malkhi, D., Nayak, K. and Ren, L., 2019. Flexible byzantine fault tolerance. *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, 1041–1053.

Manfren, M., Caputo, P. and Costa, G., 2011. Paradigm shift in urban energy systems through distributed generation: Methods and models. *Applied energy*, 88 (4), 1032–1048.

Martin, A. J., 1985. A new generalization of dekker's algorithm for mutual exclusion.

Marvic, R., Merle, P. and Geib, J.-M., 2000. Towards a dynamic corba component platform. *Proceedings DOA'00. International Symposium on Distributed Objects and Applications*, IEEE, 305–314.

Mei, W., 2019. Formalization of fuzzy control in possibility theory via rule extraction. *IEEE Access*, 7, 90115–90124.

Meolic, R., Kapus, T., Gungl, E. and Brezocnik, Z., 2001. Verification of mutual exclusion algorithms with est. *Proceedings of the Tenth Electrotechnical and Computer Science Conference ERK'2001 Portoroz, Slovenia*, Citeseer, 15–18.

Merkle, R. C., 1980. Protocols for public key cryptosystems. *1980 IEEE Symposium on Security and Privacy*, IEEE, 122–122.

Miers, I., Garman, C., Green, M. and Rubin, A. D., 2013. Zerocoin: Anonymous distributed e-cash from bitcoin. *2013 IEEE Symposium on Security and Privacy*, IEEE, 397–411.

Miller, A., Xia, Y., Croman, K., Shi, E. and Song, D., 2016. The honey badger of bft protocols. *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 31–42.

Milutinović, D. and Lučanin, V., 2005. Relation between reliability and availability of railway vehicles. *FME Transactions*, 33 (3), 135–139.

Mingxiao, D., Xiaofeng, M., Zhe, Z., Xiangwei, W. and Qijun, C., 2017. A review on consensus algorithm of blockchain. *2017 IEEE international conference on systems, man, and cybernetics (SMC)*, IEEE, 2567–2572.

Moindrot, O. and Bournhonesque, C., 2017. Proof of stake made simple with casper. *ICME, Stanford University*.

Nabilou, H., 2022. Probabilistic settlement finality in proof-of-work blockchains: Legal considerations. *Amsterdam Law School Research Paper*, (2022-04).

Nacer, M. I., Prakoonwit, S. and Alarab, I., 2020. Thechain: A fast, secure and parallel treatment of transactions. *Proceedings of the 2020 2nd International Electronics Communication Conference*, 81–89.

Nacer, M. I., Prakoonwit, S. and Alarab, I., 2021a. Blockchain as a complementary technology for the internet of things: A survey. *Internet of Things*, Springer, 1–24.

Nacer, M. I., Prakoonwit, S. and Alarab, I., 2021b. The combination of ai, blockchain, and the internet of things for patient relationship management. *Internet of Things*, Springer, 49–65.

Nakamoto, S., 2008. Bitcoin: A peer-to-peer electronic cash system. *Decentralized Business Review*, 21260.

Nakamoto, S. et al., 2008. A peer-to-peer electronic cash system. *Bitcoin.–URL: https://bitcoin. org/bitcoin. pdf*.

Naor, O., Baudet, M., Malkhi, D. and Spiegelman, A., 2019. Cogsworth: Byzantine view synchronization. *arXiv preprint arXiv:1909.05204*.

Nencha, C., 2021. Necessitism, contingentism, and lewisian modal realism. *Acta Analytica*, 1–21.

Nerurkar, P., Patel, D., Busnel, Y., Ludinard, R., Kumari, S. and Khan, M. K., 2021. Dissecting bitcoin blockchain: Empirical analysis of bitcoin network (2009–2020). *Journal of Network and Computer Applications*, 177, 102940.

Neudecker, T., 2019. *Characterization of the bitcoin peer-to-peer network (2015-2018)*. KIT Karlsruher Institut für Technologie, Fakultät für Informatik.

Nguyen, C. T., Hoang, D. T., Nguyen, D. N., Niyato, D., Nguyen, H. T. and Dutkiewicz, E., 2019. Proof-of-stake consensus mechanisms for future blockchain networks: fundamentals, applications and opportunities. *IEEE Access*, 7, 85727–85745.

Onireti, O., Zhang, L. and Imran, M. A., 2019. On the viable area of wireless practical byzantine fault tolerance (pbft) blockchain networks. *2019 IEEE Global Communications Conference (GLOBECOM)*, IEEE, 1–6.

Ozisik, A. P., Andresen, G., Levine, B. N., Tapp, D., Bissias, G. and Katkuri, S., 2019. Graphene: efficient interactive set reconciliation applied to blockchain propagation. *Proceedings of the ACM Special Interest Group on Data Communication*, 303–317.

Pantano, P., Iannetti, G. D., Caramia, F., Mainero, C., Di Legge, S., Bozzao, L., Pozzilli, C. and Lenzi, G. L., 2002. Cortical motor reorganization after a single clinical attack of multiple sclerosis. *Brain*, 125 (7), 1607–1615.

Perez, J. A., Deligianni, F., Ravi, D. and Yang, G.-Z., 2018. Artificial intelligence and robotics. *arXiv preprint arXiv:1803.10813*, 147.

Picco, G. P., Fuggetta, A. and Vigna, G., 2000. Understanding code mobility. *INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING*, volume 22, 834–834.

Pomelnikov, A. G., 2021. The impact of blockchain on emerging economies. *The Journal of Applied Business and Economics*, 23 (1), 277–284.

Popov, S., 2016. The tangle. *cit. on*, 131.

Popper, N., 2015. Decoding the enigma of satoshi nakamoto and the birth of bitcoin. *New York Times*, 15.

Prabhu, Y. and Varma, M., 2014. Fastxml: A fast, accurate and stable tree-classifier for extreme multi-label learning. *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 263–272.

Pries, R., Yu, W., Fu, X. and Zhao, W., 2008. A new replay attack against anonymous communication networks. *2008 IEEE International Conference on Communications*, IEEE, 1578–1582.

Raikwar, M., Gligoroski, D. and Kralevska, K., 2019. Sok of used cryptography in blockchain. *IEEE Access*, 7, 148550–148575.

Ramchandani, C., 1973. *Analysis of asynchronous concurrent systems by timed Petri nets.*. Ph.D. thesis, Massachusetts Institute of Technology.

Rana, R., Zaeem, R. N. and Barber, K. S., 2019. An assessment of blockchain identity solutions: Minimizing risk and liability of authentication. *2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, IEEE, 26–33.

Rathore, H., Samant, A. and Jadliwala, M., 2020. Tanglecv: a distributed ledger technique for secure message sharing in connected vehicles. *ACM Transactions on Cyber-Physical Systems*, 5 (1), 1–25.

Recabarren, R. and Carbunar, B., 2017. Hardening stratum, the bitcoin pool mining protocol. *arXiv preprint arXiv:1703.06545*.

Reyna, A., Martín, C., Chen, J., Soler, E. and Díaz, M., 2018. On blockchain and its integration with iot. challenges and opportunities. *Future generation computer systems*, 88, 173–190.

Richards, M., 2015. *Software architecture patterns*, volume 4. O'Reilly Media, Incorporated 1005 Gravenstein Highway North, Sebastopol, CA . . . .

Rizun, P. R., 2016. Subchains: A technique to scale bitcoin and improve the user experience. *Ledger*, 1, 38–52.

Rodrigues, B., Bocek, T., Lareida, A., Hausheer, D., Rafati, S. and Stiller, B., 2017. A blockchain-based architecture for collaborative ddos mitigation with smart contracts.

IFIP International Conference on Autonomous Infrastructure, Management and Security, Springer, Cham, 16–29.

Roy, N., Shen, S., Hassanieh, H. and Choudhury, R. R., 2018. Inaudible voice commands: The {Long-Range} attack and defense. *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*, 547–560.

Saad, M., Spaulding, J., Njilla, L., Kamhoua, C., Shetty, S., Nyang, D. and Mohaisen, A., 2019. Exploring the attack surface of blockchain: A systematic overview. *arXiv preprint arXiv:1904.03487*.

Saad, M., Spaulding, J., Njilla, L., Kamhoua, C., Shetty, S., Nyang, D. and Mohaisen, D., 2020. Exploring the attack surface of blockchain: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 22 (3), 1977–2008.

Salimitari, M., Chatterjee, M., Yuksel, M. and Pasiliao, E., 2017. Profit maximization for bitcoin pool mining: A prospect theoretic approach. *2017 IEEE 3rd international conference on collaboration and internet computing (CIC)*, IEEE, 267–274.

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

Sayeed, S. and Marco-Gisbert, H., 2019. Assessing blockchain consensus and security mechanisms against the 51% attack. *Applied Sciences*, 9 (9), 1788.

Segendorf, B., 2014. What is bitcoin. *Sveri gesRiksbankEconomicReview*, 2014, 2–71.

Shalini, S. and Santhi, H., 2019. A survey on various attacks in bitcoin and cryptocurrency. *2019 International Conference on Communication and Signal Processing (ICCSP)*, IEEE, 0220–0224.

Sharma, M. G. and Kumar, S., 2020. The implication of blockchain as a disruptive technology for construction industry. *IIM Kozhikode Society & Management Review*, 9 (2), 177–188.

Shi, H. and Wang, X., 2018. Research on the development path of blockchain in shipping industry. *Proceedings of the Asia-Pacific Conference on Intelligent Medical 2018 & International Conference on Transportation and Traffic Engineering 2018*, 243–247.

Silvano, W. F. and Marcelino, R., 2020. Iota tangle: A cryptocurrency to communicate internet-of-things data. *Future generation computer systems*, 112, 307–319.

Singh, A., Fonseca, P., Kuznetsov, P., Rodrigues, R., Maniatis, P. et al., 2009. Zeno: Eventually consistent byzantine-fault tolerance. *NSDI*, volume 9, 169–184.

Singh, D. and Garg, R., 2021. Ni-louvain: A novel algorithm to detect overlapping communities with influence analysis. *Journal of King Saud University-Computer and Information Sciences*.

Smith, J., 2016. An analysis of bitcoin exchange rates. *Available at SSRN 2493797*.

Sohrabi, N. and Tari, Z., 2020. Zyconchain: A scalable blockchain for general applications. *IEEE Access*, 8, 158893–158910.

Spasovski, J. and Eklund, P., 2017. Proof of stake blockchain: performance and scalability for groupware communications. *Proceedings of the 9th International Conference on Management of Digital EcoSystems*, 251–258.

Spohn, W., 2012. *The laws of belief: Ranking theory and its philosophical applications*. Oxford University Press.

Stathakopoulou, C., David, T., Pavlovic, M. and Vukolić, M., 2019. Mir-bft: High-throughput robust bft for decentralized networks. *arXiv preprint arXiv:1906.05552*.

Stroud, R., 1992. Transparency and reflection in distributed systems. *Proceedings of the 5th workshop on ACM SIGOPS European workshop: Models and paradigms for distributed systems structuring*, 1–5.

Sun, J., Yao, X., Wang, S. and Wu, Y., 2020. Non-repudiation storage and access control scheme of insurance data based on blockchain in ipfs. *IEEE Access*, 8, 155145–155155.

Swan, M., 2018. Blockchain enlightenment and smart city cryptopolis. *Proceedings of the 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems*, 48–53.

Szabo, N., 1997. Formalizing and securing relationships on public networks. *First Monday*, 2 (9).

Tang, S., Zheng, J., Deng, Y. and Cao, Q., 2021. Resisting newborn attacks via shared proof-of-space. *Journal of Parallel and Distributed Computing*, 150, 85–95.

Tang, W. and Bennett, D. A., 2010. Agent-based modeling of animal movement: a review. *Geography Compass*, 4 (7), 682–700.

Tuzi, D., 2018. *Cryptonight Gpu Mining Efficiency*. Master's thesis.

Vasin, P., 2014. Blackcoin's proof-of-stake protocol v2. *URL: https://blackcoin. co/blackcoin-pos-protocol-v2-whitepaper. pdf*, 71.

Veronese, G. S., Correia, M., Bessani, A. N. and Lung, L. C., 2009. Spin one's wheels? byzantine fault tolerance with a spinning primary. *2009 28th IEEE International Symposium on Reliable Distributed Systems*, IEEE, 135–144.

Vinodhini, R. and Gomathy, C., 2020. Momhr: a dynamic multi-hop routing protocol for wsn using heuristic based multi-objective function. *Wireless Personal Communications*, 111 (2), 883–907.

Vo, H. T., Mehedy, L., Mohania, M. and Abebe, E., 2017. Blockchain-based data management and analytics for micro-insurance applications. *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2539–2542.

Voulodimos, A., Doulamis, N., Doulamis, A. and Protopapadakis, E., 2018. Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018.

Vukolić, M., 2015. The quest for scalable blockchain fabric: Proof-of-work vs. bft replication. *International workshop on open problems in network security*, Springer, 112–125.

Vyas, C. A. and Lunagaria, M., 2014. Security concerns and issues for bitcoin. *International Journal of Computer Applications*, 10–12.

Wang, D., Jin, C., Li, H. and Perkowski, M., 2020. Proof of activity consensus algorithm based on credit reward mechanism. *International Conference on Web Information Systems and Applications*, Springer, 618–628.

Wang, H., Yan, Q. and Leung, V., 2021. The impact of propagation delay to different selfish miners in proof-of-work blockchains. *Peer-to-Peer Networking and Applications*, 14 (5), 2735–2742.

Wang, Q., Wang, T., Shen, Z., Jia, Z., Zhao, M. and Shao, Z., 2019a. Re-tangle: A reram-based processing-in-memory architecture for transaction-based blockchain. *2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, IEEE, 1–8.

Wang, X., WeiLi, J. and Chai, J., 2018. The research on the incentive method of consortium blockchain based on practical byzantine fault tolerant. *2018 11th international symposium on computational intelligence and design (ISCID)*, IEEE, volume 2, 154–156.

Wang, Z., Liu, J., Zhang, Z., Zhang, Y., Yin, J., Yu, H. and Liu, W., 2019b. A combined micro-block chain truncation attack on bitcoin-ng. *Australasian Conference on Information Security and Privacy*, Springer, 322–339.

Weatherspoon, H., Ganesh, L., Marian, T., Balakrishnan, M. and Birman, K., 2009. Smoke and mirrors: Reflecting files at a geographically remote location without loss of performance. *FAST*, 211–224.

Weisburd, D., 2021. Talking to strangers: what we should know about the people we don't know.

Wierman, M. J. and Tastle, W. J., 2005. Consensus and dissention: theory and properties. *NAFIPS 2005-2005 Annual Meeting of the North American Fuzzy Information Processing Society*, IEEE, 75–79.

Wu, D., Liu, X.-d., Yan, X.-b., Peng, R. and Li, G., 2019a. Equilibrium analysis of bitcoin block withholding attack: A generalized model. *Reliability Engineering & System Safety*, 185, 318–328.

Wu, K., Dai, G., Hu, X., Li, S., Xie, X., Wang, Y. and Xie, Y., 2019b. Memory-bound proof-of-work acceleration for blockchain applications. *Proceedings of the 56th Annual Design Automation Conference 2019*, 1–6.

Wüst, K. and Gervais, A., 2016. Ethereum eclipse attacks. Technical report, ETH Zurich.

Wylie, J. J., Bigrigg, M. W., Strunk, J. D., Ganger, G. R., Kiliccote, H. and Khosla, P. K., 2000. Survivable information storage systems. *Computer*, 33 (8), 61–68.

Xiao, Y., Zhang, N., Li, J., Lou, W. and Hou, Y. T., 2019. Distributed consensus protocols and algorithms. *Blockchain for Distributed Systems Security*, 25, 40.

Xiao, Y., Zhang, N., Lou, W. and Hou, Y. T., 2020. A survey of distributed consensus protocols for blockchain networks. *IEEE Communications Surveys & Tutorials*, 22 (2), 1432–1465.

Yan, X. and Jia, M., 2018. A novel optimized svm classification algorithm with multi-domain feature and its application to fault diagnosis of rolling bearing. *Neurocomputing*, 313, 47–64.

Yang, F., Zhou, W., Wu, Q., Long, R., Xiong, N. N. and Zhou, M., 2019. Delegated proof of stake with downgrade: A secure and efficient blockchain consensus algorithm with downgrade mechanism. *IEEE Access*, 7, 118541–118555.

Ye, C., Li, G., Cai, H., Gu, Y. and Fukuda, A., 2018. Analysis of security in blockchain: Case study in 51%-attack detecting. *2018 5th International Conference on Dependable Systems and Their Applications (DSA)*, IEEE, 15–24.

Yin, J., Wang, C., Zhang, Z. and Liu, J., 2018. Revisiting the incentive mechanism of bitcoin-ng. *Australasian Conference on Information Security and Privacy*, Springer, 706–719.

Yin, M., Malkhi, D., Reiter, M. K., Gueta, G. G. and Abraham, I., 2019. Hotstuff: Bft consensus with linearity and responsiveness. *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing*, 347–356.

Yuan, C., Xu, M.-x. and Si, X.-m., 2017. Research on a new signature scheme on blockchain. *Security and Communication Networks*, 2017.

YUEN Man-Ching, C., Lau, K.-M. and Ng, K.-F., 2020. An automated solution for improving the efficiency of cryptocurrency mining.

Zadeh, L. A. and Aliev, R. A., 2018. *Fuzzy logic theory and applications: part I and part II*. World Scientific Publishing.

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

Zampaki, T., 2018. Debating attributes: Ibn rushd (averroes) vs. al-ghazālī. *Proceedings of the XXIII World Congress of Philosophy*, volume 17, 53–61.

Zargar, F. N. and Kumar, D., 2019. Informational inefficiency of bitcoin: A study based on high-frequency data. *Research in International Business and Finance*, 47, 344–353.

Zhang, R., Xue, R. and Liu, L., 2019. Security and privacy on blockchain. *ACM Computing Surveys (CSUR)*, 52 (3), 1–34.

Zhang, S. and Lee, J.-H., 2019. Double-spending with a sybil attack in the bitcoin decentralized network. *IEEE transactions on Industrial Informatics*, 15 (10), 5715–5722.

Zhou, H., 2019. *Learning Blockchain in Java: A step-by-step approach*. Amazon.

Zhou, P., Fang, X., Fang, Y., Long, Y., He, R. and Han, X., 2017. Enhanced random access and beam training for millimeter wave wireless local networks with high user density. *IEEE Transactions on Wireless Communications*, 16 (12), 7760–7773.

Zhou, Q., Huang, H., Zheng, Z. and Bian, J., 2020. Solutions to scalability of blockchain: A survey. *Ieee Access*, 8, 16440–16455.

Zhou, Z.-H., 2021. Why over-parameterization of deep neural networks does not overfit? *Science China Information Sciences*, 64 (1), 1–3.