UNIVERISTY OF PIRAEUS - DEPARTMENT OF INFORMATICS

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ – ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

MSc «Cybersecurity and Data Science»

ΠΜΣ «Κυβερνοασφάλεια και Επιστήμη Δεδομένων»

MSc Thesis

Μεταπτυχιακή Διατριβή

Thesis Title: Τίτλος Διατριβής:	**An analysis of consensus mechanisms for Blockchain** Ανάλυση των μηχανισμών συναίνεσης για το Blockchain
Student's name-surname: Ονοματεπώνυμο φοιτητή:	**Nasopoulos Leonidas** Νασόπουλος Λεωνίδας
Father's name: Πατρώνυμο:	**Athanasios** Αθανάσιος
Student's ID No: Αριθμός Μητρώου:	ΜΠΚΕΔ21038
Supervisor: Επιβλέπων:	**Panayiotis Kotzanikolaou, Professor** Παναγιώτης Κοτζανικολάου Καθηγητής

November 2024/ Νοέμβριος 2024

3-Member Examination Committee

Τριμελής Εξεταστική Επιτροπή

Kotzanikolaou Panagiotis	**Patsakis Constantinos**	**Psarakis Michael**
Professor	Associate Professor	Associate Professor

Acknowledgements

I would like to express my gratitude to my supervisor, Professor Panayiotis Kotzanikolaou as well as Vaggeli Malama for their guidance, help and patience during the whole project. I would also like to thank my family and friends for their support.

Abstract

Consensus processes are essential protocols employed to achieve agreement among several entities. Recently, these processes have garnered considerable interest as an essential component of Blockchain systems, tasked with documenting and authenticating transactions within a network. They serve as authentication algorithms that verify the legitimacy of each new block (transaction) prior to its incorporation into the Blockchain. For a consensus mechanism to be deemed effective, it must inhibit bad actors from modifying, removing, adding, or duplicating any transaction within the Blockchain. A consensus mechanism ensures the dependability, confidentiality and completeness of a distributed system. Broadly, consensus protocols can be grouped into two categories: proof-based protocols and Vaivdene protocols. Proof-based protocols operate through competition of the mining nodes to be the first to solve a mathematical puzzle in a given mining round. On the other hand, the voting based protocols operate through rounds of elections. An accounting node is selected by way of vote from all competent mining nodes.

Importantly, the node that is the first to gather the required number of votes in quorum is chosen to endorse the new block. The goal of this study is to provide more complete insight into the structure of the main consensus protocols of both types and to give graphical illustrations to make better understanding of how they work done. Lastly, the issues of the components of the protocols will be addressed and analyzed with respect to the security level and possible use cases.

The objectives of this thesis are rather specific. It is particularly desired that this thesis will fill an existing gap in the literature so that it is able to break down the available consensus amending mechanisms. Additionally, this research examines how the mechanics and security of various consensus mechanisms differ, especially in terms of scalability and attack deterrence. Moreover, since it uses graphs, this project aids in comprehension and makes the intricate details of these protocols less daunting. Furthermore, the research analyzes the vulnerabilities of respective mechanisms in terms of these protocols' applicability in Blockchain systems especially for financial infrastructures, supply chain, or decentralized applications. Last but not the least, this investigation presents a number of comments conducive to enhancing the design of effective and secure Blockchain systems.

.

Contents

List of Figures

List of Tables

1. Introduction

Blockchain technology is currently considered one of the most impactful technologies. This phenomenon transpires as it profoundly transforms the execution of financial transactions by obviating the necessity for a centralized authority. Although mostly utilized in cryptocurrencies, especially Bitcoin, Blockchain technology has been embraced by various areas, including healthcare and the Internet of Things, due to its numerous advantages [111]. Blockchain technology was launched in 2008 with the creation of Bitcoin by the pseudonymous S. Nakamoto, and its inaugural practical use occurred in January 2009. After Bitcoin, many projects followed, such as Litecoin, NameCoin, PrimeCoin, Ripple. The research on cryptocurrency has made the global financial system to evolve more rapidly as well as brought interesting progress in the development of blockchain technology. The Emergence of Ethereum blockchain platform in 2013 with the idea of web 3:0 marks the beginning of the second generation of blockchain where Ethereum was used as an application platform rather than bitcoin. The core breakthrough of Ethereum is represented by smart contracts which expanded the application of Blockchain technology beyond digital currencies into traditional business sectors that required contractual relationships. Later, other enterprise blockchain solutions were created such as hyper ledger fabric, quorum, codra, etc. With its ability to be a record that cannot be changed and be relied on, Blockchain emerged as a Central technology enabling the construction of trust in a digital world, that announced the birth of Blockchain 3.0. Blockchain is widely regarded as a reliable framework for data sharing that receives considerable endorsement. Entities outside of the cryptocurrency realm have effectively utilized blockchain technology in numerous spheres of corporate intelligence and business processes, copyright and content transactions, including furnishing privacy and secure transmission in IoT systems, data secure identification in the IoV, metaverses, social networks, NFTs, public administration, supply chain management, etc. [18]. Along with the growth in different blockchain application, the need for blockchain technology has also increased. Every type of blockchain platform and its applications has its particular consensus mechanism to employ, particularly in financial and business intelligence environments where security, scalability, throughput and low latency become important in enhancing business functions and reducing costs.

It is worth mentioning the work of Vukolić [94], who shared insights into applicable areas and parameters like security, efficiency, and scalability, while pointing out the importance of consistency in the choice of the consensus algorithm. Zheng [116] have also addressed the topic of Vukolić with an in-depth review of the blockchain architecture and a comparison of the different consensus algorithms. Viriyasitavat [92] also focusing there included discussions on how the consensus mechanisms would improve collaboration, sharing of knowledge and decision making in Business Process 4.0 enabled by the blockchain. Following that development, in the work of Biswas [10], a new consensus mechanism PoBT – proof of block and trade purposefully designed for the IoT blockchain was presented, and its advantages compared to existing ones related to business process reengineering were justified. Studies by Xu [106]) and Wang [109] continued this trending research direction and proposed federated learning aided extensions to the consensus algorithms and thus to the blockchain technology itself.

Consequently, there has been a lot of work directed towards solving challenges of previous consensus methods, especially in the cases where applications such as financial technology require very high security, scalability, throughput, and low latency. More sophisticated consensus techniques are crucial regarding the application of blockchain technology in various fields that need accurate and reliable information for making rational decisions. This remains hovering among the major areas of effort in research.

The concept of consensus mechanisms derives from concepts found in distributed systems. Investigation of the problem of reaching distributed agreement was considered as early

as the 1980s. Mechanism like Paxos, Raft, and Practical Byzantine Fault Tolerance (PBFT) [1] was implemented almost immediately after their proposals and has been evolving ever since. Within traditional design structures, these mechanisms have satisfied all the robustness requirements for them. In real life, an optimally designed consensus system can increase the volume of transactions per system within the minimum time required for the confirmation of transactions. Based on access permissions, consensus mechanisms belong to either of these two categories; the permissioned systems or the permissionless systems. Permissioned protocols function with institutional networks; therefore, all the nodes in the system must identify and recognize each other's identities in order to maintain impartial control. Unlike the former, client or member nodes are not crooned in a permissionless setting and therefore increase privacy and decentralization. Public Training consensus protocols employed in public block chains often employ permissionless systems whose characteristic example is the proof of work' of bitcoin. In PoW systems, any node can join or leave at will, and miners must prove their work to propose a new block, a process that demands substantial computational resources.

A wide variety of consensus mechanisms has been developed, and researchers continue to investigate effective attacks, defense strategies, and security frameworks for these mechanisms [19]. That being said, it should be noted that no one consensus mechanism is effective in every situation; a given mechanism may be appropriate for one application, but not for another. There remains a wider exploration of consensus mechanisms that needs to be undertaken. Some authors, for instance, Zheng et al. (2017) carried out a study of the blockchain technology in question and compared the classical mechanisms of consensus on different platforms of the blockchain. Still, consensus mechanisms have not been exhaustively studied.

Nguyen [63] cegorized blockchain consensus mechanisms into two broad types: proof-based and vote-based, explaining the strengths and weaknesses of each. Fu [29] further refined this categorization into four distinct types: leader-based mode, vote-based mode, committee plus voting mode, and fair accounting method. Despite the fact that this classification is comprehensive, a simplified version of it could be provided for beginners in the discipline.

The critical aim of this project is to enhance the comprehension of blockchain technology, especially with regard to examining the design of blockchain systems. In this regard, a thorough inquiry will be conducted in relation to consensus mechanisms on the aspect of its fundamental role within blockchain networks, and hence towards the functionality of the distributed ledger.

1.1 Relevant Methodology

This study utilizes a thorough literature review methodology to gain an enhanced understanding of consensus mechanisms in blockchain technology. It involves collecting information, examining it and summarizing available literature and articles on various consensus mechanisms. This method provides a well-rounded approach in the investigation of the topic by making use of as many informative sources as possible, including scholarly articles, technical papers, industry reports, and case studies. The first step in the procedure is finding and gathering relevant materials. In this instance, it is the databases IEEE Xplore and Google Scholar that will be searched using: "blockchain consensus mechanisms", "Proof of Work", "Proof of Stake", "Delegated Proof of Stake", "Proof of Authority" and "Proof of Burn" search terms.This phase seeks to assemble an extensive compilation of sources addressing the range of consensus methods utilized in blockchain technologies.

The search queries yielded around 310 publications and articles. A set of inclusion criteria was implemented to refine this extensive corpus of material. The documents that focused on consensus processes and included description of principles in operation, security features as well as energy consumption, scalability and decentralization were given preference. It was important

that all chosen articles be in English so that consistency of interpretation and analysis is guaranteed.

Along with the inclusion criteria, the additional exclusion criteria were applied in order to narrow down the sample even more. Papers not explicitly centered on blockchain or consensus processes were omitted to ensure relevance. Forty-three papers were eliminated. Language specific papers that were not in English were also ruled out for consideration. In this case, 36 papers were omitted. Duplicate records and old references, especially those published before 2015. One hundred nine papers were eliminated.

Lastly, papers that showed either no technological depth or papers that were too focused on technology demonstrations and included commercialization but did not explain the basic technology, were also excluded. Sixty-six papers were excluded. Subsequent to the application of these inclusion and exclusion criteria, the original collection of 310 publications was markedly diminished to 56 articles.

Upon the collection of literature, the subsequent phase entails a comprehensive investigation of each consensus process. his assessment evaluates the functionality, security parameters, energy consumption, scalability, and degree of decentralization of each protocol. An in-depth study of these aspects will determine the benefits and the drawbacks as well as any compromises involved with the different procedures of reaching a consensus. It revolves on identifying present trends and future trends within the consensus technologies. A research of existing papers, white papers and reports by the blockchain development and research community will make it possible to achieve this goal. his aim is to forecast future developments and provide a more forward-looking perspective on the evolution of consensus mechanisms in blockchain systems

In this respect, the analysis is focused on the strengths and weaknesses of the current research studies dedicated to examining the workings of consensus mechanisms within blockchain technology and particularly offers a critical synthesis of the existing body of the literature. Consensus mechanisms provide another smart solution for driving blockchain quality forward that maintains the integrity of the record and guarantees no data was lost or abused. Consensus mechanisms have significant potential for many interdisciplinary applications, ranging from social-oriented cooperatives to attention-focused DAOs and fintech platforms reliant on the economic perspectives of the decentralized ecosystem.

The analysis employs a literature review strategy, whereby, the author critically and hermeneutically evaluates the works of various credible researchers. It helps set the boundaries of what is new about the study and what it builds upon, and furthers the opportunity to understand the narrow realm that the researcher lobbies for. Using this approach in this case enables an understanding of what previously focused studies contributed to the field and which gaps this particular focus instituting. As a future research focus, consideration should be given to the study of the role consensus mechanisms in blockchain technologies within the framework of new cooperatives, technological neutrality, and their ability to overcome non-collaborative trust issues.

2. Overview of Blockchain Technology

A blockchain, a type of distributed ledger, consists of an increasing series of entries, or blocks, securely linked by cryptographic hashes. Each block has transaction data (often depicted as a Merkle tree with leaves representing data nodes), a timestamp, and a cryptographic hash of the preceding block. Each block links to its predecessors, forming an efficient chain with the

incorporation of each new brick. Cryptocurrencies represent a significant application of blockchain technology.The concept of Blockchain as a digital, distributed, and decentralized data structure involves the creation of transaction blocks that autonomously record digital transactions without dependence on a central authority [67]. Data regarding new transactions is integrated into the chain subsequent to its encryption and certification by the majority of participating agents. Subsequent to creation, each block is timestamped and cryptographically connected to preceding blocks to verify the order of documented transactions. Blockchain operates as a decentralized database, consisting of an increasingly extensive record of transactions arranged in a sequential manner. It maintains the anonymity of contributors through digital signatures.

2.1 Core Components of Blockchain

Blockchain is a decentralized digital ledger implemented in a distributed fashion, rendering it impervious to tampering and obviating the necessity for a central authority. A blockchain consists of multiple essential components. When a node tasked with content publication contributes a block, it incorporates validated transactions into the blockchain. A block comprises a block header that contains metadata, including the block number, hash of the preceding block, timestamp, nonce, and block size. The block data comprises a collection of authenticated transactions. The nonce is essential in the Proof of Work (PoW) consensus method, as the validation of a block depends on its value [105].

Cryptographic hash functions are essential for blockchain security, transforming input data of arbitrary size into a unique fixed-size output known as a hash or message digest. The hashing method guarantees that even a minor alteration in the input substantially modifies the hash value, a phenomenon referred to as the avalanche effect [2]. SHA-256 is a prevalent hash function utilized in blockchain systems such as Bitcoin, where a hash functions as a reference to the subsequent block on the chain. Any alteration to a block or chain leads to a corresponding update in the hash value [111]. The Merkle Tree is a vital cryptographic component in blockchain, serving as a data structure that arranges cryptographic hashes of transaction blocks in a hierarchical manner. The framework enables any modification in a transaction, including additions or deletions, to disseminate along the entire hash chain, thereby augmenting security.

Asymmetric-key cryptography, also called public-key cryptography, plays a critical role in securing data in blockchain systems. It involves two keys: a private key, used for authenticating transactions and generating digital signatures, and a public key, which is publicly accessible and used to verify those signatures. The public key confirms that the person controlling the transaction holds the corresponding private key. Blockchain wallets store both the private and public keys of a user. If the private key is lost, access to the related digital assets is permanently lost, as transactions signed with the private key are immutable and irreversible.

Nodes, which are crucial components of blockchain networks, operate as independent entities that relay transactions throughout the network. Each node stores, verifies, and distributes transactions. In Proof of Work (PoW)-based blockchains, these nodes, known as miners, solve complex computational problems to add new blocks to the chain. The blockchain operates on a distributed ledger system, where all participants keep a copy of the ledger. This decentralized method ensures that all users collectively agree on the transaction history and network status [69].

Blockchain networks rely on consensus mechanisms to eliminate the need for third-party validation, ensuring that participants collectively agree on the system's status. Consensus is key to both adding blocks and maintaining the system’s integrity. These mechanisms must overcome three main challenges: reaching consensus among honest nodes, verifying the block proposed by one node, and ensuring timely completion of all processes [35]. Validators in blockchain

networks, particularly in PoW systems, are sometimes rewarded for maintaining the system, with honest miners receiving compensation for their contributions [12].

Blockchain networks utilize peer-to-peer (P2P) protocols, allowing nodes to communicate directly without relying on a central server. This decentralized structure improves security by mitigating risks associated with single points of failure [10]. Pending transactions are managed by a memory pool, or mempool, until they are added to a block and then removed from the mempool of the respective node [4].

2.2 Block

A block is composed of two main parts: the block header and the block body. According to Hameed (2019), the block header includes the following elements:

(i) Block version: Specifies the set of rules used for validating the block.

(ii) Merkle tree root hash: The cryptographic hash result that represents the combined hash value of all transactions in the block.

(iii) Timestamp: Denotes the current time in seconds since January 1, 1970, using Coordinated Universal Time (UTC).

(iv) nBits: Defines the target threshold required for a valid block hash.

(v) Nonce: A 4-byte field that typically starts at 0 and increases with each hash computation (explained further in Section III). The parent block hash is a 256-bit cryptographic hash that links to the previous block.

The block body comprises a transaction counter and an assemblage of transactions. The total number of transactions that can be accommodated within a block is dictated by the block size and the size of each transaction. Blockchain technology employs asymmetric cryptography to verify transactions, thereby guaranteeing their authenticity in potentially unreliable environments. In these situations, a digital signature utilizing asymmetric cryptography is important. A succinct elucidation of digital signatures accompanies this discourse.

A block comprises a header and a body. The header, as delineated by Hameed [38], comprises essential elements including the block version (which specifies validation criteria), the Merkle tree root hash (representing the aggregate hash of all transactions), the timestamp (in UTC since January 1, 1970), the nBits (target value for a valid hash), and the nonce (a 4-byte field initialized at zero). The parent block hash functions as a reference to the preceding block. The body, conversely, comprises a transaction counter and the transactions themselves, with their capacity dictated by block and transaction size. Asymmetric cryptography guarantees transaction legitimacy by employing digital signatures in untrustworthy contexts.

2.3 Digital Signature

Every user possesses a distinct pair of private and public keys. The private key is utilized to authenticate transactions. Digitally signed transactions are disseminated over the whole network. A digital signature typically consists of two phases: the signing phase and the verification phase. User X aims to communicate a message to User Y. During the signing procedure, X employs her private key to encrypt her data and transmits both the encrypted output and the original data to Y. During the verification process, Y authenticates the value by checking it with X's public key. By doing so, Y can easily determine if the data has been modified. The dominant

cryptographic algorithm for digital signatures in blockchains is the elliptic curve digital signature algorithm (ECDSA) [13].

A digital signature is a cryptographic method employed to authenticate and ensure the integrity of electronic documents, communications, or transactions.. Digital signatures enable the authentication of transactions between two parties in blockchain technology, hence eliminating the need for intermediaries.

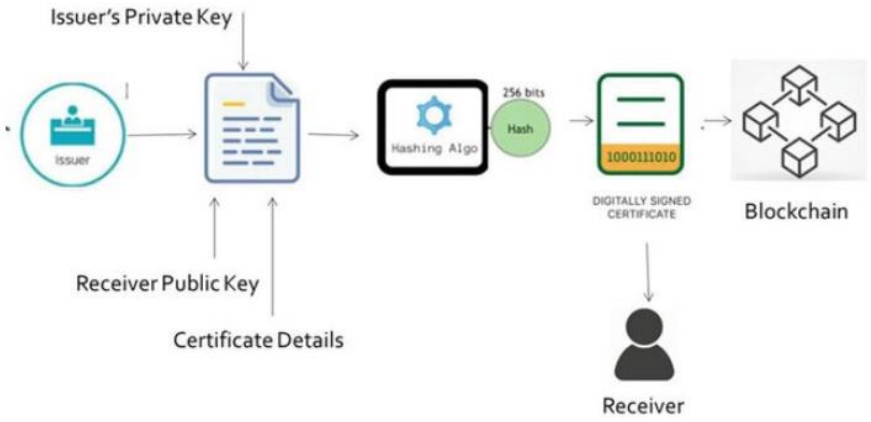

Figure 1:Framework of digital signature on blockchain

2.4 Characteristics of Blockchain

2.4.1Decentralization

Centralized systems depend on a trusted third party such as a central bank to authorize and record transactions. Central servers become expensive and slow in this architecture. By requiring consensus algorithms, blockchain can validate transactions across a distributed network within a decen4tralized ledger system replacing the need for intermediaries (Nakamoto, 2008). It is the honest miners who validate transactions, reject fraudulent ones and maintain the integrity of the network.

2.4.2 Continuity and Immutability

Once transactions are recorded on the blockchain, it is nearly hard to delete or modify them. It promptly identifies errors in blocks, hence preserving the system's visibility and trustworthiness. This is exemplified in Bitcoin by the necessity for each transaction to cite prior unspent outputs using the Unspent Transaction Output (UTXO) mechanism. Upon the creation of a new transaction, the outputs are designated as "spent". This approach facilitates the rapid confirmation and monitoring of transactions.

2.4.3 Anonimity and Treceability

Each user interacts with the blockchain via a unique cryptographic address, which helps to obscure their identity. Although this offers a degree of anonymity, the blockchain cannot guarantee complete privacy due to its inherent limitations.

2.4.4 Security

Distributed ledgers are acknowledged for their superior security protocols. Participating agents employ cryptographic encryption to generate transactions. The public and private keys linked to transacting agents guarantee integrity and validation methods that prevent manipulation [48]. Cryptographic hashing functions, which provide unique identifiers of fixed length regardless of the input, constitute the essential elements of blockchain security. Each hash serves as an identification for a block and correlates to the hash value of the preceding block. The hash function is employed in a consensus process for the validation of current transactions.

2.4.5 Immutability

A notable characteristic of blockchain is its immutability, as the distributed ledger cannot be altered by any party. The blockchain is immutable; transactions cannot be modified, erased, or reversed until over 51% of the nodes consent to the alteration. This necessitates the assailant to gain control of over fifty percent of the nodes, which is quite unlikely. Nevertheless, while violating the immutability of the blockchain is deemed unlikely and complex, it remains feasible with adequate resources [79]. The concept of immutability applies to both the data and the code in the distributed ledger. Blockchain regards the immutability of data records as indisputable; yet, data may be modified and erroneous before its incorporation into the chain. Consensus procedures are utilized for data entry verification, albeit they are limited by the dependability of participant consent. Conversely, the code's immutability is compromised by the acknowledgment that no code is created without flaws, neglecting to encompass all operational requirements. This issue is demonstrated by the continual alteration of blockchain code in numerous instances (Noyes, 2016).

2.4.6 Speed

The distributed ledger has mitigated the sluggish transactions inherent in the traditional banking system. The pace of blockchain transactions is influenced by block size, transaction fees, and network congestion [108]. Blockchain enhances global transactions by reducing the block time, the interval necessary for the addition of a new block. Furthermore, the transmission duration diminishes as block size increases, hence enhancing transaction velocity [28].

2.4.7 Consensus

Consensus methods have been integrated into blockchains as a fault-tolerant method for transaction validation. The consensus is employed to maintain accord among the nodes in the network.As the network expands, the quantity of nodes increases, making agreement increasingly difficult to attain. Public blockchain necessitates user involvement for the verification and validation of transactions. Blockchain, as a dynamic and self-regulating system, necessitates the integration of a secure way to verify transaction authenticity, hence enabling consensus among participants. Diverse consensus methodologies have been developed, differing in their foundational ideas and applications [42]. It is crucial to emphasize that blockchain is inherently unchangeable. Once a transaction is recorded on the blockchain, it becomes immutable and resistant to any modification or manipulation. Organizations aiming for enhanced reliability and transparency may utilize blockchain technology to attract clients. Furthermore, blockchain operates in a decentralized manner, markedly diminishing the risk of a single point of failure.

Miners possess the power to autonomously execute smart contracts upon their implementation on the blockchain. Although blockchain technology has considerable promise for enhancing future Internet services, it now faces various technological challenges. Scalability presents a significant challenge. The maximum capacity of a Bitcoin block is 1 megabyte, and a new block is typically generated through mining around every 10 minutes. The Bitcoin network processes just 7 transactions per second, which is inadequate for high-frequency trading. Utilizing larger blocks results in increased storage capacity and diminished network propagation speed. This will result in a gradual centralization, as a diminishing number of users will be motivated to maintain such a vast network. Consequently, the difficulty of achieving an optimal equilibrium between block size and security has posed a considerable obstacle. Furthermore, empirical evidence indicates that miners can attain greater profits than considered equitable by employing a selfish mining approach [26]. Miners conceal their extracted blocks to maximize their possible profits. Thus, the continual emergence of branching obstructs the progress of blockchain development. Therefore, it is imperative to delineate specific actions to address this issue. Furthermore, research indicates that privacy infringements can occur in blockchain systems, even when users solely employ their public and private keys for transactions [9]. Moreover, contemporary consensus procedures like as proof of labor and proof of stake are encountering significant challenges. The proof of work consensus mechanism expends considerable electrical energy, whereas wealth accumulation among the affluent may transpire in the proof of stake consensus process.Blockchain is a linear sequence of blocks that contains an extensive compilation of transaction data, akin to a traditional public ledger [49].

2.5 Taxonomy of blockchain systems

Blockchain systems are categorized into three main types: public blockchain, private blockchain, and consortium blockchain [13],[14],[15]. In the case of public blockchains, every record is available for all to see, thus any individual can take on the role of a consensus participant in that blockchain. In the scenario of a consortium blockchain only the specific set of nodes are involved in the consensus mechanism. For a private blockchain, the only permitted nodes in the consensus mechanism are the nodes belonging to some organization. A private blockchain is seen as a centralized network due to its complete governance by a singular entity. The consortium blockchain, comprising multiple organizations, exhibits a level of decentralization by employing a limited number of nodes assigned to achieve consensus.

Taxonomy of blockchain systems	
public blockchain	All records are openly visible and available to the broader public.
private blockchain	Only nodes that come from a certain organization are allowed to take part in the consensus mechanism.
consortium blockchain	Formed by various companies, demonstrates a degree of decentralization as it only includes a restricted number of nodes that are selected to establish consensus.

Table 1:Taxonomy of blockchain systems

Transactions on a public blockchain are transparently available to the public; however, the accessibility of transactions on a private or consortium blockchain may vary. The decentralized architecture of a public blockchain significantly reduces the probability of transaction manipulation. Transactions on a private or consortium blockchain are susceptible to manipulation due to the restricted number of participants. The dissemination of transactions and

blocks on a public blockchain network is hindered by the presence of several nodes. The capacity for transaction processing is limited, leading to considerable delays. Decreasing the quantity of validators can improve the efficiency of consortium and private blockchains. The primary difference among the three sorts of blockchains is in their degree of centralization. Public blockchains are tolerant of decentralization, consortium blockchains are relatively centralized while private blockchains are centralized controlled by one organization. The process of consensus involves a decision structure whereby there is an attempt by a team to come to an agreement or common understanding. All countries can work as participants in the public blockchain consensus mechanism. Unlike public blockchains, consortium blockchains and private blockchains are both permissioned. The characteristic of public blockchains where there is unrestricted participation of users improves the level of user buy in and the development of active communities. A number of the public interdependent chains are continuously coming up. Consortium blockchain can be applied for different business purposes. As of now, Hyperledger is developing blockchain structures for business associations [53]. The development of consortium blockchains has been enabled by Ethereum.

2.6 Blockchain Stack

A transition is taking place in the distribution of value from the internet stack to the blockchain stack under the blockchain framework. This produces a significant protocol layer and a comparatively tiny application layer. The integration of speculative tokens establishes a cycle that promotes application development and increases the value of the protocols. [60]. The blockchain architecture comprises three fundamental layers: the Internet Layer, the Blockchain Layer, and the Application Layer [3].The user interacts with the protocol via applications, namely decentralized applications (dApps) developed by programmers. The blockchain technology stack consists of four essential components: shared data, protocol, platform, and commodities. The information on the transactions is stored in a distributed book in a hash pattern which constitutes the basic block of the same addressed. Many models incorporate regulations that facilitate agreement, create motivators, and increase participation. Such systems act as a groundwork for creating products available for end-users.. Ethereum and NEO serve as prime examples of such platforms. The offerings consist of decentralized applications (dApps) and smart contracts [69].

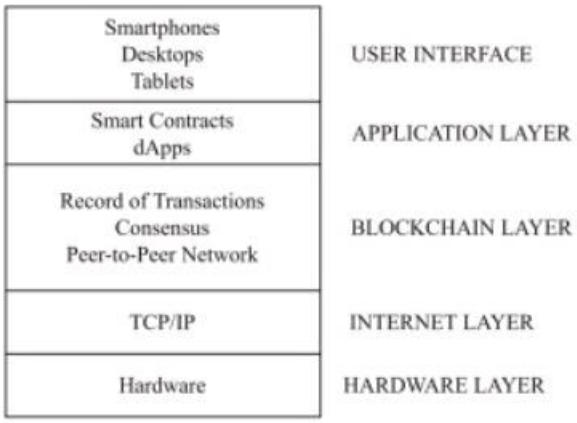

Figure 2:Ethereum's Blockchain stack and similar applications

2.7 Advantages and Disadvantages of Blockchain

The key for Blockchain is its decentralized structure, eliminating any of potentially the absolutist kind of power structures. Every action is virtually stored in the Blockchain, and every action is available for all members in the so called Blockchain. This recording illustrates the transparency and dependability of Blockchain [5], [6] . A further advantage of blockchain technology is its intrinsic immutability and resilience. Modifying or erasing data from the blocks of the Blockchain is highly impractical. If an intruder possesses the computing capability to alter or erase data across all devices within the Blockchain prior to the recording of the subsequent block, it becomes feasible to edit or eliminate information in the Blockchain. The lesser the nodes in a blockchain network, the easier it is to compromise that network. In vice versa, when more computers engages themselves in the blockchain, this system is comparatively safe as well as more transparent. Because a cryptographic hash chain is used in the construction of the blockchain, it is viewed as impervious to alteration and as if no amount of effort would ever be capable of erasing. Every block in the Blockchain contains the hash value of the previous block. Blockchain as an operational system has its strong points in the fundamental underlying nodes. The participatory security of the block chain system is conditioned by nodes quality. The bitcoin blockchain resilient will offer nodes pay to be involved in the network. This makes it practically impossible in a blockchain network without relative rewards for its user nodes. This means that it is not a computing network in which the participation and input of nodes are required for the system's activities to be carried he reliability, security, and integrity of a distributed system. Consensus protocols are primarily categorized into two types: proof-based protocols and voting-based protocols. Proof-based protocols function through competition among mining nodes that strive to be the first to solve a complex mathematical problem in each mining round. Conversely, voting-based protocols function through elections conducted in rounds. The accounting node is chosen by a voting process including all eligible mining nodes.out. A distributed system wants to make sure that any such transaction is done and the relevant verification protocols are followed and ensure that the history of each transaction is maintained in a tamper-proof manner.

Although they resemble blockchain activities, they lack mutual support, synergy, and parallelism. Although blockchain is a distributed network, it lacks the characteristics that render distributed computing solutions particularly beneficial for enterprises. Blockchains demonstrate worse scalability relative to their centralized equivalents. The execution of transactions on the Bitcoin network depends on the degree of network congestion. This issue pertains to the scalability issues of blockchain networks. The probability of network congestion escalates with the addition of nodes or users. Blockchain technology was initially introduced with Bitcoin. It employs the Proof-of-Effort consensus mechanism, which relies on the arduous efforts of the miners.

Miners are driven to resolve complex mathematical equations. These complex mathematical problems are impractical for application due to their high energy consumption. The miners must resolve the challenges that arise with each new transaction added to the ledger, necessitating considerable energy consumption. However, not all blockchain technology function identically. The problem has been addressed by other consensus methods. For example, same concerns do not occur in permissioned or private networks due to the reduced number of nodes. Furthermore, they utilize efficient consensus-building strategies as they do not require global agreement. Permissioned networks consume less energy than public networks, which can require substantial energy for operational maintenance. The immutability of data has consistently been a primary limitation of blockchain technology. It is clear that it benefits several systems, including financial and supply chain systems. Nonetheless, an examination of network operations indicates that immutability depends on the fair distribution of network nodes. The incapacity to delete recorded data is an additional concern. Every person on Earth has the right to personal privacy. However, if the same individual utilizes a blockchain-based digital platform, he will be unable to

eliminate his trace from the system at his discretion. He is unable to delete his recollection, so infringing against his right to privacy [51]. Furthermore, the ineffectiveness of blockchain data storage may lead to storage challenges for several nodes seeking to join the network. Given that nodes are required to replicate new data following modifications, it is evident that a more efficient management technique is essential. Moreover, the blockchain's size expands with the incorporation of additional transactions and nodes. The network's velocity declines with continuous expansion. In business blockchains, the simultaneous requirement for speed and security is inadequate [50]. Blockchain technology offers superior security compared to traditional solutions. This does not suggest that it is completely dangerous, however. The blockchain network is vulnerable to several types of breache

Empowering individuals to function as their own banks is crucial for achieving decentralization in blockchain technology. This also raises an additional worry. Private keys are essential for users to access their assets or information stored on the blockchain. The user must accurately document this, as it is generated during the wallet creation process. They must also ensure that it is not 0disseminated to others. Their financial stability is jeopardized if they fail to comply with this criterion. They will permanently forfeit access to the wallet if they lose the private key. A significant limitation of blockchain technology is its reliance on human involvement. The lack of technological expertise among consumers, coupled with an increased likelihood of errors, constitutes a major disadvantage. Decentralization's objective is compromised when managed by a centralized authority [36]. The use of blockchain technology entails considerable initial costs. Although most blockchain technologies, like Hyperledger, are accessible without cost, the deploying firm must invest substantial financial resources. The implementation and maintenance of a blockchain project is intricate. To execute the method, the organization must have considerable competence. A significant drawback of blockchain is the necessity to involve numerous industry experts. Furthermore, they must ensure that the management team comprehends the complexities and ramifications of a blockchain-based enterprise by offering training on the implementation of blockchain technology [34] represents an additional limitation of blockchain technology. Numerous blockchain networks exist that seek to tackle the DLT challenge with diverse techniques. Consequently, interoperability issues arise when these chains fail to interact effectively. The interoperability issue persists between blockchain-based solutions and traditional systems [33].

Advantages	Disadvantages
Decentralization: The design of the blockchain system ensures that no individual or organization can dominate the entire system. Power is thus decentralized, which enhances reliability and security.	**Not Distributed Computing System:** Although decentralized, blockchain does not possess the efficiency of distributed computing, hence constraining its applicability in enterprise settings.
Transparency: Every member can view every transaction, which provides a level of trust among members.	**Scalability**: Blockchain networks encounter difficulties in processing substantial transaction volumes, resulting in congestion.
Immutability: The accuracy of records is secured as data once validated cannot be changed, providing historical records and security.	**Energy Consumption**: Mechanisms like Proof of Work consume significant energy, raising environmental concerns.

Security: Cryptographic techniques including encryption and hashing methods are used in blockchain technology to secure transactions and ensure the integrity of the information.	**Data Immutability:** Although beneficial, immutability may compromise privacy, as users are unable to delete their data from the blockchain. .
Resistance to Attacks: In comparison to centralized databases, blockchain is less susceptible to assaults since it is decentralized.	**Inefficient Blockchains:** The current structure may not be optimal for complex applications, requiring additional improvements.
Accountability: Blockchain enhance accountability by permanently recording acts that can be traced to entities	**Private Keys:** Users endanger their ability to access their assets upon losing their private keys, as no recovery mechanism exists.
Invulnerability: Built on cryptographic chains, blockchain data is secure and tamper-resistant, enhancing data protection.	**Cost and Implementation**: Using blockchain technology requires high upfront costs and specialized expertise.
	Expertise Knowledge: Blockchain projects demand knowledgeable professionals, complicating its adoption.

Table 2:Advantages and Disadvantages of Blockchain

3. Consensus Protocols

In a blockchain, transaction validation and confirmation occur through the collective consensus of all network participants, assuring maximum security, reliability, and immutability. This phenomenon is commonly referred to as consensus in the context of blockchain technology. It operates in a decentralized and distributed manner [12]. Consensus enhances the reliability of the blockchain for decision-making purposes. Furthermore, eliminating centralized authority from the transaction system, such as centralized banking, reduces vulnerability and detrimental effects on consensus. Numerous consensus procedures have been developed to date. The implementation of a replicated log is crucial in certain consensus methodologies to guarantee that state machines or network nodes execute identical commands in a consistent and dependable order. If the log is identical, the state machines will perform the same instruction and yield comparable outcomes. This technique stays effective as long as the majority of devices operate correctly [92],[93]

3.1 Proof based Consensus Protocols

The principal objective of consensus mechanisms is to guarantee the liveness and safety properties of the distributed system. The protocol guarantees liveness by guaranteeing that consensus rounds consistently reach a conclusion, hence facilitating the continued incorporation of new blocks into the blockchain. The safety property ensures that all non-faulty participants own identical additional blocks and that a non-faulty participant initiated the block at the commencement of the consensus round. A fault-tolerant distributed system will operate correctly only if its consensus mechanism guarantees both safety and liveness. A fundamental problem in distributed systems is the inability to achieve consensus, often termed the FLP result [70] .The FLP result illustrates that the consensus problem cannot be deterministically resolved, even with a single crash failure, in an asynchronous network like the Internet. For numerous years,

consensus approaches have circumvented the FLP finding by presuming synchronous and partially synchronous communication systems, which provide differing degrees of certainty over message delivery during a consensus round. Classical consensus procedures largely emphasized safety while depending on communication technology to convey messages and guarantee liveness.

Protocols dependent on network synchronization are incompatible with the attributes of best-effort networks such as the Internet, where message delivery and routing cannot be guaranteed [41]. In the aftermath of Nakamoto's research, two alternatives have arisen to address the FLP theorem. The objective is to ensure safety, as previous methods have accomplished. The second objective is to guarantee liveness by developing a proof-based system capable of making judgments independently of synchronization.

Consequently, blockchain consensus algorithms are classified into two categories: deterministic and probabilistic consensus protocols. Protocols like Practical Byzantine Fault Tolerance (PBFT), BFT-SMaRt, Tendermint, and Ripple are founded on classical deterministic consensus mechanisms. These protocols emphasize safety over liveness, leading to reliable procedures that prevent forks. Unfortunately, deterministic protocols may cease to function if the communication system functions asynchronously [92],[93],[94]. Probabilistic consensus mechanisms, such proof of effort and proof of stake, emphasize rapid conclusion attainment, potentially at the expense of system consistency. The consensus leader in the probabilistic model is identified by any member who provides correct and irrefutable evidence, subsequently proposing the block. This technique removes the necessity for synchronous message exchanges but heightens the probability of many players simultaneously submitting proofs for different blocks, leading to a fork.

The system aims to diminish the probability of such occurrences and to create a mechanism for resolving forks in the blockchain, exemplified by the use of the longest chain rule in Bitcoin. The probabilistic consensus mechanism has high scalability, as it does not necessitate awareness of all members or the transmission of messages within the network to attain consensus. Therefore, this specific type of agreement is better suitable for public blockchains that include several participants. Consensus protocols based on proof, including Proof of Work (PoW), Proof of Stake (PoS), Proof of Burn (PoB), Proof of Authority (PoA), and Delegated Proof of Stake (DPoS), have been formulated through the probabilistic method. These protocols are extensively utilized in the majority of contemporary cryptocurrencies [70],[71].

3.1.1 Proof of Work(PoW)

Proof-of-work (PoW) is a consensus process employed by numerous cryptocurrencies to authenticate transactions and generate new blocks on the blockchain. Proof of Work (PoW) is a mechanism wherein miners vie to resolve intricate mathematical challenges to authenticate transactions and generate new blocks. The miner who first solves the riddle is awarded a specific amount of bitcoin. Proof of Work is designed as a secure and decentralized system that is difficult to manipulate or compromise. Miners must employ a designated amount of computational power and energy to solve the mathematical problem associated with a new block prior to its incorporation into the blockchain. The individuals and stakeholders whose interests are likely to be threatened by the technology are deterred from doing so. The principle of proof of work makes it such that the effort of miners is made tougher, little by little. Daunting as it may sound, because of rising sequential issues, the amount of processing power and energy that the miners have to solve the problems increases. This preserves the integrity of the blockchain and discourages bad actors from altering it. This algorithm is one of the important parts of a number of cryptocurrencies,

aimed at preventing attempts of double transactions, or fraud. The power tends to act as the water of the trees and as the power increases so does the confinement of the trees gets into the penetrating the power further. Actions meant to defraud the system through the blockchain become less practical because there is little room for malice. That ensures retention of the integrity of the blockchain and its transactions [31]. Main cryptography where rationale reveals from its function is concerned with verification of original units transfers to ensure that units are not spent twice. Such an approach has been proven to be efficient and reliable for achieving a safe and sharded decentralized database.

As the value of a cryptocurrency increases, more miners are incentivized to join the network, thereby bolstering its resilience and security. Owing to the significant computational power necessary, it is impractical for any individual or collective to breach the blockchain of a valuable cryptocurrency. Conversely, it is an energy-intensive process that may have challenges in scaling to accommodate the substantial transaction volume produced by smart contract-compatible blockchains such as Ethereum. As a result, other solutions have been suggested, the most prominent of which is known as proof of stake . Bitcoin, which is an online currency, uses log sheet technologies to operate. There is particular time at which transactions in a chain of blocks are made and a certain activity is executed after an approval is received. Every block consists of a number of transactions and considerable exertion is required to append this block to the blockchain. Each block is designated an index according to its hash value, and the hash value of the preceding block is integrated into each subsequent block. The act of incorporating a block is referred to as mining, and those who undertake this activity are called miners [40]. The miner is required to randomly choose a nonce value and compute the corresponding hash value. Should the computed hash value fall beneath a certain threshold, the block is incorporated into the blockchain. This is further corroborated by the additional miners in the network. The SHA-256 hash function is utilized in connection with bitcoin [88]. The computation of an exact hash is determined by establishing the target T value for every 2016 blocks. Occasionally, two miners may simultaneously add a block. When many miners simultaneously validate the same block, the block that achieves the highest level of synchronization within the network obtains unanimous consensus from all nodes in that network [36]. An examination of the Proof of Work (PoW) process reveals that it requires 10 minutes to produce a block and an extra 60 minutes (1 hour) to validate a single block. The building of six blocks is necessary to determine the time needed to certify a single block.Proof of Work (PoW) is utilized in Bitcoin and Ethereum. Ethereum functions as both a digital money and a platform for application development [14]. he image illustrates how the Proof of Work (PoW) consensus mechanism is supposed to work in the system of the blockchain network. The initiation of the process is done by a node that proposes a new block into the network. Such extra digits are known as nonces which are inserted on purpose to alter the hash value. Then, the node attempts to achieve a hash value that exceeds the predetermined threshold by varying the nonce and applying the hash function repeatedly. Upon reaching any of the given target values, the miner obtains the Proof of Work and hence is given the reward of a mining bonus. The final stage of the process is the binding the block into the chain, which preserves the competitive and secure nature of the blockchain. In figure 3 the function of PoW algorithm is depicted.

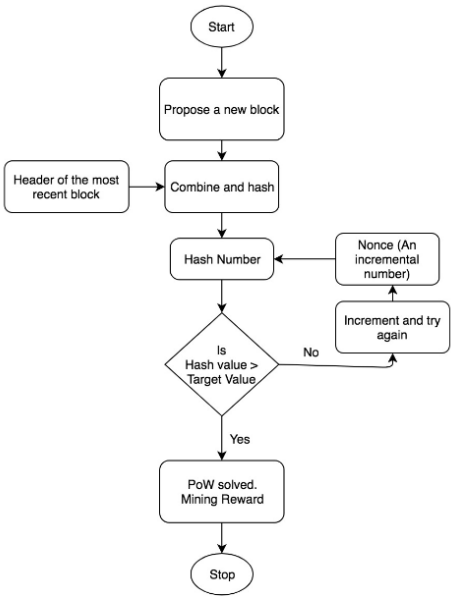

Figure 3:Proof of Work

3.1.2 Proof of Burn (PoB)

Iain Stewart is the originator of the concept of PoB. Distributed consensus serves as an alternative to Proof of Work (PoW) and Proof of Stake (PoS) [81]. In the consensus process, coins are annihilated, resulting in the generation of burn hashes by a technique unique to burn transactions [59]. The process does not require substantial computational resources or hardware. One calculation is required to derive the burn hash. A burn hash is calculated by multiplying the multiplier by the internal hash. The multiplier's value varies with each burn transaction. This multiplier results in the degradation of the depreciated currency. The cremation of currency demonstrates the user's commitment to the network, enabling them to participate in mining and transaction verification. The burning procedure can be accelerated with the help of native currencies like Bitcoin and others. Burning describes a method of sending certain coins to a relevant irrevocable online address. This ensures that the coins cannot further be used or traced. Similar to PoS, PoB engages block validators in a consensus resource allocation function.The only difference is that under Proof of Stake (PoS), a node can detach from the network and retrieve its cash for other purposes. In the PoB system, incinerated coins are irretrievable, resulting in their permanent loss. Coins utilizing the Proof of Burn (PoB) consensus technique include Counterparty and Slimcoin [91]. Miners are required to deposit their coins to the "eater address" for the Proof of Burn procedure to operate effectively. All network users can publicly authenticate this address; yet, it remains private and unattainable to others. Burn resolves the absence of private keys and is generated indiscriminately. In other terms, they are "black hole" addresses, signifying that funds transmitted to them are permanently irretrievable. The quantity of coins in the system is always diminishing, hence enhancing the value of the assets [25]. In figure 4 the function of PoB algorithm is depicted.

.

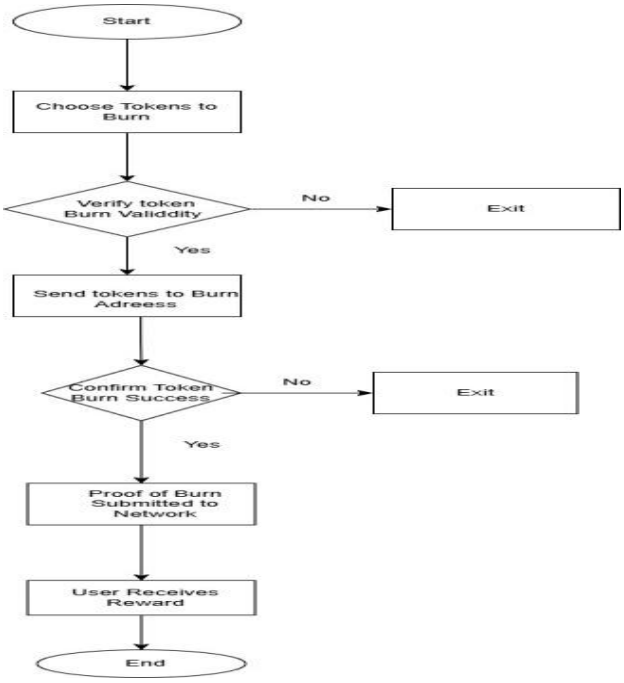

Figure 4: Proof of Burn

3.1.3 Proof of Activity (PoA)

In 2012, Charlie introduced the PoA, as documented by Sarkar [74]. The term "activity" refers to the reward granted exclusively to active stakeholder nodes that maintain a fully operational online node. Proof of Authority (PoA) is a consensus process intended to establish a decentralized cryptocurrency. In this protocol, entities executing computational tasks acquire decision-making authority through Proof of Work (PoW), while entities with a stake earn decision-making authority via Proof of Stake (PoS). The miner's objective is to produce a block header that is largely empty, having minimal header information. Upon selecting this block, its header is disseminated across the network, initiating the subsequent phase of signing and validation. The likelihood of validating or signing the new block is contingent upon the proportion of stake possessed, as determined by the Proof of Stake (PoS) process [8]. To successfully breach this system, the attacker must dominate a majority of both the network's mining power (exceeding 51%) and the staked coins (surpassing 51%). The transaction fees are allocated to the miners and the privileged stakeholders [84]. A limitation of Proof of Authority (PoA) is that the execution of Proof of Work (PoW) requires a specific level of computational capacity. The approach is resilient to Denial of Service attacks. Decreed is an independent digital currency that operates on a Proof of Authority (PoA) consensus mechanism. Decreed utilizes a hybrid consensus process that integrates Proof-of-Work (PoW) and Proof-of-Stake (PoS), allowing all players to impact the currency's direction. Thus, it tackles the issue of heightened centralization apparent in Bitcoin and other cryptocurrencies. The objective is to diminish the occurrence of hard forks [73].

Proof of Authority (PoA) alters the Proof of Stake (PoS) framework by seamlessly including aspects of Proof of Work (PoW) alongside Proof of Stake (PoS). For a miner to perform specific transactions within a block to mine a new block, the majority of nodes must approve the block for validation once the mined block has been documented in the database. In figure 5 the function of PoA algorithm is depicted.

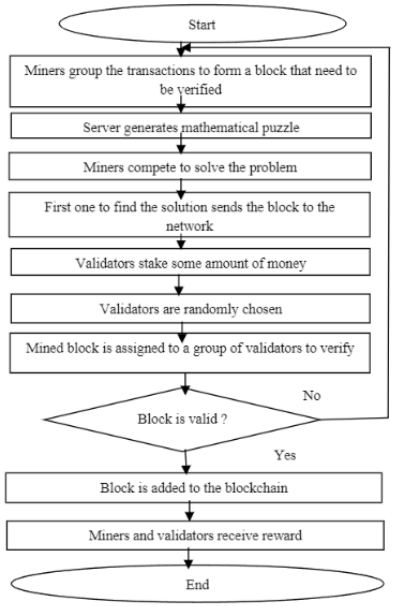

Figure 5:Proof of Activity

3.1.4 Proof of Authority (PoA)

In 2017, Gavin Wood, a co-founder of Ethereum, presented the concept of Proof of Authority (PoAu). This approach correlates with a permissioned network where participant identity is known, and such participants are trusted. Authorities are reliable nodes having unique identification and more so, credible allowing client transactions after consensus is reached, such are the validators. Unlike Proof of Stake (PoS) systems where the participant places certain amount of capital, in this case the block validators risk their reputation .PoAu has advantages such as better efficiency, faster process of transaction occurrence, and better agility. This consensus shows some of its limitations such as less decentralization and the need of the reviewers to have an identity to the public and not be anonymous since the world is digitalized [55]. This technique is practiced in many like in the ethereum consortium proof-of-authority on MS azure, Kovan testnet of ethereum and vechain thor blockchain [21]. Collaborations have been formed with the Kovan testnet for applications in Digital Asset Management (MelonPort), Financial Instrument Insurance (Nivaura), Asset Tokenization Platform (Digix), and Decentralized Energy Data Application Platform (GridSingularity) [22],[23]. In figure 6 the function of PoA algorithm is depicted.

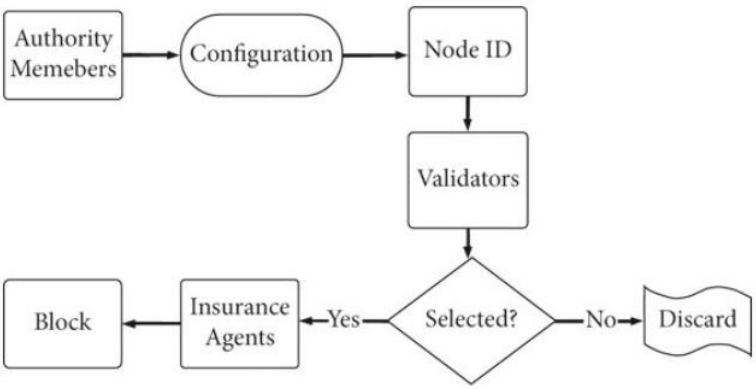

Figure 6:Proof of Authority

3.1.5 Proof of Importance (PoI)

Both Proof of Work (PoW) and Proof of Stake (PoS) are undermined by the fact that affluent users possess a heightened probability of being selected to validate the subsequent block, hence enhancing their chances of future selection as validators. The Point of Interest (PoI) allocates a trust rating to people, determining their privileges. It evaluates the quantity of transactions conducted with others and the beneficiaries of those transactions. The NEM currency employs a framework designed to function as a universal medium for quotidian transactions Timestamps in NEM are associated with transactions and blocks. The generation of new blocks is termed harvesting, and those who execute this duty are called harvesters. Harvesting is exclusively accessible to accounts with a balance over 10,000 XEM, and all such accounts must have a significance score larger than zero. XEM is the official currency employed by the NEM platform. The NEM Infrastructure Server (NIS) nodes provide the backbone of the NEM network, executing transactions alone with the authorization of an account holder. Ivanov. An algorithm ranks miners based on the volume of transactions they process in the pertinent cryptocurrency. The probability of an entity obtaining mining projects escalates with the frequency of transactions associated with its bitcoin wallet .This differentiates proof of importance from the proof of work technique established by Bitcoin. Proof of work algorithms evaluate entities based on the volume of coins they extract. This signifies that an entity's processing power directly correlates with the quantity of coins it can mine within a proof of work cryptocurrency network. The proof of labor mechanism does not facilitate the circulation of the cash. A solely proof-of-work system is prone to centralization, as priority is essentially awarded to the entity that can provide the highest computational power. The evidence of importance system differs from the proof of stake system. The proof of stake system ranks players based on the amount of the pertinent coin they have. A proof of stake algorithm autonomously assigns mining duties to miners based on the amount of the pertinent cryptocurrency held in their wallets. This provides limited incentive to employ bitcoins for transactions [98]. Proof of importance systems are designed to incentivize individuals involved in bitcoin transactions by prioritizing miners based on the significance and volume of transactions originating from their wallets. A proof of importance system may incorporate other criteria, such as the wallets involved in transactions. The amalgamation of evidence of importance, proof of stake, and proof of work systems is achievable. An algorithm may evaluate both wallet transactions and the amount of cryptocurrency held when prioritizing mining [46]. These components collectively assist in assessing a node's relative importance, potentially influencing its role or advantages within the network. This technique transcends mere wealth, as evidenced in Proof of Stake, inside a PoI-based blockchain framework by incorporating transactional activity

as a vital criterion of a participant's importance [57]. In figure 7 the function of PoW algorithm is depicted.

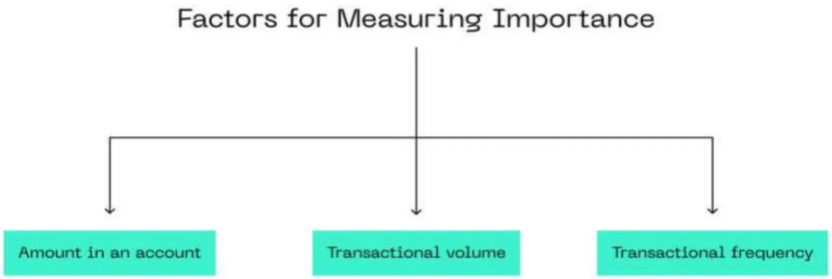

Figure 7:Proof of Importance

3.2 Vote based Consesnsus Protocols

Another class of algorithms seen in blockchain, and distributed ledger technology (DLT) is vote-based consensus procedures. Instead of proof-derived systems like Proof of Work or Proof of Stake where consensus is reached in accordance with the prescribed rules, voting procedures are used in the protocols. Vote-based methods do not resort to having nodes resolve difficult puzzles or lock up assets. Rather, they depend on communication between nodes in order to vote about the validity of a transaction or block [56].

3.2.1 Proof of Stake (PoS)

Proof of stake, also known as PoS, is a consensus protocol or mechanism employed in blockchain platforms which leverages on the stake of crypto currency to choose validators thereby making it less power-consuming compared to the proof of work PoW. Whereas PoW modes necessitate intense computation activities, which are energy high costs based on bicontainers, PoS is intended to mitigate the above outlined excessive energy levels consumed comparatively when it comes to blockchain validation activities. The first attempt to implement PoS was made together with the introduction of cryptocurrency Peercoin in 2012, still, its architecture was somewhat similar to PoW

Due to consensus, once certain transactions are completed, they are considered verified as they are in line with the already existing transactions in the chain. For the PoS blockchains, it is the validators or the minters who carryout this activity while the PoW blockchains have it done by the miners. Validators in PoS systems are typically rewarded for their role in maintaining the network’s security and integrity. To prevent malicious actors from taking control of the network, PoS systems require validators to hold a significant amount of the blockchain’s tokens. As far as it may benefit them, it is difficult such as obtaining sufficient tokens to try to attack the system. On the other hand, PoW operates with computation power and therefore the attackers need to have the majority of the network computing resources, which again is more costly in terms of energy.

One of the things that puts PoS improvement over PoW is the energy cost [7]. However, the first implementations of PoS were easy targets for particular classes of attacks that were responsible for emergence of two major design patterns for PoS: Byzantine Fault Tolerance based ones and chain based ones.

Bashir identifies three other PoS variations (Shifferaw & Lemma, 2021):

- Committee-based Proof of Stake, also called Nominated Proof of Stake (NPoS)

- Delegated Proof of Stake (DpoS)

- Liquid Proof of Stake (LpoS)

In the PoW paradigm, the miners race to solve difficult cryptographic puzzles for transaction validation. The first solver of the puzzle is rewarded monetarily with funds. However, this method of PoW is focus towards excessive computational work wherein almost all energy is expended in solving the cryptographic puzzle challenge. In order to improve the probability of them being the next companies to mine the next bitcoin block, the miners also cooperate within technology groups called mining pools and share the earnings. Although it is secure, PoW also has its limitations, as it is energy inefficient and leans towards centralization since the larger pools take control of the network as its dominant pool [104]. In figure 8 the function of Pos algorithm is depicted.

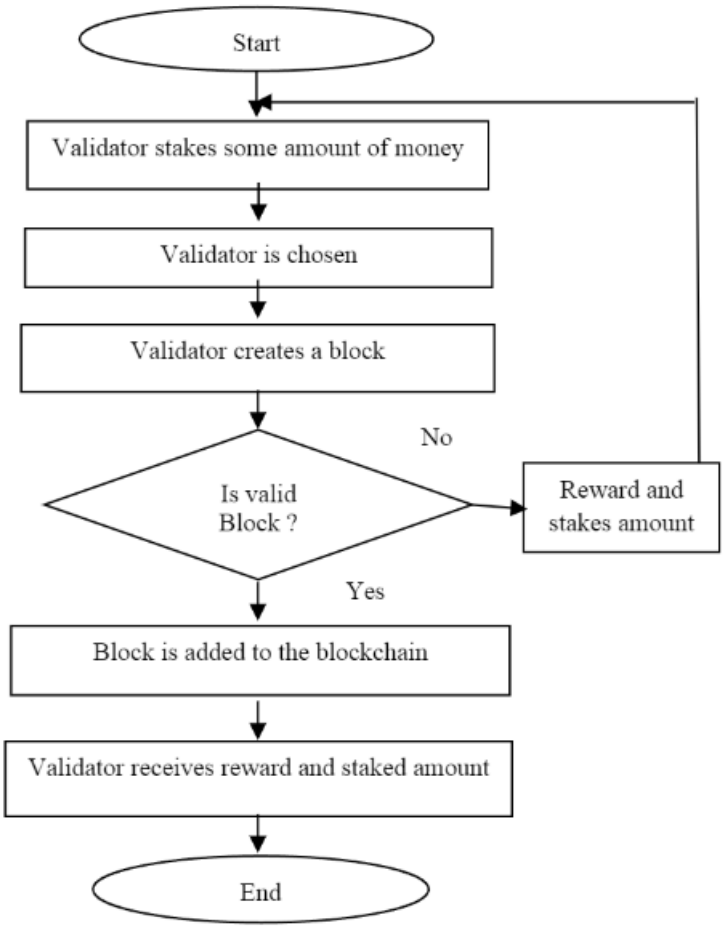

Figure 8:Proof of Stake

3.2.2 Delegated Proof of Stake (DPos)

The Dpos was conceived for the first time in BitShares by Larimer [93]. In the DPoS system, it is the coinholders who use their votes and elect delegates or witnesses/block producers. These delegates are responsible for the endorsement of new blocks as well as for retrieving rewards. Shareholders elect some particular number of the delegates. Each investor gets a proportional amount of voting rights to the number of coins they possess. Each delegate is given time period of a particular slot.. In the event of block validation failure or other malicious activities, the delegate will be substituted with an alternative delegate. DPoS systems may require delegates to hold a designated amount of coins as a stake to exhibit their commitment to the network. Fan and Chai [27] assert that DPOS surpasses PoW regarding performance and energy efficiency.The stakeholders designate a particular number of witnesses to generate blocks. Upon the consensus of a majority of voting stakeholders regarding sufficient decentralization, the selected principal witnesses are nominated. The witness is compensated for producing each block, with the reimbursement rate established by stakeholders via elected representatives. Any failure to provide a block is recorded and may result in eventual removal from the witness list. Delegated Proof of Stake (DPoS) is utilized in Bitshares, Lisk, and Steem [44]. In the Delegated

Proof of Stake (DPoS) consensus mechanism, token holders elect a limited number of delegates to validate transactions and append new blocks. Individuals designated as witnesses are responsible for ensuring the efficacy and security of the network. Delegated Proof of Stake (DPoS) fosters a democratic governance framework by enabling the community to select its representatives, hence aiming to enhance scalability and reduce centralization [57]. In figure 9 the function of DPoS algorithm is depicted.

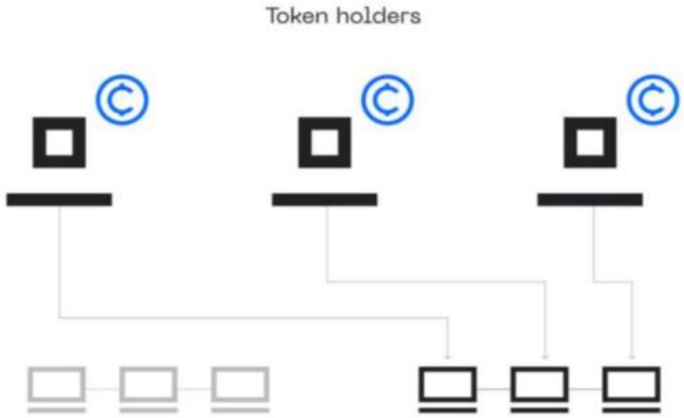

Figure 9:Delegated Proof of Stake

3.2.3 Practical Byzantine Fault Tolerance

Practical Byzantine Fault Tolerance (PBFT) is a consensus algorithm developed in the late 1990s by Barbara Liskov and Miguel Castro, designed specifically for asynchronous systems, where there is no set timeframe for receiving responses. PBFT enhances both time and resource efficiency, overcoming limitations in earlier Byzantine Fault Tolerance (BFT) algorithms. It has been widely applied in distributed computing and blockchain technology [99].

In PBFT, if a message is not received within a certain timeframe, a default vote can be assigned, labeling the corresponding node as "faulty." Additionally, if the majority of nodes produce the correct result, a default response can be chosen. Leslie Lamport's research demonstrated that consensus can be achieved with 3m+1 processors, even if m processors fail, meaning that nearly two-thirds of the system must remain reliable [30].

PBFT addresses two types of failures: fail-stop, where a node completely halts, and arbitrary-node failures, which are more complex. Examples of arbitrary failures include:

- Failure to produce results,

- Producing incorrect results,

- Deliberately misleading outputs,

- Giving inconsistent responses to different parts of the system (Chen et al., 2022).

Unlike Proof of Work (PoW), PBFT doesn't require intensive mathematical computations to reach consensus. For instance, Zilliqa integrates PBFT with PoW for every 100th block. In contrast to Bitcoin's PoW mechanism, where multiple confirmations are needed and transactions can take between 10 and 60 minutes depending on the number of validators, PBFT allows faster

consensus without such delays. All nodes in a PBFT network participate in processing client requests, which reduces volatility in rewards and incentivizes nodes to actively contribute to decision-making [109].

The PBFT process involves three main steps:

1) The client submits a request,

2) The system undergoes a three-phase consensus procedure,

3) The client receives and verifies the response, confirming that consensus was achieved [104]

In figure 10 the function of PBFT algorithm is depicted.

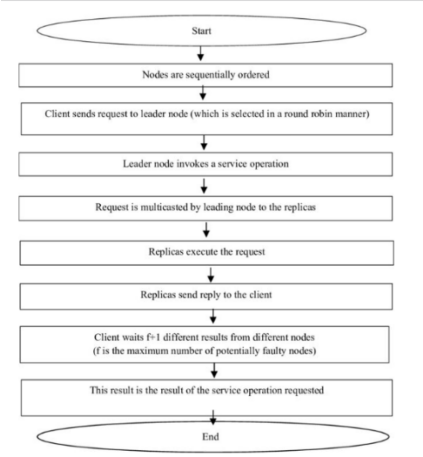

Figure 10:Practical Byzantine Fault Tolerance

3.2.4 Ripple

Ripple is a prominent and established blockchain network, with its XRP token ranked fourth in market capitalization as of October 2020. The principal objective of the Ripple network is to enable swift international transactions, asset exchanges, and financial settlements. The distributed consensus method is implemented using a peer-to-peer network of validator nodes. These nodes are tasked with preserving a comprehensive ledger of all transactions occurring on the network [77]. Unlike the consensus process utilized by Nakamoto in Bitcoin or Ethereum, the Ripple consensus protocol eliminates "mining" and implements a voting mechanism that takes into account the identities of its validator nodes to attain consensus. Ripple exceeds Bitcoin in transaction processing efficiency, capable of managing up to 1500 transactions per second. Furthermore, it attains exceptionally rapid transaction settlement speeds, often between 4 to 5 seconds. Nonetheless, Ripple's consensus protocol diverges from traditional concepts and methodologies related to Byzantine agreement or Byzantine fault tolerance (BFT) consensus [85]. These systems commence with a collective network of nodes that collaborate to attain consensus,

and the relevant protocols have undergone extensive analysis for many years. The Ripple consensus protocol, unlike previous systems, integrates the notion of subjective validators.

This strategy involves each node selecting specific validators it trusts and just engaging with them to reach consensus on transactions. The architects of Ripple aimed to broaden the membership of validator nodes about BFT consensus via this method. The reliable validators of a node are established by a Unique Node List (UNL), which is essential for formalizing the protocol. Each node retains a designated UNL in its configuration file and considers only the perspectives of nodes within its UNL to achieve consensus. Ripple utilizes a consensus mechanism based on its proprietary protocols, thereby circumventing transaction validation using Proof-of-Work (PoW) or Proof-of-Stake (PoS) methods. This algorithm functions on a permissioned blockchain, unlike Bitcoin and Ethereum. All nodes periodically employ the RPCA mechanism to verify the accuracy of transactions. The Ripple consensus technique is validated by a node [95]. In figure 11 the function of Ripple algorithm is depicted.

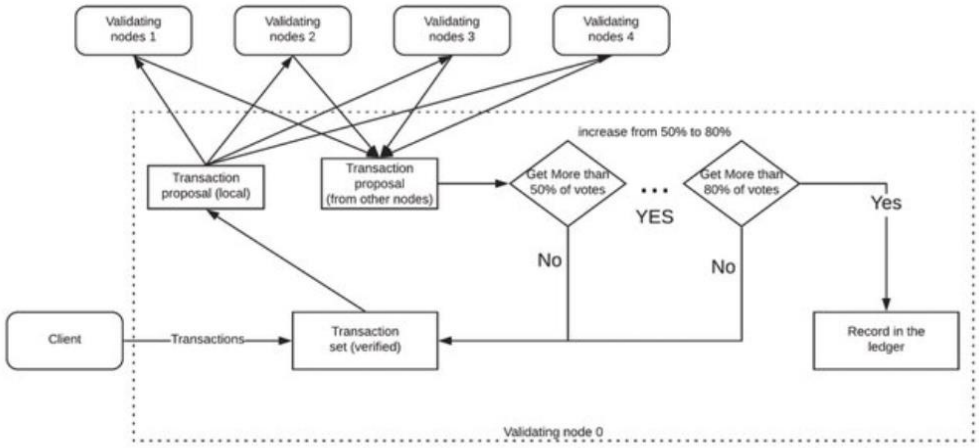

Figure 11:Ripple

3.2.5 Raft

Paxos is a foundational consensus algorithm known for providing crash fault tolerance. However, its complexity in explanation and implementation has led to many efforts to simplify its principles [58]. Raft was developed as an alternative, designed to be easier to understand while offering the same level of efficiency as Paxos. It breaks down the consensus process into three main components: leader election, log replication, and safety. This modular structure improves clarity and simplifies Raft's implementation, making it widely used in blockchain systems and distributed storage [62].

In the Raft algorithm, the leader is the central node responsible for managing data distribution within the system. When the leader goes offline due to network or performance issues, a new election is triggered, and the system votes to choose a new leader. However, this structure

gives more power to the leader than to non-leader nodes, creating an unequal dynamic within the network [103].

Raft was specifically designed to address these challenges and, according to its creators, is more understandable than Multi-Paxos while maintaining the same efficiency. Its simplicity makes it easier to replicate and compare to Multi-Paxos in practical [65].

Raft shares some underlying assumptions with Multi-Paxos. According to Ongaro and Ousterhout , these assumptions include:

1. The system is asynchronous, meaning there is no fixed limit on message delays or processing times, and global clock synchronization is impossible.

2. Network communication can be unreliable, with possible delays, packet loss, duplication, or reordering.

3. Byzantine failures (malicious nodes) are not accounted for.

4. Clients must interact with the current leader and are responsible for identifying the leader.

Additional conditions include:

1. The protocol has access to monotonically increasing values.

2. All nodes begin in the same state and respond deterministically to client actions.

3. Nodes have an immutable, infinite permanent storage system, and write operations must be completed before any system failure occurs.

4. Nodes are aware of all other nodes in the cluster, and membership cannot change dynamically without additional engineering, although modifications like log compaction and dynamic membership changes are possible.

Raft organizes nodes into three possible states: leader, follower, or candidate. Time is divided into terms, with one node acting as leader during each term. The leader collects transaction requests, verifies them, and orders them chronologically. These transactions are then packaged into blocks and sent to follower nodes for replication and confirmation. If there are no new transactions, the system enters an election timeout phase, and a new leader is elected for the next term.

When a node becomes a candidate, it requests votes from other nodes to secure a majority. If it wins the vote, it assumes leadership and informs the other nodes. If a candidate receives a message from another node claiming to be the leader, the candidate compares the term index of the message with its own. If the term index of the other node is lower, the candidate ignores the message and continues seeking votes. If the term index is higher, the candidate recognizes the other node as the leader. If no candidate secures a majority, a new election is triggered after a random delay, and the node with the shortest delay initiates the next voting round [43].

A key difference between Paxos and Raft is that Raft always elects the most up-to-date node as the leader, while Paxos allows any node to become leader. Raft is also the consensus mechanism used by major blockchain platforms such as R3 Corda and Quorum.

Raft was designed to be simpler than Paxos in such a way that all the nodes come together and choose one of the nodes as the leader and the other nodes act on the orders given by the leader. The leader is also responsible for the replication of the state transitions to the

follower nodes, and the flow of transactions is only from the leader to the followers consolidating the top-down control. Where there are no overlapping roles allowed in Paxos, Raft has clearly defined roles which are leader, candidate and follower nodes.

Terms are distinct intervals employed by the Raft consensus protocol. In every term, there is an election in which some Raft nodes actively solicit the other nodes in an attempt to be elected into a leader role. After a leader has been chosen, he or she runs the system through the end of the term, holding the responsibility for the smooth processing of transactions. The structured transition between these states and the leader's management of transaction flow provide a more organized and transparent system compared to Paxos. Raft is often illustrated in technical diagrams to visually represent these state changes and leadership transitions. In figure 12 the function of Raft algorithm is depicted.

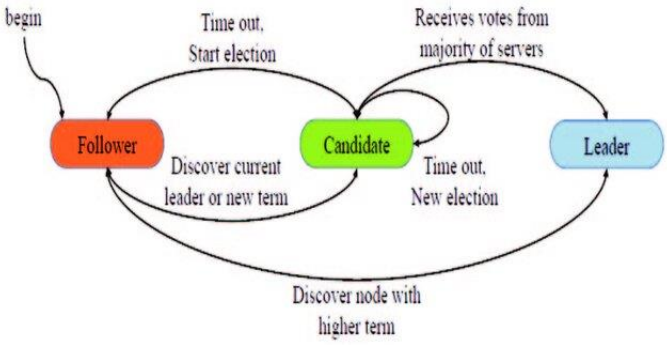

Figure 12:Raft

4. Protocol Comparison

Evaluating permissionless blockchains is very simple, as they pursue a same objective yet vary considerably in terms of investment, throughput, and scalability. A multitude of studies have assessed these blockchains across diverse parameters. However, the examination of permissioned blockchain is a more complicated one. Such systems are usually divided into two kinds depending upon the level of security they offer, and as a consequence the functions they perform vary in terms of performance. In relating studies on this topic, there is lots of variation in terms of terminology, with some works focusing on consensus algorithms in the context of platform performance [68]. In contrast to permissionless systems such as Proof of Work (PoW), which necessitate substantial computational power and energy, permissioned blockchains do not require considerable initial investments, as their protocols do not depend on energy-intensive techniques. Nonetheless, they exhibit significant heterogeneity in scalability and throughput.

Permissioned blockchains exhibit significant variation in the level of mutual trust among nodes. A comparative analysis of these systems was performed based on qualities including security, trust, throughput, and scalability, utilizing established research in the field[47]. The first two consensus protocols, Paxos and Raft, are designed to protect only against crash failures and ignore the possibility of Byzantine failures, which means that their design assumes that all nodes are assumed to be honest. These systems do not address Byzantine failures, which result from subversion, but rather address purely crash-inducing faults and are thus less secure.

In permissionless blockchains, nodes lack prior acquaintance, rendering trust a significant difficulty. In permissioned systems, nodes need not be previously familiar, yet their involvement in the network confers a fundamental level of confidence. In Practical Byzantine Fault Tolerance (pBFT), nodes commence communication with each other grounded in reciprocal trust within the system. Conversely, Delegated Byzantine Fault Tolerance (dBFT) and Federated Byzantine Agreement (FBA) enable nodes to assess trustworthiness through personal discretion. This is particularly apparent in FBA, where flexible trust is prioritized, and players depend on the judgments of a limited number of trusted entities instead of the entire network.

Proof of Authority (PoA) engenders trust by pre-verified identities and the responsibilities associated with them. Paxos and Raft, conversely, presuppose that nodes trust one another's integrity, though not their responsiveness. A primary disadvantage of permissionless blockchains is their inability to match the transaction throughput of conventional payment systems, mostly due to the design constraints inherent in functioning within a trustless environment. For a blockchain protocol to properly support financial or business systems, it must efficiently manage enormous transaction volumes.

This research reveals that, although the methods demonstrate good throughput, significant variances exist across them. For example, pBFT attained a throughput of approximately 200 transactions per second (TPS) during a test including 10,000 transactions on the Hyperledger Fabric platform [62]. dBFT is purported to provide up to 4,000 TPS; however, data from the NEO blockchain indicates a lower actual throughput of approximately 1,000 TPS. FBA demonstrates promise for large throughput, with practical applications like as Ripple and Stellar managing approximately 4,000 TPS, while theoretical projections indicate this might surpass 10,000 TPS. Proof of Authority (PoA) demonstrates significantly reduced throughput, approximately 80 transactions per second (TPS).

The scalability of these protocols varies considerably. Studies indicate that a pBFT network with 40 nodes can authenticate a block of 10,000 transactions in under four seconds. As the number of nodes rises to 200, the validation duration extends to 26 seconds, indicating exponential growth [86]. This indicates that pBFT is more appropriate for smaller, regulated systems but encounters scaling challenges in bigger networks due to the necessity for all nodes to communicate with each other. on the other hand, with dBFT implemented on the NEO blockchain, a block can be finalized in 20 seconds independent of the number of nodes, which can be from 7 to 1024. dBFT is stated to provide a better scalable solution in relation to pBFT, however, it still has some limitations.

FBA has a characteristic that its scalability is well catered for, and this can be shown by the networks Ripple and Stellar which have 130 and 136 nodes respectively, and still function well with such a small number of nodes. These networks are known to have consistent scalability properties; however, further research is needed to support this assumption. Proof of Authority (PoA) provides consistent independent scalability allowing block validation to take from 5 to 8 seconds even as the number of nodes reaches over 1,000.

Still, for both Paxos and Raft, communication though narrowed into proceeding through a single leader node becomes the focal point of distortion. This centralization may slow down the process especially as the network grows. In conclusion, the protocols, although each has its good and bad points, there are some opportunities and constrictions which are universal within the paradigms but differ radically in the implementation of scalability and throughput [47].

Protocol	Security	Throughput (TPS)	Scalability	Energy Efficiency
PBFT	Requires mutual knowledge	200 TPS	Limited by node communication	Low (less computational demand)
Paxos	Full mutual trust	Varies with network load	Limited by leader node bottleneck	Moderate (not energy-intensive)
Raft	Full mutual trust	Varies; efficient	Better than Paxos, but still limited	Moderate
dBFT	Limited, permissioned trust	4,000 TPS	Reasonably scalable, but some constraints	Low (efficient design)
FBA	Flexible trust	4,000 TPS (up to 10,000 TPS theoretical)	Highly scalable	Low
PoA	Established identity	80 TPS	Consistent, scales well	Low
PoET	Permission-based	Moderate	Limited, ideal for small networks	Very low (fair lottery system)
Ripple	Minimal trust required	Varies; efficient	Scalable through sub-networks	Low (focus on efficiency)

Table 3:Protocol Comparison

4.1 Security Perspective

The consensus mechanisms bolster the security, consistency, and integrity of the network. They enable the creation of a uniform agreement within a peer-to-peer network, removing the necessity for a central trusted authority to supervise the process. The Proof-of-Work consensus extensively employs computational resources to maintain the chronological order of transactions throughout the network. The Proof-of-Stake (PoS) consensus mechanism attains agreement inside the blockchain network by depending on the durability of cryptographic signatures and their economic value. Consequently, we may say that the integrity and security of Proof of Work (PoW) are inherently linked to energy consumption.

The integrity and security of Proof of Stake (PoS) essentially rely on the value of economic incentives. Proof of Work (PoW) expends considerable energy, is affected by economies of scale, and is vulnerable to self-serving mining practices. The blockchain consensus method is defined by its speed, efficiency, decentralization, democratic ideals, adaptability, and scalability. Nonetheless, the concealed balances and the secretive nature of the currency may hinder the adoption of DPoS. To address this difficulty, DPoS must attain an equilibrium between transparency and efficiency. A considerable number of privacy coins encounter difficulties in sustaining their transparency levels during declines. The DPoPS algorithm, an advanced iteration of the DPoS framework, is utilized to tackle the privacy coin issue [104]. Public Key (PoB) is an alternative consensus technique that significantly decreases the considerable energy consumption linked to Proof of Work (PoW).

The operational approach depends on the reduction of currency possessed by the mining nodes. The ability of a node to produce new blocks depends on the amount of bitcoin it distributes. The burning activity guarantees the uninterrupted functioning of the network, and the participating

miners receive adequate remuneration for their contributions. The PoB periodically incinerates cash to maintain mining restriction and reduce the likelihood of disproportionate rewards for early adopters. The efficacy of burned coins declines to a degree following the creation of a new block after each occurrence. Miners must invest in state-of-the-art equipment to maintain a competitive edge as technology evolves. The probability of hostile assaults in a P2P network is increasing, resulting in erratic conduct from a compromised peer [11]. PBFT is an enhanced version of the BFT consensus mechanism utilized in asynchronous blockchain systems to increase efficiency and reduce overhead runtime.PBFT ensures that the aggregate of malicious nodes within the system does not exceed the simultaneous engagement of all nodes in the network within a defined vulnerability interval, thereby guaranteeing robust liveness and safety. It ensures transaction finality without requiring confirmations and is far less computationally intensive than Proof of Work (PoW). This method is susceptible to Sybil attacks due to considerable communication overhead requiring node interaction and is designed for traditional small-scale networks [16]. Paxos is a notable consensus algorithm formulated for asynchronous networks that tackle a non-Byzantine problem demanding resolution. The achievement of consensus is complicated by the possible existence of several malevolent nodes in the asynchronous system. A notable difference between Paxos and PBFT is that PBFT experiences a substantial increase in traffic during message delivery. In Paxos, the leader conveys a message to nodes, gathers responses, and subsequently propagates the message throughout the entire network. To compute the TPS, one must sum the input time (N), broadcasting time, and output time, and thereafter divide this amount by the overall duration. If the PBFT consensus method encounters issues with network traffic, then Paxos will similarly confront hurdles under identical network conditions. Paxos allows a maximum node tolerance of 50%, whereas Byzantine accepts only 15%. Ideal synchronization conditions allow Paxos to achieve a fault tolerance of 99.9% [100]. The principal objective of Byzantine Fault Tolerance consensus methods is to alleviate the energy inefficiencies associated with Proof of Work (PoW) mining. PoET is a Protocol-Based Formal Trust (PBFT) consensus mechanism aimed at augmenting Proof of Work (PoW) consensus and offering a unique possibility for permissioned blockchain networks [83]. The PoET consensus process reduces excessive resource consumption and substantial power usage using a fair lottery system. Unlike PoW, the PoET consensus process necessitates minimal energy consumption. This feature enables miners to rest or undertake different activities for a designated period, so enhancing their efficiency. An external node cannot interfere with the functionality of reliable programs in a secure environment. Moreover, it ensures that the conclusions may be validated by other participants, hence improving the clarity of the collective consensus inside the network. PoET utilizes trusted computing to produce unpredictable delays in the process of block production. Despite its deficiencies, the trustworthy computer component is wholly dependable. The current adversarial framework of PoET endangers the blockchain protocol by undermining a restricted number of nodes [20]. Consensus mechanisms are essential for guaranteeing the security and efficacy of a blockchain system. Proof of Work (PoW) and Proof of Stake (PoS) consensus algorithms demonstrate significant efficacy in mitigating internal flaws, enhancing adaptability, and ensuring security within a blockchain.Consequently, Proof of Work (PoW) and Proof of Stake (PoS) are widely acknowledged alternatives for public blockchains. Nonetheless, Proof of Work (PoW) and Proof of Stake (PoS) exhibit a prolonged transaction confirmation rate, potentially constraining their applicability in scenarios requiring swift confirmation. In a consortium/private blockchain system, each node must function in an unequivocally "correct" manner. Consequently, PBFT, Paxos, and RAFT are increasingly appropriate selections for consortium or private blockchain networks. The RAFT consensus mechanism is founded on the multi-Paxos computation yet possesses a distinct organizational structure in contrast to Paxos. This design substantiates its rationale and establishes a more robust basis for developing an effective consensus procedure. The RAFT approach delineates the fundamental elements of consensus, including leader selection, log replication, and security, to improve comprehension. This technique ensures greater coherence, hence decreasing the number of states requiring

inclusion. The RAFT algorithm is a consensus method designed for private blockchains, particularly applicable to ad-hoc networks like intranets. The research conducted by Wu [96],[97] illustrates that RAFT attains safety performance comparable to Paxos.It significantly enhances the ease of application development and understanding. It can withstand malicious nodes constituting up to 50% of the total nodes in the system's failure. Addressing crash fault issues is more intricate than overcoming the Byzantine constraints of private blockchains. Numerous contemporary consensus techniques for the Byzantine Generals Problem exhibit considerable communication latency owing to the synchronous connectivity among all nodes inside the network. The Ripple consensus technique utilizes collectively trusted sub-networks within the broader network to resolve this issue. The necessary trust level for these sub-networks is minimal and can be further reduced through intentional selection of the member nodes. Furthermore, it ensures that just a minimal degree of connectivity is required to maintain consensus throughout the whole network. This method ensures a low-latency consensus mechanism that is robust against Byzantine defects. The Ripple consensus process in blockchain enables the integration of banks, payment providers, and digital asset transactions, thereby improving the efficiency and cost-effectiveness of global payments. A Unique Node List (UNL) is utilized on the server. The server would solicit information regarding the node within the UNL system to ascertain whether to document a transaction in the ledger. Upon reaching an 80% consensus rate, the transaction is recorded on the ledger. If the proportion of defective nodes in the UNL is below 20%, the ledger for a node in the system remains precise [19].

Consensus Mechanism	**Security Perspective**
Proof of Work	Elevated security yet energy-inefficient. Susceptible to extensive assaults and centralization.
Proof of Stake	Robust security, although necessitates a significant investment. Susceptible to assaults by significant token holders and centralization.
Delegated Pos	The centralized character may compromise decentralization. Susceptible to delegate collusion and centralization.
Proof of Burn	Decreases energy consumption; yet economic manipulation and token scarcity may result in inequitable power distribution.
Practical Byzantine Fault Tolerance	Effective for small-scale networks; nevertheless, scalability and vulnerability to attacks render it inappropriate for extensive public blockchains.
Paxos	Optimal for compact, authorized networks. Issues of scalability and performance in extensive systems.
Raft	Easy to build with robust safety features, although prone to failures in high-traffic or partitioned settings.
Ripple	Rapid transactions, however potential concerns of centralization and manipulation by

	trustworthy validators. Optimal for authorized systems.
Proof of Elapsed Time	While energy-efficient, dependence on trusted hardware presents hazards if the hardware is compromised.

Table 4:Consensus Mechanism Security Perspective

4.1.1 Vulnerabilities and Real-World Examples

While each consensus mechanism improves blockchain performance, it is subject to unique types of attacks which may severely undermine this aspect. These risks have been made clear through several historical cases which have emphasized the hazards posed by diversity of blockchain protocol.

- Proof of Work (PoW):

Vulnerabilities: PoW systems suffer from 51 percent attacks, in which the attacker command over fifty percent of the total hash rate on the network. This would allow the assailant to perform doubly spend coins. PoW, in addition to self-serving mining, enables miners to make the most profit at the disadvantage of the network;s stability. PoW also increasing centralization, as operational expenditures can only be supported by major mining enterprises.

Real-World Example: In 2018, for instance, PoW systems defended over the risk of being exploited through a 51 percent attack during which the attacker was able to control iron gold over 50 per cent the total hash rate and increase its chances of earning money. As a result of this, dual spending assaults were carried out. The attack on Bitcoin Gold unquestionably underscored the dangers of BPOW systems with their tendency toward domination by consolidated pools; larger piles of iron control so much of the consensus.

- Proof of Stake (PoS) :

Exposure: PoS systems are susceptible to long-range attacks where an attacker splices a clone chain to a block that is very far back the history than the present. An attacker is able to exploit Nothing-at-Stake attacks, where the validators are able to validate multiple competing blockchains due to a lack of disincentive which may lead to the network experiencing fragmentation. Centralization is also the vast problem, since the big token holders now control some good fraction of the validation power of the network thus defeating the very essence of distribution.

Real-World Example: Ethereum Classic faced perhaps the most convincing 51% attack in 2019 where an attacker on a subnet staking over 51% was able to reorganize the whole blockchain and double-spent some tokens. This demonstrates the perils of centralization in PoS where just a very few people are able to control the network practically and perform the degrading tasks.

- Delegated Proof Of Stake (DPoS):

Vulnerabilities: Problems of the strength of centralization where the validation is carried out by a few delegates arise in DPOS systems. Also, the disruption of the normal operation of the system by delegates by way of conspiracy is a problem. Governance manipulation vulnerability is another potential weakness as it pertains to the selection of delegations parroted by voters who have a significant amount of stake.

Real World Example: In 2018 several accounts, frozen by block producers in EOS, received attention as a governance and centralization-related attack on the system. DPoS is especially

susceptible for this kind of attack as delegate collusion occurs where few delegates can make decisions that benefits themselves but these go against the supposed ideals of decentralization.

- Proof of Burn (PoB):

Vulnerabilities: There are always issues with economic disruption in PoB systems wherein big holders could control the rate of burning and manipulate the value of tokens leading to dictatorial powers. There are also possibilities of inflation risks to emerge if the volume of coins burnt in total is not appropriately controlled leading to an overbearing quantity of tokens in circulation. As with many new launches it is possible that a form of token manipulation can take place because it is the early adopters that will likely benefit the most for burning a larger percentage of the tokens issued.

Real-World Example: Counterparty, a Bitcoin sidechain that uses PoB, lowers the demand for energy but it has the disadvantage that significant holders emit more tokens and hence influence the market price of tokens which leads to possibilities of the poob system being abused. This type of manipulation may cause similar problems in unbalanced power distributions and highly concentrated power structures.

- Practical Byzantine Fault Tolerance (PBFT):

Vulnerabilities: PBFT suffers from Sybil attack whereby a malicious party may create numerous fake nodes to disrupt agreement on consensus. The PBFT protocol also incurs a great communication cost which makes its use in extensive networks uneconomical. Scalability poses a constraint because as the number of participating nodes increases, the usefulness of the protocol decreases, particularly in public blockchains.

Real-World Example: Hyperledger Fabric, which in some instances incorporates PBFT into its implementation, comes across some performance bottlenecks when deployed in bigger networks due to the increasing communication overhead. While it is very secure for small networks configurations, its poor performance and not being scalable has refurbished its use to bigger and public blockchain systems.

- Paxos:

Vulnerabilities: Paxos is not immune to the risk of network partitioning, which refers to a scenario when the network gets divided into two or more segments that cannot communicate with each other, thereby losing the consensus. There is also the problem of message flooding since the protocol depends on a large amount of messages to preserve consensus and this, in turn, results into time lag and degradation of performance. Latency-related problems tend to happen, particularly in larger or more intricate networks.

Real-World Example: Paxos has been implemented within numerous permissioned blockchains, but its limits were reached within large scale deployments due to latency and communication bottlenecks. In the case of Google Spanner which uses Paxos, problems of scale and message delivery were seen when moving across networks characterized by high latency.

- RAFT:

Vulnerabilities: One of the weaknesses of RAFT is that it does not work properly if communications messages are delayed or when the leader fails. In the case of failure of the leader

node, there are delays in reaching consensus as the protocol calls for the election of a new leader. Another problem is that of network partitioning, which can lead to a situation where the leader is partitioned from other nodes, causing a delay in reaching consensus and leading to inconsistent states. RAFT may also experience issues in maintaining consensus in high traffic environments.

Real-World Example: Consul, a system that employs the RAFT protocol for reaching consensus, has also experienced delays in leadership election and has reported the existence of network partitioning issues. These vulnerabilities have affected its consistency and availability levels especially when it comes to adoption in large scale systems.

- Ripple:

Vulnerabilities: Ripple's approach to consensus has some centralization vulnerability because it has a limited number of credible validators. If those whom the network relies upon are either compromised or colluding with each other, they may process altering transactions and therefore, violate the consensus. Moreover, any validator may be replaced by an attacker if the attacker is able to compromise a sufficient number of validators. This concentration of trust induces also the risk of the concentration of the validation process, when companies and organizations become possessors of sufficient resources and thereby no matter how many validators validates the block, certain companies will always hold the larger portion of validators.

Real World Example: Ripple has been accused of no spread out decentralization since a small number of entities hold majority of the trusted delegates. There has been substantial contention on the degree to which large stake holders or corporations would be able to dominate the consensus in the same way that they do with majority of the power over the transaction verification process.

- Proof of Elapsed Time (PoET):

Vulnerabilities: PoET is liable to assault as with the PoET challenge, blockmakers are reliant on intel's Software Guard Extensions (SGX), which works for trusted execution. If an attacker succeeds in either avoiding or compromising the SGX by any means, he/she is most likely to commandeer the block's making. Other hosts of the threat also lie in the interference from the outside, trusted computing environments can be easily compromised if they are not designed securely. The dependence on trusted parts of hardware lowers the level of trust in the system as a whole.

Real-World Example: Hyperledger Sawtooth or better still a sawtooth in hyperledger, where the PoET notion functions, depends on Intel Systems to generate the secure environment inside the trusted enclaves using Intel SGX. By 2018, there had been reports of loopholes in Intel SGX, posing potential threats to the security of the PoET system. So the recent notices bear evidence of the weaknesses that underscored the security and integrity of the hardware because such hardware would be susceptive to attacks that would defeat the entire consensus mechanism.

4.2. Application Perspective

The Proof of Work (PoW) consensus mechanism is extensively used in various cryptocurrencies and blockchain systems due to its robust security features. Several notable examples include:

1. Litecoin: Created as an improvement on Bitcoin, Litecoin enables fast, nearly cost-free global transactions. Like Bitcoin, it uses the Proof of Work algorithm for mining but employs a memory-intensive method called scrypt instead of a computation-intensive one. This design aims to democratize mining by allowing more participants to mine using regular CPUs rather than the high computational power needed for Bitcoin. Additionally, Litecoin significantly reduces

transaction confirmation times from Bitcoin's 10 minutes to about 2.5 minutes, allowing for increased transaction capacity. Litecoin has a capped supply, with a maximum of 84 million coins available for circulation [72].

2.Ethereum: While Ethereum's mining method is similar to Bitcoin's, it uses a distinct PoW algorithm called Ethash, specifically tailored for the Ethereum network. Ethash was developed to combat the issue of mining centralization caused by the reliance on specialized hardware, particularly ASICs. By optimizing for commodity hardware, Ethash reduces the benefits of using ASICs over standard hardware, helping maintain decentralization and strengthen the network against potential attacks. Ethash's memory-hard feature makes system performance dependent more on memory capacity than processing speed, making GPUs ideal for mining [18],[37].

In addition to Litecoin and Ethereum, cryptocurrencies like Bitcoin Cash, Zcash, and Bitcoin SV also use PoW consensus. However, PoW-based networks are vulnerable to a 51% attack, where a group gains control over the majority of the network's hash rate. This control allows them to validate fraudulent transactions, halt new ones, and monopolize block mining. A successful 51% attack can lead to double-spending, where previously confirmed transactions are reversed while attackers dominate the network [76].

Cryptocurrency	Pow Algorithm	Use Case	Advantages
Bitcoin	SHA-256	Store of value	Highly secure and decentralized
Litecoin	Scrypt	Quick transactions	Lower fees, faster block generation
Ethereum	Ethash	Smart contracts	Supports decentralized apps
Bitcoin Cash	SHA-256	Peer-to-Peer transactions	Faster transaction times than Bitcoin
Zcash	Equihash	Privacy-focused transactions	Strong privacy features, shielded transactions
Bitcoin SV	SHA-256	Large scale applications	Focus on scalability, larger block sizes

Table 5:Use of PoW Alforithm

5. Future Research

Research must focus on the improvement of consensus algorithms that facilitate higher transactions per second (TPS), while still being secure and decentralized. Layer 2 solutions, sharding, and hybrid consensus models that combine several protocols' strengths may be ways to this (Wang et al., 2020). It's unreasonable to not take steps to secure blockchain networks as the levels of advanced cyber threats are on the rise. Future studies should also work toward increasing the threat diversity and purposeful threats such as the 51% attack, Sybil attack, and any other malicious functions aimed at the consensus systems. This could include designing stronger Byzantine Fault Tolerant (BFT) algorithms or exploring the possibility of quantum-based consensus mechanisms before quantum computers are available [100].

The key drawback of the current blockchain systems is the risk seen in the principles of cryptography due to quantum computing. The immediate objective of future research should be the search for the post-quantum blockchain ecosystems security sustaining consensus

algorithms. At the same time, hybrid solutions that combine classical and quantum systems may be applicable for this purpose during the transition period.

The rapid rise in the adoption of blockchain systems calls for the need to ensure that there is interoperability between various blockchains. The study of cross-chain consensus mechanisms should help ensure bi-directional bridging between a large number of blockchain networks, to improve the transfer of value, information, and smart contracts between the heterogeneous infrastructures. Mechanisms for cross-chain consensus implementation such as sidechains, atomic exchanges or bridge protocols may enable inter-chains to work hand in hand easily. This will contribute towards the establishment of a more coherent and flexible blockchain system [102].

Considerations on the development of consensus algorithms include, and perhaps most importantly, how to ensure a reasonable trade-off between decentralization and efficiency. Further studies ought to focus on either the invention of novel or modification of existing consensus protocols to centralize the oppression of authority or avoid the tendencies towards central tyranny any one person or body having control that may be against the core ideology of blockchain as a technology. It is anticipated that the comparative study of alternative forms of decentralized governance mechanisms, especially in PoS and DPoS systems, will further improve equity and inclusiveness in decision making processes [102] From an ethical standpoint, DPoS has been critiqued for concentrating power on only a few stakeholders and undermining decentralization as a core principle of blockchain technology. Voting procedures and procedures that would deter collusion are critical for the reliability of the system.

The increasing adoption of blockchain technology for purposes other than cryptocurrency such as supply chain management, healthcare and even the Internet of Things (IoT) demands the adaptation of special consensus methods to suit these specific needs. On the utilization of consensus methods, possibilities for further research include light weight consensus algorithms for IoT devices and protocol compliance methods in health care and financial services. For example, in medicine, the implementation of blockchain technologies for the storage of confidential information about patients has implications for data security, especially in terms of compliance with data protection laws, including GDPR or HIPAA. It has been previously established that future consensus approaches would require strong and effective security controls and measures that are compliant to legal requirements in order to avert data misuse or confidentiality breach. While in finance, consensus protocols must integrate AML and KYC requirements but furthering than that must not compromise the decentralization.

Lastly, the conflict between the deployment of consensus algorithms and the environment, the most embarrassing of which is the post-PoW model consensus algorithms due to energy requirements, remains a serious challenge. However, the changeover to more energy-efficient solutions and/or the creation of mining protocols based on the use of renewable energy sources may integrate the expansion of blockchain development and technology with the development goals established in the broader context.

In summary, it can be concluded that the future development of consensus mechanisms will have to take into consideration modern technological progress as well as the existing regulatory frameworks, ethical perspectives, and the requirements of sustainable development. These concerns enable the horizontal expansion of blockchain technology that does not contradict its basic principles: decentralization, transparency, and security.

6 Conclusion

The emergence of the Internet of Things (IoT) and big data has rendered a substantial amount of critical and sensitive information available online. Research studies on internet security

and privacy are conducted in response to the public's waning trust in online information. In the imminent future, blockchain technology possesses the potential to profoundly revolutionize data storage, sharing, and accessibility across societal, organizational, and industrial domains. Through the utilization of cryptographic computations and hashing, it exhibits remarkable attributes in safeguarding information privacy, transparency, and security, exceeding those of other nascent technologies. To preserve the decentralized and autonomous nature of the blockchain network, it is essential to implement an automated method that ensures unanimous consensus among all participating nodes solely on legitimate transactions. The consensus technique improves the security, coherence, and integrity of the network, enabling the establishment of standardized agreements inside a decentralized peer-to-peer network that functions autonomously, devoid of a central, trusted authority for decision-making duties. Consensus methodologies have received the most focus and effort among all elements. The consensus algorithm is the essential element of the blockchain. The accuracy, effectiveness, and efficiency of the consensus mechanism are closely linked to the security and resilience of any blockchain system. Consensus mechanisms constitute the essential framework of blockchain technology, enabling the secure and efficient functioning of decentralized networks in the absence of a central authority. These protocols ensure consensus among all participants in a blockchain network concerning the present state of the distributed ledger, which is crucial for maintaining the integrity and stability of the system. Various consensus mechanisms have arisen throughout blockchain evolution, each with unique advantages and disadvantages. Proof of Work (PoW) has functioned as a foundational protocol, providing robust security but resulting in considerable energy consumption and scalability challenges. Proof of Stake (PoS) and its variations, particularly Delegated Proof of Stake (DPoS), offer more energy-efficient alternatives; but, they introduce complexities, including the risk of centralization issues. Practical Byzantine Fault Tolerance (PBFT) and Proof of Authority (PoA) are divergent methodologies tailored to address specific network needs, particularly in permissioned blockchains, where the primary aim is to optimize efficiency and guarantee transaction finality.

Bibliography

1. Abraham, I., Gueta, G., Malkhi, D., Alvisi, L., Kotla, R. and Martin, J.-P. (2017). Revisiting Fast Practical Byzantine Fault Tolerance. arXiv:1712.01367 [cs]. [online] Available at: https://arxiv.org/abs/1712.01367.
2. Al-Kuwari, S., Davenport, J.H. and Bradford, R.J. (2011). Cryptographic Hash Functions: Recent Design Trends and Security Notions. [online] ePrint IACR. Available at: https://eprint.iacr.org/2011/565.
3. Anjani Barhanpure, Paaras Belandor and Das, B. (2018). Proof of Stack Consensus for Blockchain Networks. doi:https://doi.org/10.1007/978-981-13-5826-5_8.
4. Atlam, H.F., Alenezi, A., Alassafi, M.O. and Wills, G.B. (2018). Blockchain with Internet of Things: Benefits, Challenges, and Future Directions. International Journal of Intelligent Systems and Applications, 10(6), pp.40–48. doi:https://doi.org/10.5815/ijisa.2018.06.05.
5. Bahga, A. and Madisetti, V. (2014). Internet of Things: A Hands-On Approach. [online] Google Books. VPT. Available at: https://books.google.com/books?hl=el&lr=&id=JPKGBAAAQBAJ&oi=fnd&pg=PA20&dq=A.+Bahga.
6. Bahga, A. and Madisetti, V.K. (2016). Blockchain Platform for Industrial Internet of Things. Journal of Software Engineering and Applications, [online] 09(10), pp.533–546. doi:https://doi.org/10.4236/jsea.2016.910036.
7. Barbian, G., and Mellentin, F. (2021). The cardano proof-of-stake protocol “ouroboros”.
8. Belfer, R. (2020). Proof-of-Activity Consensus Protocol Based on a Network’s Active Nodes. [online] Available at: https://ceur-ws.org/Vol-2623/paper21.pdf.
9. Biryukov, A., Khovratovich, D. and Pustogarov, I. (2014). Deanonymisation of Clients in Bitcoin P2P Network. Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security. doi:https://doi.org/10.1145/2660267.2660379.
10. Biswas, S., Sharif, K., Li, F., Maharjan, S., Mohanty, S.P. and Wang, Y. (2020). PoBT: A Lightweight Consensus Algorithm for Scalable IoT Business Blockchain. IEEE Internet of Things Journal, [online] 7(3), pp.2343–2355. doi:https://doi.org/10.1109/JIOT.2019.2958077.
11. Bodkhe, U., Mehta, D., Tanwar, S., Bhattacharya, P., Singh, P.K. and Hong, W.-C. (2020). A Survey on Decentralized Consensus Mechanisms for Cyber Physical Systems. IEEE Access, 8, pp.54371–54401. doi:https://doi.org/10.1109/access.2020.2981415.
12. Bruyn, A. S. (2017). Blockchain an introduction. University Amsterdam, 26.
13. Buterin, V. (2014). A NEXT GENERATION SMART CONTRACT & DECENTRALIZED APPLICATION PLATFORM. [online] Available at: https://www.weusecoins.com/assets/pdf/library/Ethereum_white_paper-a_next_generation_smart_contract_and_decentralized_application_platform-vitalik-buterin.pdf.
14. Buterin, V. (2015). On public and private blockchains.
15. Buterin, V., Ryan, D., Wang, H. W., Tsao, T., & Liang, C. C. (2018). Ethereum 2.0 phase 1–shard data chains.
16. Capocasale, V., Danilo, G. and Perboli, G. (2022). Comparative analysis of permissioned blockchain frameworks for industrial applications. Blockchain: Research and Applications, p.100113. doi:https://doi.org/10.1016/j.bcra.2022.100113.
17. Chen, Y., Li, M., Zhu, X., Fang, K., Ren, Q., Guo, T., Chen, X., Li, C., Zou, Z. and Deng, Y. (2022). An improved algorithm for practical byzantine fault tolerance to large-scale consortium chain. Information Processing & Management, 59(2), p.102884. doi:https://doi.org/10.1016/j.ipm.2022.102884.

18. Cho, H. (2018). ASIC-Resistance of Multi-Hash Proof-of-Work Mechanisms for Blockchain Consensus Protocols. IEEE Access, 6, pp.66210–66222. doi:https://doi.org/10.1109/access.2018.2878895.
19. Christodoulou, K., Iosif, E., Inglezakis, A. and Themistocleous, M. (2020). Consensus Crash Testing: Exploring Ripple's Decentralization Degree in Adversarial Environments. Future Internet, 12(3), p.53. doi:https://doi.org/10.3390/fi12030053.
20. Corso, A. (2019). Performance analysis of proof-of-elapsed-time (poet) consensus in the sawtooth blockchain framework (Master's thesis, University of Oregon).
21. Curran, B. (2018). What is proof of authority consensus? staking your identity on the blockchain. Blockonomi.
22. De Angelis, S. (2018). Assessing Security and Performances of Consensus algorithms for Permissioned Blockchains. [online] arXiv.org. doi:https://doi.org/10.48550/arXiv.1805.03490.
23. 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. In CEUR workshop proceedings (Vol. 2058). CEUR-WS.
24. Dwork, C. and Naor, M. (1992). Pricing via Processing or Combatting Junk Mail. Advances in Cryptology — CRYPTO' 92, [online] pp.139–147. doi:https://doi.org/10.1007/3-540-48071-4_10.
25. Elijah , N. (2022). What is Proof of Burn?.
26. Eyal, I. and Sirer, E.G. (2018). Majority is not enough. Communications of the ACM, [online] 61(7), pp.95–102. doi:https://doi.org/10.1145/3212998.
27. Fan, X. and Chai, Q. (2018). Roll-DPoS. doi:https://doi.org/10.1145/3286978.3287023.
28. Foroglou, G., and Tsilidou, A. L. (2015, May). Further applications of the blockchain. In 12th student conference on managerial science and technology (Vol. 9). Athens University of Economics and Business, Athens, Greece.
29. Fu, X., Wang, H. and Shi, P. (2020). A survey of Blockchain consensus algorithms: mechanism, design and applications. Science China Information Sciences, 64(2). doi:https://doi.org/10.1007/s11432-019-2790-1.
30. Gao, S., Yu, T., Zhu, J. and Cai, W. (2019). T-PBFT: An EigenTrust-based practical Byzantine fault tolerance consensus algorithm. China Communications, 16(12), pp.111–123. doi:https://doi.org/10.23919/jcc.2019.12.008.
31. Gervais, A., Karame, G.O., Wüst, K., Glykantzis, V., Ritzdorf, H. and Capkun, S. (2016). On the Security and Performance of Proof of Work Blockchains. Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security - CCS'16. [online] doi:https://doi.org/10.1145/2976749.2978341.
32. Ghimire, S. and Selvaraj, H. (2018). A Survey on Bitcoin Cryptocurrency and its Mining. [online] IEEE Xplore. doi:https://doi.org/10.1109/ICSENG.2018.8638208.
33. Golosova, J.,and Romanovs, A. (2018). The Advantages and Disadvantages of the Blockchain Technology. 2018 IEEE 6th Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE), 1(1). https://doi.org/10.1109/aieee.2018.8592253
34. Görkey, I., El Moussaoui, C., Wijdeveld, V., & Sennema, E. (2020). Comparative study of byzantine fault tolerant consensus algorithms on permissioned blockchains.
35. Gramoli, V. (2017). From blockchain consensus back to Byzantine consensus. Future Generation Computer Systems, 107, pp.760–769. doi:https://doi.org/10.1016/j.future.2017.09.023.
36. Grewal, S. (2018). Komodo's delayed proof of work (dpow) security, explained.
37. Haffke, F. (2017). Technical Analysis of Established Blockchain Systems. [online] Available at: https://wwwmatthes.in.tum.de/file/6h5slbhs3i2p/Sebis-Public-Website/Student-Theses-

Guided-Research/Current-Bachelor-s-and-Master-s-Theses/Master-Thesis-von-Florian-Haffke/Master.
38. Hameed, B.I. (2019). Blockchain and Cryptocurrencies Technology: a survey. JOIV : International Journal on Informatics Visualization, 3(4). doi:https://doi.org/10.30630/joiv.3.4.293.
39. Hassan, A., Ali, Md.I., Ahammed, R., Khan, M.M., Alsufyani, N. and Alsufyani, A. (2021). Secured Insurance Framework Using Blockchain and Smart Contract. Scientific Programming, 2021, pp.1–11. doi:https://doi.org/10.1155/2021/6787406.
40. Hazari, S.S. and Mahmoud, Q.H. (2019). A Parallel Proof of Work to Improve Transaction Speed and Scalability in Blockchain Systems. 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC). doi:https://doi.org/10.1109/ccwc.2019.8666535.
41. Henrique, L., Serge Fdida and Duarte, O. (2001). Hop by hop multicast routing protocol. HAL (Le Centre pour la Communication Scientifique Directe). doi:https://doi.org/10.1145/383059.383079.
42. Hileman, G. (2016). State of blockchain q1 2016: Blockchain funding overtakes bitcoin. CoinDesk, New York, NY, May, 11.
43. Howard, H. and Mortier, R. (2020). Paxos vs Raft. Proceedings of the 7thWorkshop on Principles and Practice of Consistency for Distributed Data. doi:https://doi.org/10.1145/3380787.3393681.
44. Hu, Q., Yan, B., Han, Y. and Yu, J. (2021). An Improved Delegated Proof of Stake Consensus Algorithm. Procedia Computer Science, [online] 187, pp.341–346. doi:https://doi.org/10.1016/j.procs.2021.04.109.
45. Ismail, L., & Materwala, H. (2019). A review of blockchain architecture and consensus protocols: Use cases, challenges, and solutions. Symmetry, 11(10), 1198.
46. Ivanov, A., Babichenko, Y., Kanunnikov, H., Karpus, P., Foiu-Khatskevych, L., Kravchenko, R., Gorokhovskyi, K. and Nevmerzhitskyi, I. (2018). Technical Comparison Aspects of Leading Blockchain-Based Platforms on Key Characteristics. NaUKMA Research Papers. Computer Science, 1(0), pp.58–64. doi:https://doi.org/10.18523/2617-3808.2018.58-64.
47. Jalalzai, M.M., Busch, C. and Richard, G.G. (2019). Proteus: A Scalable BFT Consensus Protocol for Blockchains. 2019 IEEE International Conference on Blockchain (Blockchain). doi:https://doi.org/10.1109/blockchain.2019.00048.
48. Kosba, A., Miller, A., Shi, E., Wen, Z. and Papamanthou, C. (2016). Hawk: The Blockchain Model of Cryptography and Privacy-Preserving Smart Contracts. 2016 IEEE Symposium on Security and Privacy (SP), pp.839–858. doi:https://doi.org/10.1109/sp.2016.55.
49. Kuo, L. (2015). Info Books. Tu.ac.th. [online] doi:https://digital.library.tu.ac.th/tu_dc/frontend/Info/item/147850.
50. Kuperberg, M. (2020). Towards an Analysis of Network Partitioning Prevention for Distributed Ledgers and Blockchains. [online] IEEE Xplore. doi:https://doi.org/10.1109/DAPPS49028.2020.00011.
51. Li, A., Wei, X. and He, Z. (2020). Robust Proof of Stake: A New Consensus Protocol for Sustainable Blockchain Systems. Sustainability, 12(7), p.2824. doi:https://doi.org/10.3390/su12072824.
52. Li, K., Li, H., Hou, H., Li, K. and Chen, Y. (2017). Proof of Vote: A High-Performance Consensus Protocol Based on Vote Mechanism & Consortium Blockchain. 2017 IEEE 19th International Conference on High Performance Computing and Communications; IEEE 15th International Conference on Smart City; IEEE 3rd International Conference on Data Science and Systems (HPCC/SmartCity/DSS). doi:https://doi.org/10.1109/hpcc-smartcity-dss.2017.61.

53. Link, G. and Lumbard, K. (2016). Communication Technology Selection in a Young Open Source Community SIGOPEN Developmental Workshop at ICIS. [online] p.1. Available at: http://www.georglink.de/media/2019/02/2016paper-Communication-Technology-Selection-in-a-Young-Open.pdf.
54. Liu, Y., Qian, K., Chen, J., Wang, K. and He, L. (2020). Effective Scaling of Blockchain Beyond Consensus Innovations and Moore's Law. [online] arXiv.org. doi:https://doi.org/10.48550/arXiv.2001.01865.
55. Manolache, M.A., Manolache, S. and Tapus, N. (2022). Decision Making using the Blockchain Proof of Authority Consensus. Procedia Computer Science, 199, pp.580–588. doi:https://doi.org/10.1016/j.procs.2022.01.071.
56. Mechanic, Q. (2011). Proof of stake instead of proof of work. In Bitcoin forum.
57. Meena (2024). different consensus protocols in blockchain. [online] SlideShare. Available at: https://www.slideshare.net/slideshow/different-consensus-protocols-in-blockchianpptx/265419554.
58. Meling, H. and Jehl, L. (2013). Tutorial Summary: Paxos Explained from Scratch. Lecture notes in computer science, pp.1–10. doi:https://doi.org/10.1007/978-3-319-03850-6_1.
59. Menon, A.A., T. Saranya, Sheetal Sureshbabu and Mahesh, A.S. (2021). A Comparatıve Analysis on Three Consensus Algorithms. Lecture notes on data engineering and communications technologies, pp.369–383. doi:https://doi.org/10.1007/978-981-16-3728-5_28.
60. Monegro, J. (2017). Fat Protocols. Union Square Ventures.
61. Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. [online] Available at: https://assets.pubpub.org/d8wct41f/31611263538139.pdf.
62. Nasir, Q., Qasse, I.A., Abu Talib, M. and Nassif, A.B. (2018). Performance Analysis of Hyperledger Fabric Platforms. Security and Communication Networks, 2018, pp.1–14. doi:https://doi.org/10.1155/2018/3976093.
63. Nguyen, G. T., and Kim, K. (2018). A survey about consensus algorithms used in blockchain. Journal of Information processing systems, 14(1).
64. Noyes, C. (2016). BitAV: Fast Anti-Malware by Distributed Blockchain Consensus and Feedforward Scanning. arxiv.org. [online] Available at: https://arxiv.org/abs/1601.01405.
65. Ongaro, D. and Ousterhout, J. (2014). In Search of an Understandable Consensus Algorithm. [online] www.usenix.org. Available at: https://www.usenix.org/conference/atc14/technical-sessions/presentation/ongaro.
66. Panda, S.S., Mohanta, B.K., Satapathy, U., Jena, D., Gountia, D. and Patra, T.K. (2019). Study of Blockchain Based Decentralized Consensus Algorithms. [online] IEEE Xplore. doi:https://doi.org/10.1109/TENCON.2019.8929439.
67. s, G.W., Panayi, E. and Chapelle, A. (2015). Trends in crypto-currencies and blockchain technologies: A monetary theory and regulation perspective. arXiv:1508.04364 [cs]. [online] Available at: https://arxiv.org/abs/1508.04364.
68. Polge, J., Robert, J. and Le Traon, Y. (2020). Permissioned blockchain frameworks in the industry: A comparison. ICT Express, 7(2). doi:https://doi.org/10.1016/j.icte.2020.09.002.
69. Primer, T. (2018). Blockchain. no. July, 0-38.
70. Rebello, G.A.F., Camilo, G.F., Guimaraes, L.C.B., de Souza, L.A.C. and Duarte, O.C.M.B. (2020). On the Security and Performance of Proof-based Consensus Protocols. 2020 4th Conference on Cloud and Internet of Things (CIoT). doi:https://doi.org/10.1109/ciot50422.2020.9244295.
71. Rebello, G.A.F., Camilo, G.F., Guimarães, L.C.B., de Souza, L.A.C., Thomaz, G.A. and Duarte, O.C.M.B. (2021). A security and performance analysis of proof-based consensus

protocols. Annals of Telecommunications, 77(7-8), pp.517–537. doi:https://doi.org/10.1007/s12243-021-00896-2.
72. Reed, J. (2017). Litecoin: An introduction to litecoin cryptocurrency and litecoin mining.
73. Samid, G. (2015). Tethered Money: Managing Digital Currency Transactions. [online] Google Books. Academic Press. Available at: https://books.google.com/books?hl=el&lr=&id=FE-2BgAAQBAJ&oi=fnd&pg=PP1&dq=Decred+-+Autonomous+Digital+Currency.+&ots=v6sRqSrlRD&sig=N932pO3ujB5vLh3k5ImOUQTL4Bs.
74. Sarkar, P. (2019). A New Blockchain Proposal Supporting Multi-Stage Proof-of-Work. Cryptology ePrint Archive. [online] Available at: https://eprint.iacr.org/2019/162.
75. Sathya, A. R., Panda, S. K., and Hanumanthakari, S. (2021). Enabling Smart Education System Using Blockchain Technology. Intelligent Systems Reference Library, 169–177. https://doi.org/10.1007/978-3-030-69395-4_10
76. Sayeed, S. and Marco-Gisbert, H. (2019). Assessing Blockchain Consensus and Security Mechanisms against the 51% Attack. Applied Sciences, 9(9), p.1788. doi:https://doi.org/10.3390/app9091788.
77. Schwartz, D., Youngs, N. and Britto, A. (2014). The Ripple Protocol Consensus Algorithm. [online] Available at: https://exponentialstocks.com/wp-content/uploads/wpforo/default_attachments/1634556441-ripple_consensus_whitepaper.pdf.
78. Secure, A. (2018). The zilliqa project: A secure, scalable blockchain platform.
79. Sharples, M. and Domingue, J. (2016). The Blockchain and Kudos: A Distributed System for Educational Record, Reputation and Reward. Adaptive and Adaptable Learning, 9891, pp.490–496. doi:https://doi.org/10.1007/978-3-319-45153-4_48.
80. Shifferaw, Y. and Lemma, S. (2021). Limitations of proof of stake algorithm in blockchain: A review. Zede Journal, [online] 39(1), pp.81–95. Available at: https://www.ajol.info/index.php/zj/article/view/216947.
81. Shuliar, T. and Goldsmit, N. (2019). PROOF OF VALUE ALIENATION (PoVA) - a concept of a cryptocurrency issuance protocol. [online] arXiv.org. doi:https://doi.org/10.48550/arXiv.1901.04928.
82. Singh, A., Kumar, G., Saha, R., Conti, M., Alazab, M. and Thomas, R. (2022). A survey and taxonomy of consensus protocols for blockchains. Journal of Systems Architecture, p.102503. doi:https://doi.org/10.1016/j.sysarc.2022.102503.
83. Sukhwani, H., Martínez, J.M., Chang, X., Trivedi, K.S. and Rindos, A. (2017). Performance Modeling of PBFT Consensus Process for Permissioned Blockchain Network (Hyperledger Fabric). [online] IEEE Xplore. doi:https://doi.org/10.1109/SRDS.2017.36.
84. Suresh, A., Nair, A.R., Lal, A., Mohana Kumaran S and Greeshma Sarath (2020). A Hybrid Proof based Consensus Algorithm for Permission less Blockchain. 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA). doi:https://doi.org/10.1109/icirca48905.2020.9183109.
85. Todd, P. (2015). Ripple Protocol Consensus Algorithm Review. [online] Available at: https://cdn3.baserank.io/reviews/9035a989-c2cd-4b5b-85a7-219abbed0900_RippleProtocolConsensusAlgorithmReview.pdf.
86. Tomić, N.Z. (2021). A Review of consensus protocols in permissioned blockchains. Journal of Computer Science Research, 3(2). doi:https://doi.org/10.30564/jcsr.v3i2.2921.
87. Viriyasitavat, W. and Hoonsopon, D. (2019). Blockchain characteristics and consensus in modern business processes. Journal of Industrial Information Integration, [online] 13,

pp.32–39. Available at: https://www.sciencedirect.com/science/article/pii/S2452414X18300815.
88. Vranken, H. (2017). Sustainability of bitcoin and blockchains. Current Opinion in Environmental Sustainability, 28(1877-3435), pp.1–9. doi:https://doi.org/10.1016/j.cosust.2017.04.011.
89. Vukolić, M. (2016). The Quest for Scalable Blockchain Fabric: Proof-of-Work vs. BFT Replication. Open Problems in Network Security, pp.112–125. doi:https://doi.org/10.1007/978-3-319-39028-4_9.
90. Wahab, A. and Mehmood, W. (2018). Survey of Consensus Protocols. [online] arXiv.org. Available at: https://arxiv.org/abs/1810.03357.
91. Wai Kok Chan, Chin, J.-J. and Vik Tor Goh (2020). Proof of Bid as Alternative to Proof of Work. Communications in computer and information science, pp.60–73. doi:https://doi.org/10.1007/978-981-15-2693-0_5.
92. Wang, Q., Huang, J., Wang, S., Chen, Y., Zhang, P. and He, L. (2020). A Comparative Study of Blockchain Consensus Algorithms. Journal of Physics: Conference Series, 1437, p.012007. doi:https://doi.org/10.1088/1742-6596/1437/1/012007.
93. Wang, T., Zhao, C., Yang, Q., Zhang, S., and Liew, S. C. (2021). Ethna: Analyzing the underlying peer-to-peer network of ethereum blockchain. IEEE Transactions on Network Science and Engineering, 8(3), 2131-2146.
94. Wang, Y., Peng, H., Su, Z., Luan, T.H., Benslimane, A. and Wu, Y. (2022). A Platform-Free Proof of Federated Learning Consensus Mechanism for Sustainable Blockchains. IEEE Journal on Selected Areas in Communications, 40(12), pp.3305–3324. doi:https://doi.org/10.1109/jsac.2022.3213347.
95. =wo, A. T., Sulistyono, M. T., and Hariadi, M. (2020). Cryptospatial coordinate using the RPCA based on a point in polygon test for cultural heritage tourism. Communications-Scientific letters of the University of Zilina, 22(4), 211-217.
96. Wu, X., Qiu, H., Zhang, S., Memmi, G., Gai, K. and Cai, W. (2020). ChainIDE 2.0: Facilitating Smart Contract Development for Consortium Blockchain. [online] IEEE Xplore. doi:https://doi.org/10.1109/INFOCOMWKSHPS50562.2020.9163051.
97. Wu, Y., Song, P. and Wang, F. (2020). Hybrid Consensus Algorithm Optimization: A Mathematical Method Based on POS and PBFT and Its Application in Blockchain. Mathematical Problems in Engineering, 2020, pp.1–13. doi:https://doi.org/10.1155/2020/7270624.
98. Xiao, B., Jin, C., Li, Z., Zhu, B., Li, X. and Wang, D. (2021). Proof of Importance: A Consensus Algorithm for Importance Based on Dynamic Authorization. IEEE/WIC/ACM International Conference on Web Intelligence. doi:https://doi.org/10.1145/3498851.3499007.
99. Xu, H., Long, Y., Liu, Z., Liu, Z. and Gu, D. (2018). Dynamic Practical Byzantine Fault Tolerance. doi:https://doi.org/10.1109/cns.2018.8433150.
100. Xu, H., Zhang, L., Liu, Y. and Cao, B. (2020). RAFT Based Wireless Blockchain Networks in the Presence of Malicious Jamming. IEEE Wireless Communications Letters, 9(6), pp.817–821. doi:https://doi.org/10.1109/lwc.2020.2971469.
101. Xu, J., Lin, J., Liang, W. and Li, K.-C. (2021). Privacy preserving personalized blockchain reliability prediction via federated learning in IoT environments. Cluster Computing. doi:https://doi.org/10.1007/s10586-021-03399-w.
102. Xu, J., Wang, C. and Jia, X. (2023). A Survey of Blockchain Consensus Protocols. ACM Computing Surveys. doi:https://doi.org/10.1145/3579845.

103. Xu, J., Wang, W., Zeng, Y., Yan, Z. and Li, H. (2022). Raft-PLUS: Improving Raft by Multi-Policy Based Leader Election with Unprejudiced Sorting. Symmetry (20738994), [online] 14(6), p.N.PAG–N.PAG. doi:https://doi.org/10.3390/sym14061122.
104. Yadav, A.K. and Singh, K. (2020). Comparative Analysis of Consensus Algorithms of Blockchain Technology. Advances in Intelligent Systems and Computing, pp.205–218. doi:https://doi.org/10.1007/978-981-15-1518-7_17.
105. Yaga, D., Mell, P., Roby, N. and Scarfone, K. (2018). Blockchain Technology Overview. National Institute of Standards and Technology, [online] 1(1). doi:https://doi.org/10.6028/nist.ir.8202.
106. 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, pp.118541–118555. doi:https://doi.org/10.1109/access.2019.2935149.
107. Zhang, S. and Lee, J.-H. (2019). Analysis of the main consensus protocols of blockchain. ICT Express, 6(2). doi:https://doi.org/10.1016/j.icte.2019.08.001.
108. Zhang, Y. and Wen, J. (2015). An IoT electric business model based on the protocol of bitcoin. 2015 18th International Conference on Intelligence in Next Generation Networks. doi:https://doi.org/10.1109/icin.2015.7073830.
109. Zheng, Z., Xie, S., Dai, H., Chen, X. and Wang, H. (2017). An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends. 2017 IEEE International Congress on Big Data (BigData Congress).
110. Zhou, S., Li, K., Xiao, L., Cai, J., Liang, W. and Castiglione, A. (2023). A Systematic Review of Consensus Mechanisms in Blockchain. Mathematics, [online] 11(10), p.2248. doi:https://doi.org/10.3390/math11102248.
111. Zîle, K., and Strazdiòa, R. Blockchain Use Cases and Their Feasibility, Applied Computer Systems, 23 (2018) 12–20.