COMPARISON OF ACCLAIMED CONSENSUS ALGORITHM

MINT-709 CAPSTONE PROJECT REPORT

APRIL 5, 2020

RAJI CHOKKALINGAM

Table of Contents

TABLE OF FIGURES

COMPARISON OF ACCLAIMED BLOCKCHAIN CONSENSUS ALGORITHMS

1 INTRODUCTION

The usage of cryptocurrencies is gaining momentum with the mainstream economists, and so is the underlying blockchain technology. Blockchain is a distributed digital ledger, where peer-to-peer network validates the transactions using consensus algorithms. More than cryptocurrencies, blockchain technology will be substantially involved in financial services, healthcare, government services, and the Internet of Things where there are digital transactions that require authentication and validation. Blockchain technology helps to maintain the records in a decentralized manner without compromising the veracity of the records in it. Currently, there are over fifteen consensus algorithms to implement blockchain technology, and the list is only growing every year.

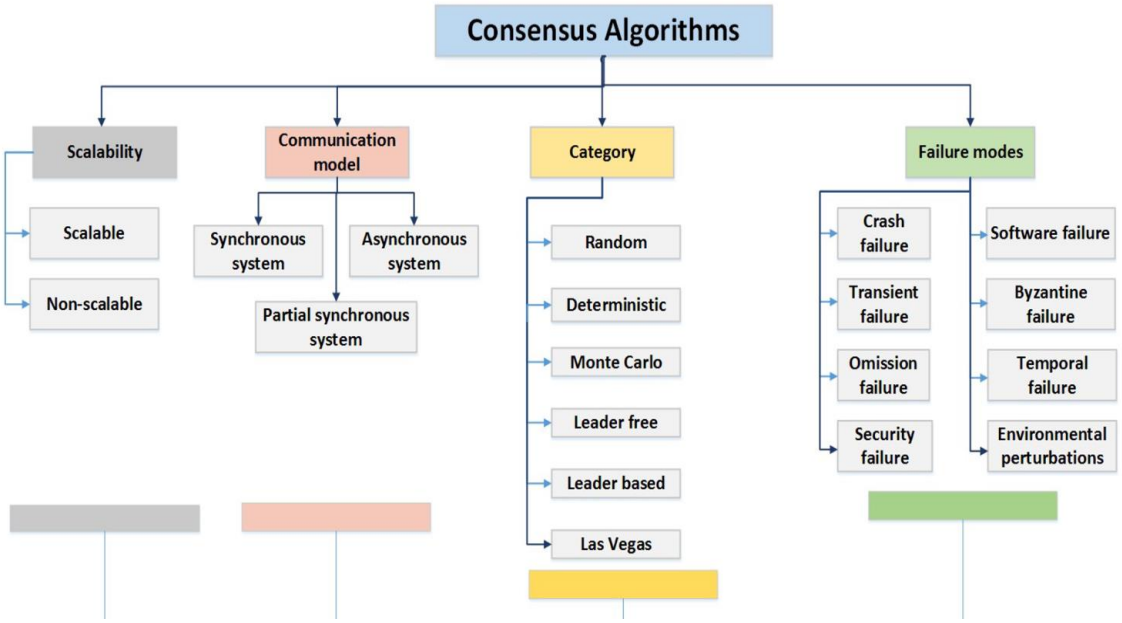


Figure 1 Broad Categorization of consensus algorithms [1]

The blockchain consensus algorithms are classified into a broad range. They are mainly classified as randomized, deterministic, Monte Carlo, Las Vegas, leader based, and leader - free algorithms. The randomized consensus provides a guarantee to an agreement, validity, and termination properties with some probability value. The deterministic is the opposite of randomized, and they do not provide a probability value. The Monte Carlo consensus functions by running the Monte Carlo algorithms on each node or process with a specific set of data. The Las Vegas consensus has probability with each consensus round. The main difference between Monte Carlo and Las Vegas is the running time - The monte Carlo has deterministic running time which means there is no probability value associated with it, on the other hand, the Las Vegas have probabilistic running time. The leader-based consensus has the authority to terminate on consensus when needed. Moreover, each algorithm is

categorized based on a communication model like synchronous, partial synchronous, asynchronous.

The above mentioned is a general categorization of consensus algorithm, consensus algorithms when explicitly based on the blockchain, they are classified into two main types: proof-based and vote based. In the proof-based consensus, when a node wants to join a network, then it must prove itself that they are better than other nodes that are carrying out the appending work. In the vote-based consensus, each node is liable to communicate and exchange the new transaction block that it verifies to the other nodes in the network. The final decision is made based on the majority of nodes. For example, node Z can append a block 'x' to the blockchain only when at least T nodes append the same block x to the blockchain, where T is the threshold parameter decided by the system. By itself, blockchain is categorized into three different types, Private, public, and consortium. The type of blockchain implemented in design determines the membership control in the consensus algorithm. The type of blockchain requirement is decided based on the application or business demands. When evaluating a consensus algorithm, knowing the type of blockchain is vital to understand the membership control

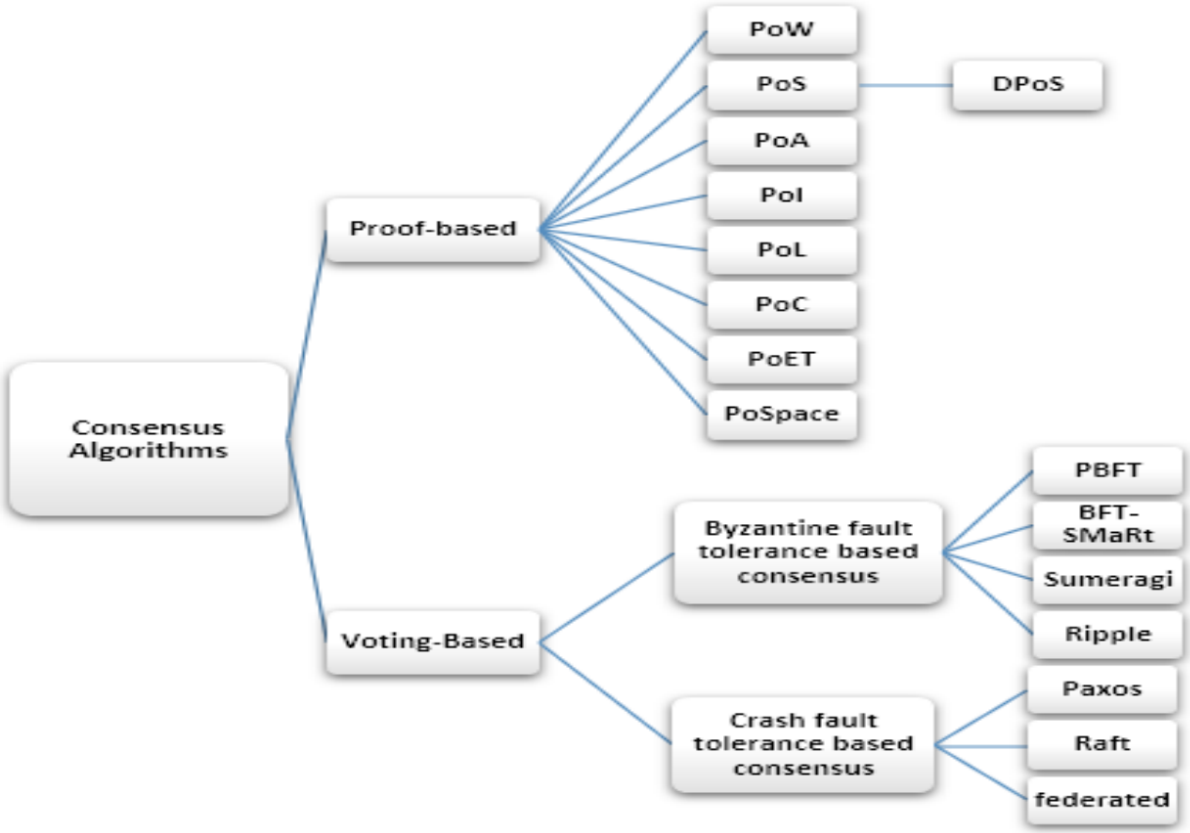


Figure 2 Consensus Algorithm Types [1]

The objective of the proposed project is to compare and evaluate blockchain consensus algorithms that are widely in use (i.e., Proof of Work (PoW), Proof of Stake (PoS), Delegated Proof of Stake (DPoS)and Practical Byzantine Fault Tolerance (PBFT). The metrics that will be used to compare the algorithms are:

SALIENT FEATURES	Node Identity Management
	Data Model
	Communication Model
	Energy Savings
	Tolerated Power of Adversary
INCENTIVE	Transaction Fees
	Block Reward
PERFORMANCE	Communication Complexity
	Verification Speed
	Throughput
	Scalability
THREAT EXPOSURE	Sybil Attack
	51% Attack
	Double Spending

Although strides have been made in analyzing the various consensus algorithms, there is enough motivation to compare, further investigate and make recommendations that will be helpful in future implementations. The report is organized in a manner to provide a background of the metrics used, consensus algorithm description, algorithm implementation using program, comparison of the selected algorithms and the conclusion.

2 BACKGROUND

The consensus is a term that goes hand in hand with a distributed system, a distributed system here is a network which contains multiple nodes and processes but needs to maintain a standard among the network. The consensus algorithm is the backbone of blockchain technology. Blockchain is a decentralized distributed ledger system; a consensus algorithm is required among the peer nodes for its proper working. One consensus algorithm is not enough for application with dynamic demands. The principal concept of the consensus algorithm is to adapt to a collective agreement about the current state of the distributed ledger. All the nodes within a network need to agree upon the decision made by the majority, whether they like it or not. Even though there is no central authority, the transactions in the blockchain are considered safe and verified.

In the blockchain, the loose trilemma law exists, the law sets that the blockchain can have at most two of three properties like (i) decentralization (ii) scalability and (iii) security [2]. In

permission-less blockchains, the transactions and blocks are transmitted to every peer in the network, tested, and recorded by all the participants in that decentralized peer-peer network. These characteristic makes the whole system perpetual, stable, and resistant when more than half of the nodes cooperate, to be honest. The majority of the participants are required to be honest for the security property to be proper, but it is expensive in terms of scalability as all the participants need to be informed and need to agree implicitly. The decentralization property is nothing but no single point of control, and here it is defined similarly to redundancy. The permissioned blockchain makes a trade-off so that it only allows specific participants to control the underlying ledger [2].

2.1 Decentralization

We know blockchain is a decentralized distributed ledger; the decentralization has been a critical component that led to a massive growth of many blockchains. The decentralization comes with its limitations in terms of scalability, it limits blockchain from achieving the desired scalability. Some findings have proved that there are many hurdles linked when trying to achieve decentralization, it is a challenge due to the skewed mining powers and blockchains that are entirely decentralized are inherently limited to scalability as it influences a throughput upper bound and prevents calibrating smart contract execution.

2.1.1 Quantifying Decentralization

This section discusses decentralization as a quantitative measure, and this measure helps us in finding the blockchain improvements. In recent years the decentralization of system nodes has gained vast attention, and it has been the critical component of blockchains to democratize trust.

Before we discuss decentralization, we need to understand what centralization means? Centralization is a quantitative measure that reports the degree of centralization in blockchains. This measure represents the distribution of transactions contributed by blockchain providers. Decentralization has some adverse effects on the layers like (i) Physical layer; assumption of the decentralization of mining power does not hold as the distribution of mining powers in the real-world is skewed. (ii) Platform Software layer, decentralization causes scalability issues in the transactions throughput of blockchains as there are proofs that decentralization causes low upper bound of transaction throughput of the blockchains, which are independent of specific protocols. (iii) Smart Contract layer, in this layer, the decentralized blockchains, accomplish the replicated execution and sequential programming models, which results in preventing the scalability options of the smart contract layer from execution.

2.1.2 Measure

Using the centralization level, we are going to describe the concept of decentralization in blockchains. A blockchain $N\epsilon$ is centralized when the top N nodes outperform the $1 - \epsilon$ fraction of transactions. If the N value is smaller, then the blockchain is more centralized

for the same ϵ. Figure 4 shows the decentralization in different types of blockchains. If the small ϵ is given, then the public chains would incur a more significant N value, which automatically results in a poor or lower level of centralization. When $\epsilon = 0$, the consortium chains incur a lower centralization N0. The private chains are otherwise known as extreme cases of consortium chains, and they are fully centralized N0=1. The centralization level also able us to investigate central trust. The Nakamoto consensus requires 51% of computing power or any form of mining to tolerate Byzantine faults. Therefore, the level of the public chain of central trust is T = N0.49. Whereas in the Practical Byzantine Fault Tolerance, it needs to trust $(2n+1)/(3n+1) \approx 67\%$ nodes to withstand the Byzantine faults. So, the consortium chain's central trust levels are T = N0.33. The private chain's trust levels are T=1. The analysis shows that when the trust levels are low, then the central trust levels are high. This interpretation shows us the quantitative analysis of decentralization and the scalability of various blockchains up-gradation achieved. The decentralization in blockchain introduces some inherent problems not only in the blockchain providers but also in the full stack [3].

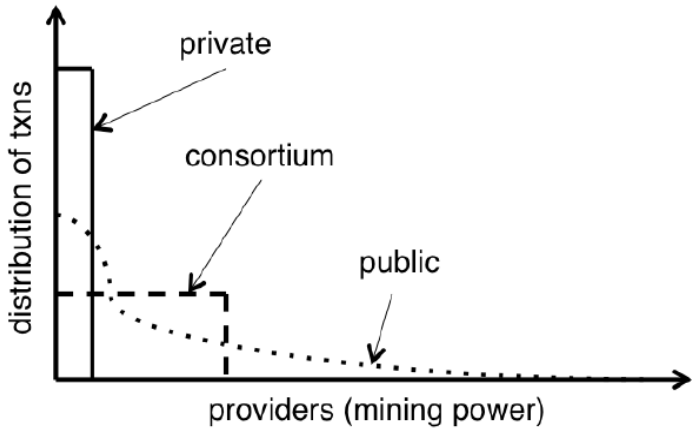


Figure 3 Cumulative Distributions - No. of Transactions By Providers [4]

Type	Centralization Level	Central Trust	Consensus	Mining	Examples
Public	$N_\epsilon, \exists\ c, \epsilon>0 \rightarrow N_\epsilon>c$	$N_{0.49}$	Nakamoto	PoW, PoS, DPoS	Bitcoin, Ethereum, EOS
Consortium	$N_\epsilon, \exists\ c \rightarrow N_0<c$	$N_{0.33}$	PBFT	N/A	Hyperledger Fabric
Private	$N_0=1$	1	N/A	N/A	N/A

2.2 Node Identity Management

There are four types of Blockchains networks (i) public blockchain (ii) private blockchain (iii) consortium blockchain (iv) hybrid blockchain

Public blockchain - This type of blockchain does not have any access restrictions. Public blockchains can be accessed by anyone for transactions and become a validator. Proof of work consensus blockchain and Proof of Stake blockchain are some of the most used public blockchains.

Private blockchain - Unlike public blockchains, they have restrictions, and it is permissioned. The participants and validators need to have permission from the network administrator or concerned people to use this blockchain.

Consortium Blockchain - It is a partially decentralized blockchain. The consortium is a combination of the public and private blockchain type. Limited nodes control the consensus mechanism of this blockchain. The access to this blockchain can be either to a predetermined set of participants or made available to the public.

Hybrid blockchain - This blockchain is a combination of centralization, and decentralization features, the working of it depends on which portion of centralization and decentralization is being practiced.

We shall focus more on public and private blockchains as some of the famous consensus algorithms are a part of public and private blockchains [5].

2.2.1 Analysis between Public vs. Private blockchains

In a public blockchain network, any node can participate in the transaction or be a validator in a peer-to-peer network, and they can leave the network when they wish to. It is merely a permission-less network. The public blockchains are decentralized networks with no single entity controlling the network. The data on the public chain is secure and cannot be modified or tampered once the blockchain validates them. Bitcoin based on PoW, Ethereum based on PoS, are some of the examples which fall under the public blockchains [30]

The private blockchain, on the other hand, requires the nodes to carry special permission or access to be authenticated to be a part of the peer-to-peer network. Banking sectors and Financial institutions have shown interest in this type of blockchain due to its secure nature. Some studies have also shown that the private blockchain type may disrupt the conventional centralized system, which is currently being used. Hyperledger is one of the most popular types of private blockchains which allow only permissioned participants within the network [31]

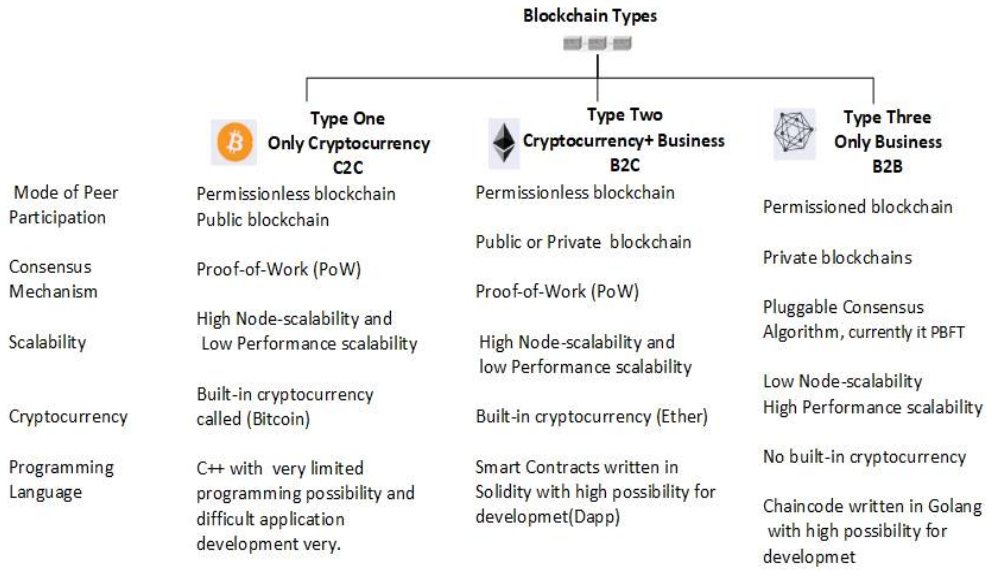


Figure 4 Blockchain Types [6]

2.2.2 Similarities - Public & Private Blockchain

(i) Distributed Ledger - Both public and private blockchains function as distributed ledgers, which means multiple versions of the same data are stored in a distributed nature through a network. Although data is geographically distributed, data once added cannot be altered or removed. Hence, they are permanent records.

(ii) Unaltered Data - Theoretically, data that is added and stored using public and private blockchains are unaltered without enough power over the network. Even if intruders manage to modify the data, the cryptographic hashes also change, which notifies the concerned personnel.

(iii) Consensus mechanism - The public and private blockchains use the consensus mechanism to decide how a ledger must look from an array of versions.

(iv) Redundancy of ledger - These blockchains are decentralized and distributed, but every node has a replica of the ledger over a peer-to-peer network.

(v) The Public and Private blockchains rely on numerous users to authenticate improvisations or correction in a distributed ledger, the edited version is then made available to everyone as a new copy of the existing data [7].

2.2.3 Differences - Public & Private Blockchains

(i) The primary difference between a public and private blockchain, in a public blockchain, anyone can take part and add data to the network, but in a private blockchain,

a participant needs user rights or access to be a part of a network to make necessary changes in a network.

(ii) The public blockchain is decentralized and private is more of a centralized. Examples of public blockchains are Bitcoin and Ethereum, and examples of Private is Hyperledger.

(iii) The possibility of minor collision is less in private blockchains as the validators in the private network have suitable credentials to be a part of the network. Whereas the public blockchains are more prone to collisions or a 51% attack - a group of miners control over 50% of the network's computing power.

(iv) The energy consumption between public and private blockchains vary, the public chain requires more energy in terms of electrical energy to function and attain network consensus, private blockchains consumes less energy.

(v) The public blockchain needs to authenticate a node if it requires to oversee the centralized authority. At this point, the public blockchain transforms into a private blockchain.

(vi) The public blockchain is more secure due to its decentralization, and there is no single node that can take control over the network. The private blockchain, though they authenticate nodes to be a part of the network hackers and intruders, can impose themselves as legit and manipulate or perform data breaches in order to gain control over the network. Thus, they modify the transaction according to their needs.

(vii) The order of magnitude of a public blockchain is lesser when compared to private blockchains as the public provides transactional throughput and lighter.

(viii) There are some consensus algorithms which can be used only in private blockchains like Proof of Elapsed Time (PoET), Raft, Istanbul Byzantine Fault Tolerance.

(ix) The public blockchain has less room for growth in the scalability area as it is slow when it comes to processing the transactions. The private blockchain, the transactions are much faster as they have few nodes that need to manage the data, which makes it a more scalable option [7].

2.2.4 Characteristics of Public Blockchains

(i) Transparency - Due to the default design, public blockchains are bound to be transparent. They are obligated to give incentives to the users to trust the network. The public blockchain, which is a transparent network, must give users all the access except for private keys.

(ii) Decentralized - They are decentralized, which means there is no single node controlling the whole network or can edit the ledger. The public blockchains work based

on the consensus mechanism, so any changes done are achieved only if 51% of the nodes agree to it.

(iii) Digital Assets - This type of blockchain contains user-incentivized tokens whose values differ based on relevancy and state of the blockchain they belong to. Based on the purpose they are designed for, they either use monetary or utility tokens [6].

2.2.5 Characteristics of Private Blockchains

(i) Governance - The business network members make decisions in this blockchain. They depend on various dynamics to settle on the central level. Private blockchains do not depend on the consensus mechanism in order to receive a majority to choose to change.

(ii) Decentralization - The private blockchains have the freedom to choose the level of decentralization. They are mostly centralized, but they can prefer to be partially decentralized as well. Depending on their preference, they can choose any consensus mechanism that they wish to employ.

(iii) Transparency - The private blockchains are not required to be transparent, but they can choose to be one depending on the internal organization's business requirements. In terms of privacy, it depends on the user-case basis. Private blockchains carry extensive data related to user's transactions and other operations [6].

2.2.6 Advantages of Public Blockchains

They are secure when compared to private blockchains, as it is challenging to impersonate as "bad actor" among numerous nodes in the network. The attempts to intrude and manipulate the network is less due to the expansive infrastructure. As public blockchains are open and permission-less, anyone can be a part of the network and verify the transaction's correctness and consistency of the data. As the public chain is open and available to anyone, Participants do not create an additional infrastructure to check the accuracy of the system [6].

2.2.7 Disadvantages of Public Blockchains

The public blockchains are very slow, and they can validate fewer transactions per second, which leads to more energy consumption. The 51% attack is the biggest threat faced by the public blockchains, as a small number of nodes are enough to make the network more susceptible to collisions and hacking [6].

2.2.8Advantages of Private Blockchains

The private blockchains are relatively more scalable and customizable than the public chains. Private chains have a more defined governance structure. They perform more efficiently when compared to public blockchains [6].

2.2.9 Disadvantages of Private Blockchains

As private blockchains are less transparent, the participants within the network lack trust. Although private blockchains are considered secure, this factor depends on the integrity of its members. They are vulnerable to manipulation and hacks if a "bad actor" is within the private network. Due to centralized infrastructure, careful maintenance of a private network, preserving intricate identity, and access management system for users violates the concept of blockchain - decentralized distributed ledger system [6].

2.3 Data Model

An overall conceptual understanding of the data models in the blockchain is essential to understand the framework of these models. The popular cryptocurrency platforms Bitcoin built on proof of Work and Ethereum built on proof of Stake use two different data models. The Bitcoin based on Proof of Work employs the Transaction model as well as the account-based data model depending on the application's requirements. The Ethereum based on Proof of Stake consensus uses the account-based data model only [8].

The role of consensus in cryptocurrency is to secure the blockchain network, validate the state of the blockchain, ensure the data model deployed by the platforms proves possession of the tokens. The consensus algorithms consolidate cryptography and economic incentives to implement correctness and immutability in the network [9]. Our focus here is to know more about the data models, transaction-based, and account-based implemented in the cryptocurrency platforms and other platforms in blockchains. The PoW employs the Unspent Transaction Output Scheme (UTXO), and the PoS based Ethereum uses Account-based data models. The data models UTXO and Account-based have specific roles in the massive structure of the program [8].

The Proof of Work based Bitcoin was the first cryptocurrency platform to use the unspent transaction output scheme (UTXO); back then, it was more of an abstract model. The account-based data model has similarities to the standard banking account-model [8].

2.3.1 Transaction Model

The unspent transaction model used in the PoW based Bitcoin and its derivatives Zcash and Litecoin is more of an abstract type. The UTXO is a vital component in PoW as it allows the transactions to be more transparent; a chain of digital signatures links the transactions.

In UTXO, the token owner transfers their coins to another owner by digitally signing off the hash of the preceding transaction and the public key, which is nothing but the address of the owner who is intended to receive the coins. This act is more of a continuous infraction of input and output. Given, the owner does not own the tokens but retains the output of a specific number of tokens. The output of a definite number of tokens is signed over as an input to the owner expected to receive them, who then considers them as a new

output [36].In a UTXO based ledger, the protocol layer contains coins stored in the form of unspent transaction outputs UTXO. Accounts and wallets are not found in the protocol layer. The transactions are accomplished by using the existing UTXO's and replacing the new UTXO"S in their place. The UTXO has set some standards and criteria for spending the coins; they are explained as three basic schemes [9].

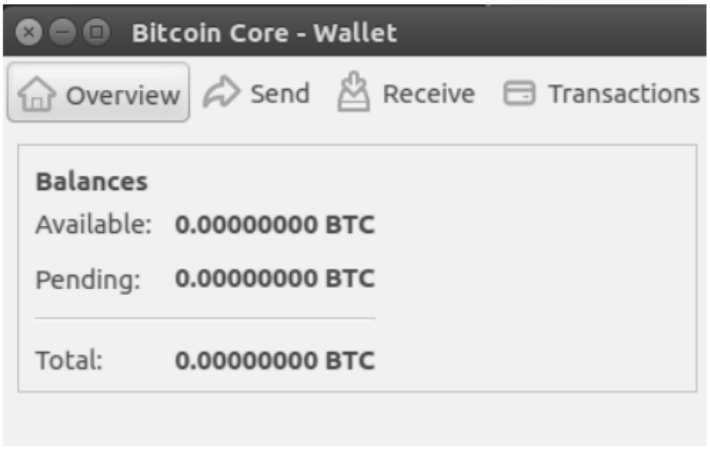


Figure 5 Represents a Bitcoin Wallet [10]

(i) Each transaction's sum of inputs must be higher than the sum of its outputs.

(ii) Every transaction must hold a valid signature of the owner in its input (every input).

(iii) The referenced input must be valid and not spent [8].

The functioning of UTXO is similar to a pile of coins that get transferred if the spending criteria are met. The UTXO can be combined or separated to create the denomination needed for a particular transaction.

To understand the functioning of the UTXO, we are going to consider an example. Alice owns 10 Bitcoins (BTC), and Bob does not own any BTC. The BTC's owned by Alice is a combination of two different transaction outputs, which consists of 6 BTC and 4 BTC. Now Alice wants to transfer the BTC's to Bob, and the wallet picks the best BTC to be transferred to Bob. The 6 BTC gets sent to Bob, and Bob becomes the owner of 5 BTC by having the digital key, digital signature, and the address to prove the purchase of the BTC. Among the 6 BTC sent from Alice to Bob, 5 BTC, are owned by Bob, and the 1 BTC in change is returned to Alice. The returned BTC is called the unspent transaction output UTXO, is sent back to Alice. Alice now owns two outputs, composed of 1 and 4 BTC, respectively. Bob owns one output worth of 5 BTC. Bob, i.e., is the payee who can check the ownership of the tokens by verifying the signatures on a distributed public ledger [8].

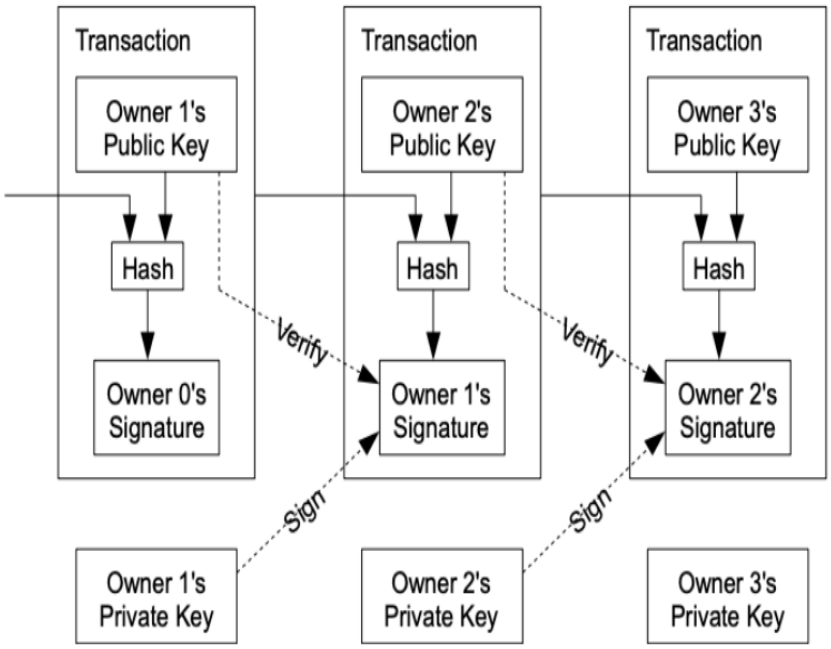


Figure 6 Representation of Transaction in UTWO Data-Based Model [9]

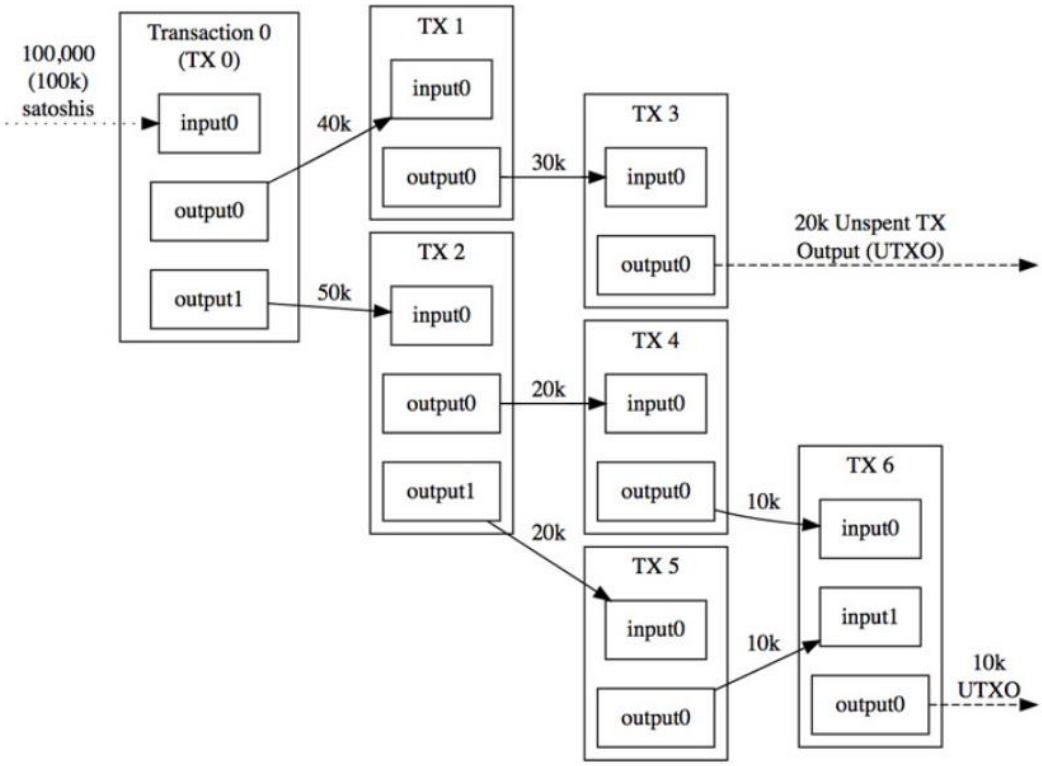


Figure 7 Representing the Transaction to Transaction payment in PoW bitcoin [8]

The UTXO model is relatively simple as it allows the network to be more scalable and less intensive. It also makes the consensus mechanism of Bitcoin, which is Proof of Work to be manageable. Scalability has a significant impact on the performance of the overall network. PoW based Bitcoin can work on multiple scripting types, which makes it more preferred when it comes to processing complex payment logic.

2.3.1.1 Advantages of Transaction Model

(i) The design of the UTXO model has numerous advantages, one of them being compatible with PoW architecture. The notable feature of the scheme is the Simple Payment Verifications (SPV) method, these light wallets make the interaction with the PoW based Bitcoin in a decentralized way and trust-less mode without having the necessity to download the entire blockchain, thus remarkably reducing the storage space. It also supports Phone applications to communicate with the PoW based Bitcoin network.

(ii) The UTXO enables parallel processing capacity across multiple addresses. This factor helps the infrastructure being more scalable. Every input is considered unique and independent of the other, which enables transactions to be processed parallelly. [8] [9]

2.3.1.2 Disadvantages of UTXO

The UTXO transaction scheme can inherently support when only one user owns each output. The application is not suitable for growing smart contracts, and it may crash if more than one owner consumes the same output at the same time. UTXO limits the developers on the amount of state impacted by each output when it comes to a complex application that is built on the UTXO model. Due to the limitation on the spending criteria, smart contracting abilities are bounded as it requires signatures from multiple parties. [8] [9]

2.3.2 Account-Based Model

The account-based models are very different in working compared to the transaction-based model. As said earlier, they are based on the conventional banking account model. In the account-based model, instead of uniquely referencing individual coins, the coins are represented as balance in an account. These accounts can be either controlled by private key or smart contracts. Each account encounters direct value and information transfers with the state transition. There are significant differences between the UTXO and account-based data models. In the UTXO, the transaction gives away information about the resulting state; this way, even before the transaction begins, we will get to know the results. Every transaction carries information about the resulting coin's location. Unlike UTXO, the account-based models are called stateful models, as the transactions in this data model rely on the existing state. The statefulness feature in the account-based model makes it more flexible than the UTXO data model.

To illustrate the working of the account-based model, we shall consider an example, Alice and Bob. Alice and Bob want to interact with each other via transactions. Alice has ten tokens while Bob has nothing in his account. Alice transfers five tokens to Bob, now there are five tokens in Alice's account, and Bob has five tokens in his account transferred to him by Alice. This is a simple transaction based on the standard banking account model. Due to its inherent simple design, the transactions are easily traceable. The account-based model prevents the double-spending attack as centralized authorities monitor the complete flow of the transaction [8] [9].

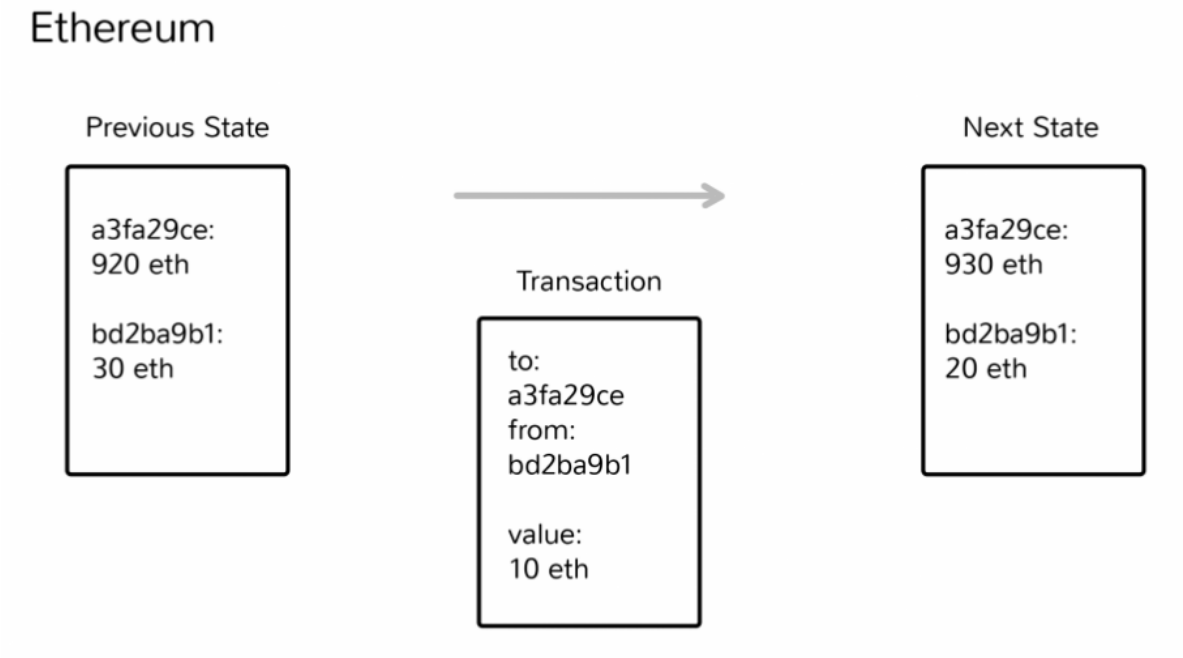


Figure 8 Ethereum Transaction state [11]

The PoS based Ethereum has two types of accounts, the private-key controlled user accounts and contract-code controlled user accounts- which is nothing but the smart contract. This happens to be the reason why PoS chose the account-based model over UTXO. Proof of Stake based Ethereum employs Turing Complete programming language; the main feature of the program is the smart contracts. The PoS based Ethereum has an ample amount of decentralized applications that contain arbitrary state and code. Now it makes sense as to why PoS employs account-based instead of UTXO, as UTXO may limit the execution of smart contracts. Account-based models are much simpler compared to UTXO due to their straightforward transaction flow.

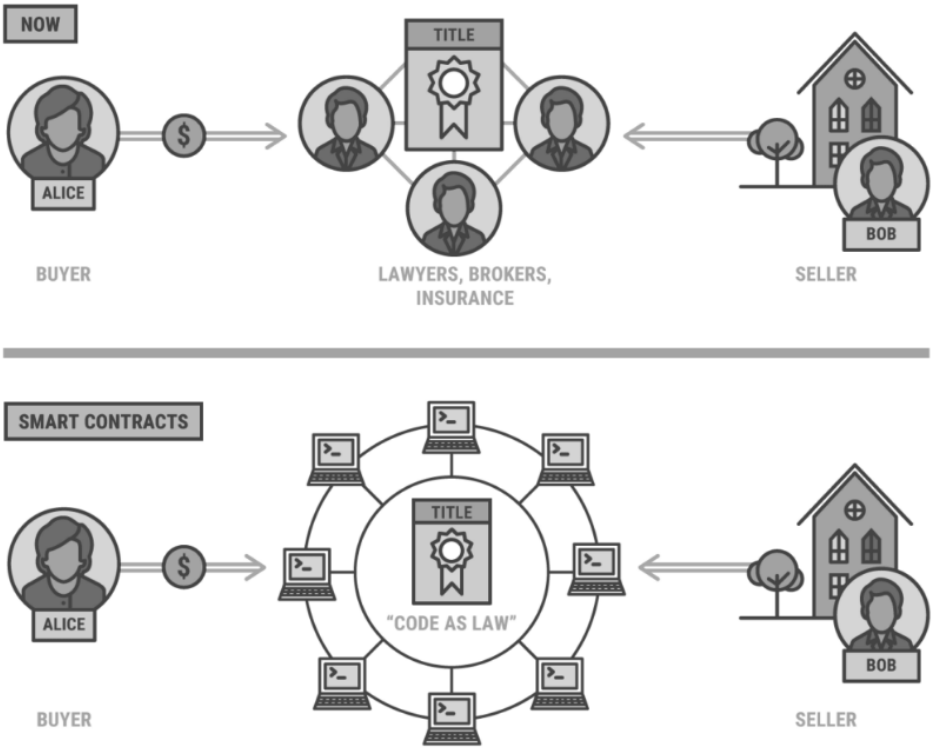


Figure 9 Proof of Stake based Ethereum Transaction using smart contracts [12]

Every PoS based Ethereum account requires to hold a balance, storage, and code-space for calling other accounts. A transaction is said to be valid only when the transferring account holds enough balance to pay for the transaction. If the receiving account contains any code and runs the code, it changes anything from the internal storage to creating additional messages, which may have the following effects on the debits and credits. Due to this effect, every newly created block will influence all the accounts in the network. [9]

2.3.2.1 Advantages of Account-Based Model

The transaction in the account-based model requires only one signature and reference, which produces one output contrary to the UTXO model. This feature saves a lot of storage space, which is much required in a sophisticated cryptocurrency platform like Ethereum based upon the Proof of Stake consensus. A very high degree of fungibility is exercised, the users in PoS based Ethereum make the transactions via Client Remote Procedure Calls, which makes the internal tracking of transactions across the Ethereum ledger difficult. Transaction sizes are smaller as they only include the existing state and not the resulting state of the output. Specific advantages of account-based models include simplicity, saving larger storage space, and the statefulness helps the resulting transactions to be influenced by oracle and other logics.

2.3.2.2 Disadvantages of Account-Based Models

The account-based models encourage address reuse, which is a drawback to privacy, as the account itself links the transactions to the single owner; this makes it more accessible. Proper care must be taken when executing transactions in parallel, specifically transactions belonging to the same account should be performed one after the other as the result of a transaction depends on the input state of the transaction. The account-based model has severe limitations towards the scalability over the platform where it is implemented. If steps are taken to eradicate the scalability issue by building the account-based model with logic, then potential implication on other design concepts in the platform arises.

To sum it up, the transaction models used in cryptocurrency platforms employ cryptography to verify ownership of tokens across the network. The UTXO model is well suited for consensus like proof Work and cryptocurrency applications like Bitcoin based on PoW, and for complex applications like Ethereum based on PoS consensus, account-based models are well suited. Subsequent iteration is done, which tweaks and optimizes the platforms to be more adapted for future developments.

2.4 Communication Model

The communication model influences and makes the process of achieving consensus difficult in a blockchain network. When designing a network, proper groundwork about the communication channel and potentially faulty transmission lines must be checked and rectified. The network model serves as a means of communication between the nodes in a blockchain network. They can influence the transaction rates and other necessary attributes of the blockchain network. The type of communication in a blockchain consensus network is mostly peer-to-peer communication, which is facilitated by the Gossip Protocol.

The type of consensuses and their function depends upon the types of network models a blockchain network has to accommodate. Some protocols are designed to be unreliable and cause arbitrary delays by just dropping the message and while another set of applications involves reliability and could carry time-sensitive data-real-time transactions. In the blockchain, this is called operating under differing assumptions of synchrony [69].

For the of transactions, blocks, digital signatures that need to be transmitted from one node to other and then to the central distributed ledger, the network model or the communication model must be designed keeping all these criteria in mind to avoid delays or lose in data which could inflict a substantial loss on users and eventually lose trust in the system.

2.4.1 Types of Communication Models

2.4.1.1 Synchronous communication model

In the synchronous type of communication, the network knows there is an upper bound on message delays, which means any message exchanged between the nodes must reach the destination in a predetermined time. Every node in the network is aware of the predetermined time, so when a node transfers a data to another, it knows the time it requires for the content to reach the destination.

In synchronous communication, the protocols take discrete rounds. All the nodes are aware of the amount of time for a message to be delivered. Let us assume that all nodes synchronize to send data at the same time T, and then they wait T+1 seconds (1 is the time taken fo the message to be delivered) at this point; all the data would be delivered to the intended destination. The discrete round composes one round at T and then when the nodes receive the new information, which is T+10 seconds, and if required, the nodes again send out new messages at T+20 seconds. This process of discrete rounds continues. Setting up a synchronous communication model to reach consensus is ideally smooth. In reality, a network encounters various hurdles like message delays, data losses, and more. Some findings suggest that due to the unreliable nature of the network, the synchronous model is not suited, especially for cryptocurrencies, considering the network is vast, and the transfer of data with massive computational power causes network congestion [69].

2.4.1.2 Semi-Synchronous communication Model

The semi-synchronous network is similar to the synchronous network except for, in a synchronous network, the nodes or the users are aware of the predetermined time delay when a data is sent through the network. In a semi-synchronous communication, the propagation delay is linked to a random value; this probability distribution is known to the nodes in the network.

2.4.1.3 Partially Synchronous Communication Model

In partially synchronous communication, the nodes are not informed of the predetermined time that takes to transfer data. The time could range anywhere from 1 second to two years, but the nodes know the message is guaranteed to reach the preferred destination. In this communication model, the consensus tends to be more robust, and it represents a typical internet model.

2.4.1.4 Asynchronous Communication Model

The asynchronous communication model is the most challenging communication setting in which achieving consensus is a challenge. There is no time delay set in this model. The messages may take an infinite time to reach the destination. The asynchronous communication model may seem like it can never reach a consensus if the data does not get delivered in a finite amount of time. Nevertheless, one feature in this model that helps reach consensus is that the network allows nodes to arbitrarily drop and rejoin the network [69].

Timing Model	Implications
Synchrony	There exists a known upper bound on the delay of messages.
Semi Synchrony	Assumes there is a known probability distribution expressing the message delay.
Partial Synchrony	Assumes there is either 1. An unknown upper bound on the delay of messages 2. An unknown global stabilization time after which the protocol continues in synchrony
Asynchrony	There is no upper bound on the delay of messages.

2.5 Energy Consumption

The energy requirement of blockchain, in general [13], is significantly high due to transactions. There is limited data about the sources of this energy consumption. Eventually, the carbon footprint of blockchain is more, the reason being the roots of blockchain are associated with cryptocurrency Bitcoin, which involves attempts to validate transactions with a decentralized data protocol. The process required to validate the transactions consumes a significant amount of electricity, which turns into carbon emission. Although one thing to remember is not all blockchain protocol and consensus mechanisms are energy intensive. Among the consensus, proof of work is the most criticized one due to its unchecked energy consumption, according to survey the digiconomist suggests that for a single bitcoin transaction the energy consumed is 800 kilowatt-hours (kWh) of electricity. By comparison, the average household in the United States consumes 900 kilowatt-hours (kWh) of electricity typically in a month. Strategies to mitigate the overconsumption of energy are in existence, and further developments are in progress. Various consensus mechanisms have proposed ways to circumvent this issue. Companies need to keep an eye on this excessive energy consumption and carbon emission for a sustained future [14] [15]

2.6 Tolerated Power of Adversary

The tolerated power of the adversary can measure the level of security in the consensus algorithms. Some research found that a significant amount of power lies in the adversary to attack a blockchain network security infrastructure. According to the findings, the

highest adversary power recorded was 51%. In order to tolerate the attack inflicted by the opponent, the consensuses like Proof of Work have a tolerated power less than 25% and the Practical Byzantine Fault Tolerance, whose tolerated power was less than 33.3%. These results indicate a feeble tolerated power against the adversary. The designing of consensus algorithms needs to be more robust to be able to resist security breaches, and that can increase the tolerated power in the consensuses [8].

In order to analyze the adversary in detail, we shall consider a message-passing model. In a message-passing model, all the nodes communicate with each other by exchanging messages. There are two different types of assumptions on the capabilities of an adversary. They are

(i) Behavior Assumptions

(ii) Synchrony Assumptions

(i) Behavior assumptions - The behavior assumptions exhibit the amount of control the adversaries can exert their control over the behavior nodes. In a distributed environment, these assumptions are called fault-threshold assumptions. The replicas follow the protocol, and replicas are left uninterrupted by the adversaries.

(ii) synchrony assumptions- The synchrony assumptions set how much the adversary has control over the speed of computation nodes, message transmission delay, the performance guarantee of a network. A deep understanding of the capabilities of an adversary is essential. The Byzantine adversary generally has no restrictions, but it is restrictive in terms of the solutions that it allows. The cryptographic assumptions prevent the adversaries from inverting the hash functions or inverting a valid signature without the knowledge of the corresponding private key [16].

There are some common assumptions that the Byzantine adversary cannot invert the cryptographic primitives. One other assumption about the Byzantine adversary is, it does not interfere with nodes which it cannot control directly. A typical assumption is that data sent by the correct nodes eventually reach the destination; the adversary cannot stop the correct nodes from communicating with each other indefinitely by denial of service attacks, partition the network, or unremitting dropped messages [16].

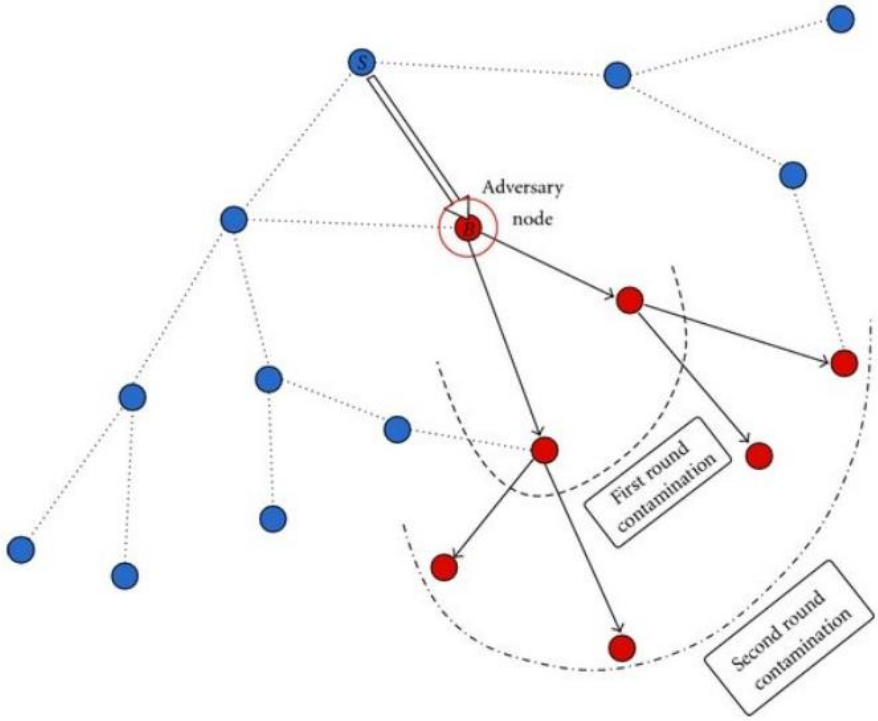


Figure 10 Error Propagation [17]

There are two types of protocols designed to maintain the correctness of the blockchain.

(i) A indulgent protocol - This protocol concentrates more on the safety and fewer restrictions implemented on the adversaries. It is indulgent to asynchrony.

(ii) B indulgent protocol - This protocol is indulgent toward the malicious node behavior. B indulgent protocol focusses more on the safety aspect while enduring a high number of malicious nodes [16].

Algorithm	Node Identity	Energy Saving	Tolerated power of adversary	Data model	Language	Execution	Application	Example
CAP [7]	Private	No	50% Rate should be	Key-Value	SQL, Python, Java, Go	SQL, No SQL	Database, Big Data, Storage	MongoDB, Cassandra, MS Azure Storage
BGP [8,9]	Private	Yes	33.3% Fail	Key-Value	Any	N/A	General Applications	General
PBFT [10]	Private	Yes	33.3% Faulty Replicas	Key-Value	Golang, Java	Dockers	General Applications	Hyperledger
PoW [11]	Public	No	25% Computing Power	Transaction-Based, Account-Based	Golang, C++, Solidity, Serpent, LLL	Native, EVM	Crypto-currency, General Applications	Bitcoin, Litecoin, ZCash, Ethereum
PoS [12,13]	Public	Partial	50% Stake	Account-Based	Michaleson	Native	Michaleson Applications	Peercoin, Tezos, Tendermint

2.7 Transaction Fees

The transaction fees are charged on the users for facilitating cryptocurrency transactions. The transaction fee is the primary tool based on which the speed of the transaction depends. If the transaction fee is low, then the priority of the transaction is also low. [18]

2.8 Block Reward And Properties

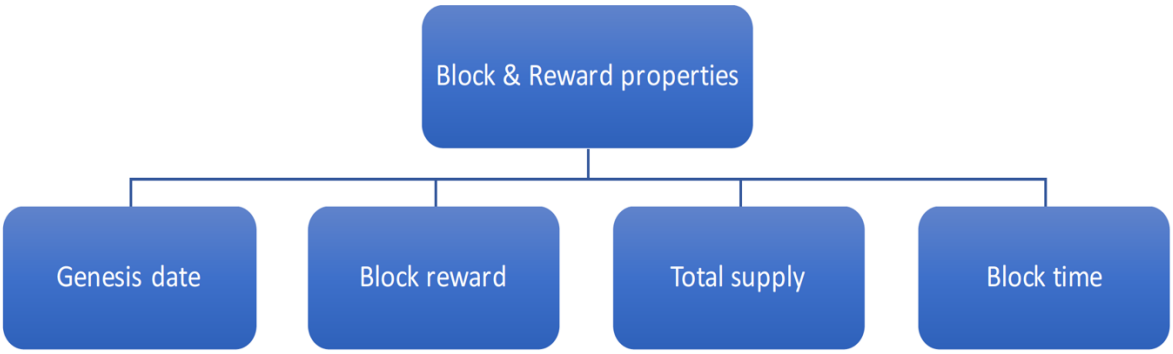


Figure 11 Taxonomy of Block & Reward Properties [19]

The properties that we can see in Figure 11 like Genesis Date, Block reward, Total supply, and Block Time are quantitative metrics that are used to differentiate different cryptocurrencies. However, these properties do not influence how a consensus should be designed or operate, but they have either a direct or indirect impact on how a consensus is achieved on cryptocurrencies.

2.8.1 Genesis Date - This type of reward represents the timestamp of the first block created in that cryptocurrency.

2.8.2 Block Reward - In a cryptocurrency, a node is being rewarded for creating a new block.

2.8.3 Total Supply - It is the total amount of cryptocurrency mined so far. The total supply can be higher or equal to the ongoing supply.

2.8.4 Block Time - This property represents the average time to create a block in cryptocurrency.

Reward Process - The reward process in each consensus differs. Some consensus offer reward for creating new valid blocks, consensus incentivize the stakeholders to participate in the minting process.

2.9 Communication Complexity

One way of calculating the efficiency of any consensus algorithm can be attributed to its network usage. Latency and communication complexity will provide an idea about the trade-offs of these consensus algorithms when compared to each other.

Our assumption based on the surveyed literature is that almost every consensus algorithm implements some form of Atomic Broadcast i.e., an Atomic Broadcast permits the processes to pass messages reliably to the entire network. This concept will enable us to utilize Atomic Broadcast as an abstraction layer within the consensus algorithms, which helps us to compare the algorithms based on standard metrics and parameters.

By detecting some parameters in each algorithm, we can consistently analyze the total number of bits transacted via the network (communication complexity) and the duration (latency) it takes for all the participating nodes to receive it. This analysis will help us to evaluate the consensus algorithm theoretically and design future experiments for practical proof. Also, one will be able to decide the chosen consensus algorithm is suited for a permission or permissionless application using this method.

Asymptotic analysis is the method that will be used to evaluate the consensus algorithm. The performance of the algorithm is measured in terms of input size and not based on the run time of the algorithm. This approach will provide the time taken by the algorithm with an increase in the input size.

There are three asymptotic notations one has to familiarize before analyzing the algorithms.

i. Θ (n) is the precise asymptotic behaviour of the algorithm.
ii. O (n) is the upper bound behaviour of the algorithm.
iii. Ω (n) is the lower bound behaviour of the algorithm

Out of the 3 described above, Ω(n) is the least used because the best-case performance of the algorithm is not always useful in making a recommendation for a particular algorithm to be used in an application.

Now, let us define the rules before analyzing the algorithms. An Atomic Broadcast or Total Order Broadcast is a distributed systems algorithm that should implement two operations: A-broadcast (*msg*) and A-deliver (*msg*). To qualify as an Atomic Broadcast protocol, it has to satisfy the following properties.

- Validity: If a valid process A-broadcast a message *msg*, then some valid process subsequently A-delivers *msg*.

- Agreement: If a valid process A-delivers a message *msg*, then all valid processes subsequently A-delivers *msg*.

- Integrity: If p is a valid process, then for all A-delivery of *msg* by p, *msg* has been A-broadcast.

- Atomicity: If two valid processes A-deliver two messages *msg*1 and *msg*2, then both the processes A-delivers the messages in the same order.

The protocols don't solve precisely the same problem because they don't consider the fundamental network model to be the same. Thus, to have a consistent analysis, we ought to consider them all as Atomic Broadcast Protocol. We consider that they implement the process A-Broadcast (*msg*) and communicate messages delivery by calling A-Deliver (*msg*). Below sections will describe how these processes gets mapped to each protocol for solving the Atomic Broadcast conditions and enunciate the model variations. [20]

2.9.1 Network Model

The algorithm analysis is based on Interactive Turing Machines. Any assumptions on the network itself are conveyed through extra parameters to that protocol during instantiation. In general, the following are the parameters.

Δ bound on network delays
n number of participants
α byzantine power
κ security parameter

The maximum number of time steps taken by a message to reach its endpoint is shown as Δ. Synchronous protocols usually require Δ, a bound. For the asynchronous algorithms, Δ is the rounds in which messages are given a round number r, where all r-1 messages should have been delivered before sending new r+1 message.

Now, let the algorithms assume $|\Pi| = n$ - number of nodes that have asynchronous clocks, which transacts messages reliably between each node. The term asynchronous refers that there are no limits on the number of time steps taken for a message to be transacted and reliably means that neither messages are lost or changed during transaction. The communication on these protocols is bidirectional i.e., all participating nodes can send or receive messages.

All the distributed systems algorithms in question will tolerate reasonable levels of byzantine failures i.e., when nodes behave randomly. We define a node is honest when it follows the protocol rules, otherwise it can be tagged as a faulty node or malicious node. In order to achieve this, the algorithms need a sybil tolerant quantity that cannot be artificially increased by the adversary. We define α is that parameter, that denotes the fractional byzantine power owned by the opponent.

For e.g. if there are maximum of Q byzantine nodes in the set Π, then $\alpha = \frac{Q}{n}$

For a Proof of Work algorithm, α will be a fraction of the total hash rate the byzantine node can create. Alternatively, if it was a proof of stake-based algorithm, then α will be the fraction of sum of all stakes owned by the byzantine nodes.

All of the algorithms need a parameter to represent the security level, which is defined by 'κ', the security parameter. When the security parameter κ is varying and model parameters are constant, the O(n) notations only refers to κ. This outcome is not favorable for our analysis because we want to show the performance variation when the model parameters are changing, especially 'n' when the number of nodes is changing. Thus, O(n) cannot be directly used for metrics but it will be denoted on the subscript on the notation to show which of the model parameters it is dependent on. For instance, this approach still hides the bound expressions of the model parameters $\alpha = O_{\alpha}(1) = O(1)$ but practically we will use the subscript and avoid simplifying the notation.

In addition to the aforementioned model parameters, there will be an extra parameter specific to each protocol which will be pertinent for our analysis. Let us name the parameter q which could be used for making some tradeoffs between efficiency and security.

To summarize, latency and communication complexity are very pertinent metrics as it affects the user experience directly. Communication complexity is the number of bits received by honest nodes for transmitting A-delivered by all honest nodes. Typically, communication complexity is provided in terms of overhead, but we consider the total value, to show the network capacity required to execute the algorithm. This metric is shown as a function of the overall message size b.

2.10 Verification Speed

The verification speed is the amount of time required in seconds to validate a transaction. The block creation speed is affected by the verification speed. The speed of verification depends upon the blockchain network and computation power acquired in that blockchain network [19].

2.11 Throughput

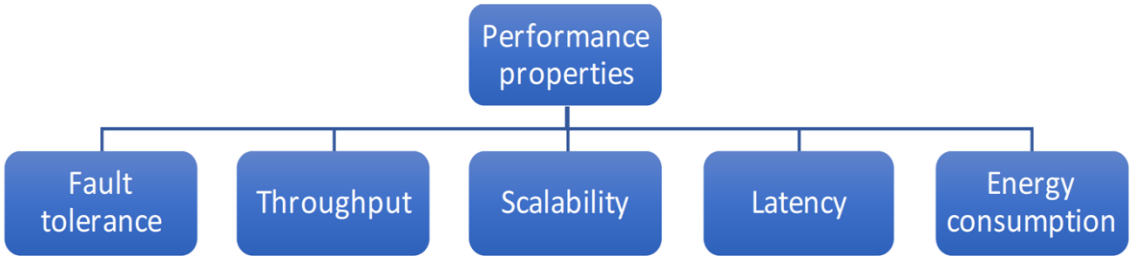


Figure 12 Performance Properties [19]

The throughput is a performance analysis metric, which is represented as the number of transactions achieved per second or valid blocks added to the blockchain network. The throughput of a blockchain network depends upon the consensus algorithm, which is used in their platform. The throughput is calculated on various factors like verifying the transactions by the miners before they are entered into the distributed ledger, maintaining a uniformity of the transaction database in the shared ledger to avoid double spending and other malicious attacks.

The factors which may affect the throughput in a blockchain are, design of the network, size of the data, and the scope of the consensus. The design limits the throughput of the network, as the network design may be for either critical or non-critical applications. If throughput is the goal, then moving only limited data should be the focus [21].

2.12 Scalability

The scalability can be achieved by increasing the number of nodes in order to achieve a maximum number of transactions. Scalability cannot be achieved with every consensus, like Proof of Work PoW, the implicit consensus is not scalable, but proof of trust is a scalable consensus algorithm. In general, scalability in the blockchain is affected by endogenous factors like block time interval and block size. If the interval time is reduced, the performance might improve, but it may lead to blockchain forking - when two nodes do not agree to standard rules. Blockchain forking, in turn, leads to security vulnerabilities. The block size is also another contributing factor to reduce scalability; with bigger block size, the transactions are made faster by compromising the network speed, which again causes poor security features. Intuitively modifying the interval time and block size causes the system to be more susceptible to attacks like selfish mining and double spending [2].

The blockchain is typically organized in different layers, irrespective of the type of blockchain, permissioned, or permission-less type. Layer 1 is responsible for the formation of the blocks-blockchain. For this to occur, important communication must take place, which exclusively happens in the network layer, which makes communication happen through the internet. The hardware layer is responsible for implicitly computing

the received values and producing the corresponding outputs. The figure [2] represents the layers involved in blockchain and the formation of blocks at layer 1.

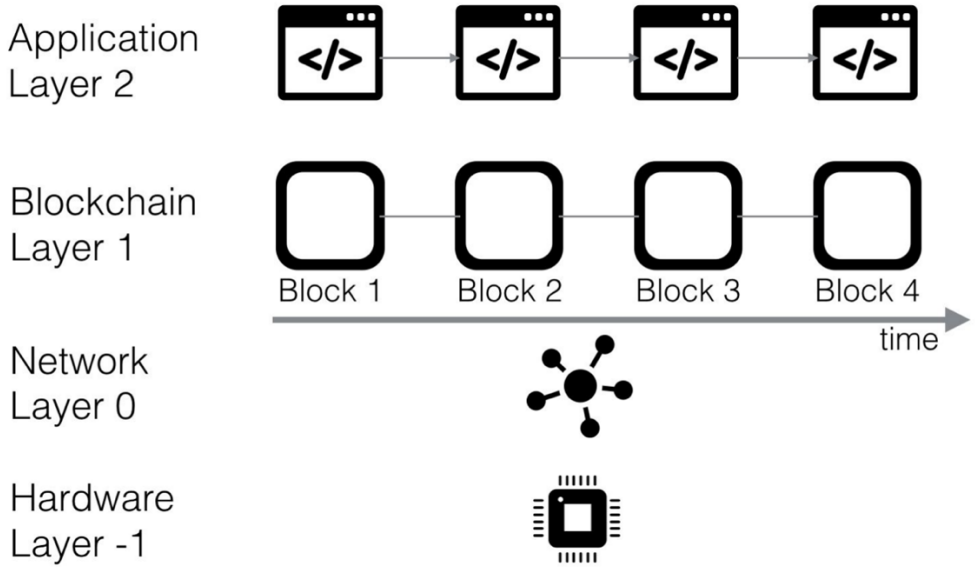


Figure 2: The different layers that a blockchain is constructed with. Note that layer 2 could also be another blockchain, for example an application specific blockchain.

Figure 13 Blockchain Layers [2]

How do we make the blockchain more scalable? Layer two, called as the application layer, is responsible for the financial transactions and introduces methods to make the layer one more scalable. Here are some of the options to improve scalability in these layers; layer 0 can be made more scalable by introducing a more accelerated network propagation in the mechanism to achieve the desired throughput. At layer 1, blockchain structures such as directed acyclic graphs (DAG), can be implemented, which comes with additional complexities. Now layer 2, off-chain, also called hubs or side chains, can be performed. The off chain does not require any additional consensus algorithm, but the side chain by itself is an independent chain governed by a consortium. One thing to remember is if layer one is slow does not mean layer two also has to function slowly necessarily, layer two by itself is a standalone blockchain.

2.12.1 Blockchain Forks

The fork is a phenomenon when blockchain gets split into different versions. The divergent blockchains facilitate security breaches by aiding few features for a forking attack. There are two different types of forks (i) Hard fork and (ii) Soft fork. The hard fork is a rule that ensures all the nodes adapt and follow the change. The soft fork, on the other hand, is used for backward compatibility - that is, when only half of the nodes were able to adapt to the hard fork rule, the remaining half of the nodes can go back to its original state by dispersing themselves into separate chains. One such example of this is in 2016 when Ethereum hard forked into Ethereum and Ethereum classic. Forks are

sometimes introduced intentionally to make upgradations in the protocol, fix bugs, and used as a performance enhancement. Proof of work PoW Fork can occur accidentally when two or more miners find a different block, fork during these situations can be resolved when subsequent blocks are mined, and chains with maximum proof of work are determined [22].

2.13 Sybil Attack in Blockchain

The Sybil attack happens at nodes where the malicious activities take place. The nodes where virtual identities are created is known as the Sybil nodes. In order to have the network power, the adversary tries to create several duplicate or virtual identities. The attacker eliminates all the genuine nodes from the whole network so that it would be easier to attack the system.

Moreover, the attacker has the ability to construct many ID's in the blockchain technology. A network consists of a mining pool where miners join the network. In the mining pool, the rewards/incentive are shared among the network miners. Even the pool operator may act as a malicious operator. In this attack, if the attacker creates several virtual identities in the mining process without participating in the actual process will lead to data dissemination. Because of the more data utilized, the mining process of the genuine miner is inevitably stopped/blocked. Ultimately only the attacker becomes the block creator, and that block is connected to the chain network. This attack results in the loss of rewards/incentives for genuine miners. The system transaction per second(throughput) is reduced. [2]

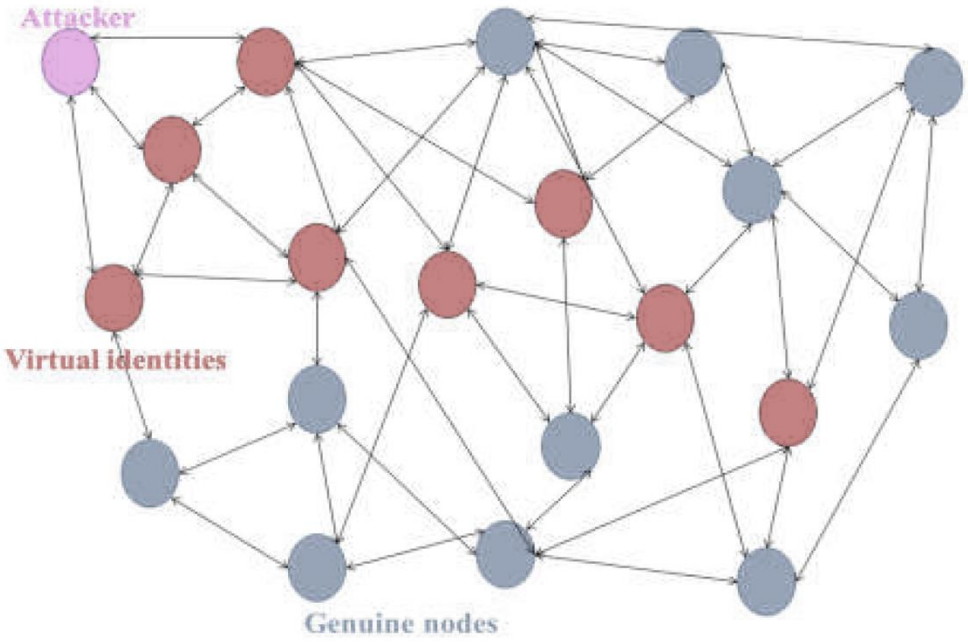


Figure 14 Sybil attack scenario [2]

For instance, if the genuine and the attacker each create one block and now according to the blockchain consensus algorithm, each miner should broadcast/display their transaction to the neighbouring nodes; this process continues for the whole network. The attacker blocks all the genuine miner blocks. The attacker also creates several virtual identities throughout the network, which results in only the attacker's block moving forward in the network, blocking the progress of the genuine miner's block. Another way of the genuine miner's block broadcasting its transaction in the network is through a random peer-peer network structure. However, this is a much slower approach and reduces the propagation of the genuine miner's block as there are many duplicate identities throughout the network. The genuine miner encounters several drawbacks during this process, such as energy wastage as much computational power is used during a transaction, incentives/rewards are not received with the reduction in the throughput of the overall system. The majority bar becomes higher as a result of the attack, which also increases the energy consumption of the network. The constraint in the Public model of the blockchain is that it is difficult to detect the Sybil nodes. On the other hand, it is effortless to identify the Sybil node and prevent this attack from happening.

The Sybil attack is possible only when the attacker has sizeable computational power. The traditional methods that are used to identify the Sybil attack are radio resource testing and Registration based method. In the Registration based method, the node consists of a list known as the known-good list from which the identities of the mining process can be validated/verified. Nevertheless, if the attacker adds the identities to that list, it would result in malicious activities again. Hence a known identity list is essential to prevent the Sybil attack.
In the radio resource testing mechanism, the attack is prevented by verifying the neighbour node. Each node is allocated with a channel. It can be verified that the neighbouring node is not under attack when the messages are received adequately in all the nodes of the network by hearing it on the channel. The general assumption in this testing mechanism is that only one radio is not able to transmit and receive in the network. If there is a shortage in the number of channels, then only some node subset are tested. A distributed network is required to reduce the risk in the network.

2.14 The 51% Attack

The blockchain is a Decentralized and Distributed ledger system. It prevents any single or centralized entity from taking control of the network for their private use. The mining power in a Proof of Work-based system involves investing and owning computational resources. The power of a miner lies in the computational assets owned by them. This is referred to as the hash rate r hash power. The miners are distributed around the globe, and they compete with each other to be elected as the 'next' to find a valid block hash and be rightfully rewarded with newly generated Bitcoins [23].

About the above context, the hash rate, in general, tends to be distributed over different nodes across the world, which means the hash rate is also widespread and cannot be confined to a single node. Sometimes the hash rate, if not well distributed, a single entity

or an organization holds the majority of the share, i.e., at least 50% of the share. One possible consequence of an uneven distribution of hash rate can result in the 51% attack or also called the majority attack [23].

2.14.1 Analysis of 51% Attack

The 51% attack is one of the famous attack methods in the blockchain when a single or group of miners control more than 50% of the hash rate of a particular blockchain network, which prevents any new transactions from getting confirmation by pausing the communication between the buyer and seller. The attackers can complete the Proof of Work much quicker than the legitimate miners. Hence their transaction will be connected to the longest chain. If the number of hast rates controlled by the attackers goes high, then they can accomplish their task of undertaking the blockchain network much sooner [23]. The Proof of Work designed the probability of finding blocks based on the work done by the miners. Eventually, people mine more blocks at the same time and create a mining pool- A group of miners who have very high computational power. When a single entity or an organization gains 51% of computational power, they put that into use by finding the blocks faster than the miners with relatively less computational power. This gives the majority holder the permissibility of the blocks, which ultimately allows them to modify the transaction data [24].

The 51% attack is also capable of reversing the transactions, which means they can circulate the same coins to be spent multiple times. This can occur if the attacker has gained control of more than 50% of the nodes in a blockchain network. Satoshi Nakomoto calculated the probability of the attack on the Proof of Work-based Bitcoin on varied computing powers. The probability calculation is represented in binomially. This binomial function represents the rate at which the attacker can catch up with the honest chain is shown as [25].

$$q_z = \begin{cases} 1 & p \leq q \\ (q/p)^z & p > q \end{cases}$$

The p represents the probability that the honest nodes discover the blocks, q represents the attacker tracks the new block before the honest miners and qz is the probability that attackers catch up with the honest miners from z blocks behind. If the attacker's power is higher than the honest miners, then the probability of an attacker catching up with honest miners will be successful. The probability of this occurring depends upon various factors like a change in the network's mining power, the design of Proof of Work [25].

2.14.2 51% Attack Strategy

The blockchain being a decentralized platform, anyone including the attacker, can be a part of the network and maintain it. The attackers aim at reversing the transaction by spending the same Bitcoins more than once. The honest miners always are not aware of

the content of the block, but an attacker is very clear of the data in a block, whether it is real or illusionary. The attackers bring in a new illusionary block into the blockchain network to speed up the process [25].

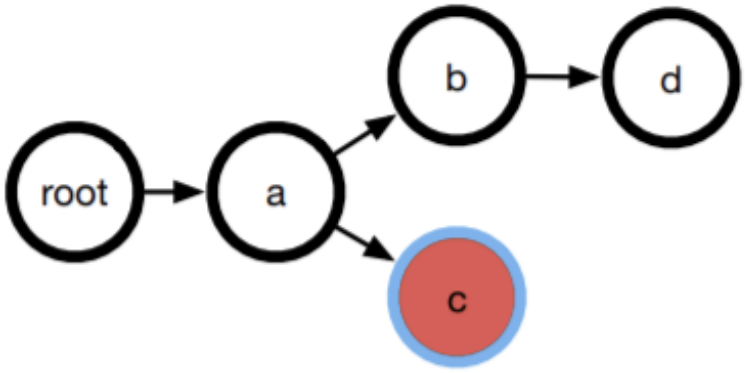


Figure 15 Attackers Strategy [25]

The above figure shows the perspective of blockchain from an attacker's point of view. The red node 'c' contains blocks with unreal data or illusionary data in them. The blue border around the node 'c' shows that this node will be selected. The other nodes 'a,' 'b,' 'd' are legal nodes with actual data in blocks. When a new node arrives into the network, it chooses the node 'c.' The attackers intend to connect more nodes to the node 'c.' By making new connections to the red node, i.e., 'c,' the chains become longer - the longer the chain, the safer it is for the attacker to make transactions. The more blocks connected to a block, the safer the block is. According to some research, when a block is connected with six other blocks, the data in the block cannot be changed. When this illusionary block is connected to six other blocks, it means the attack is successfully accomplished. The attacker connects a new node to the longest chain to increase the probability of attacking [25].

2.14.3 Chances of 51% attack

The blockchain is a secure network as a distributed node maintains it. All the nodes in the blockchain network need to cooperate in reaching the consensus. Bigger the blockchain network, less are the chances of an attack due to the challenges and complications involved.

In Proof of Work, when a miner has a higher hash rate, then the chances of finding a new block are more. The Proof of Work network involves a multitude of hashing, making a robust network with massive computational power produces more trials per second. The

applications build over Proof of Work consensus have fewer chances of being attacked due to the magnitude of the network. With the vast network, it is a challenge for a single entity or group to formulate a plan to attack the network by overwhelming the honest participants.

Attempts to make changes in an already confirmed block is difficult as the chain grows as the blocks are linked through cryptographic proofs. To alter data and revert the transactions in the confirmed blocks, the cost involved is more. Even if the attacker is prepared to invest in making such high cost involved changes to destroy the blockchain network, the Proof of Work based applications can modify and adapt to respond to the attack. This requires other nodes to reach consensus and agree on the changes. PoW based applications are very resilient to these types of attacks; on the other hand, blockchain networks with smaller cryptocurrency networks with low hash rate and relatively low computational power are susceptible to 51 % attacks [23]

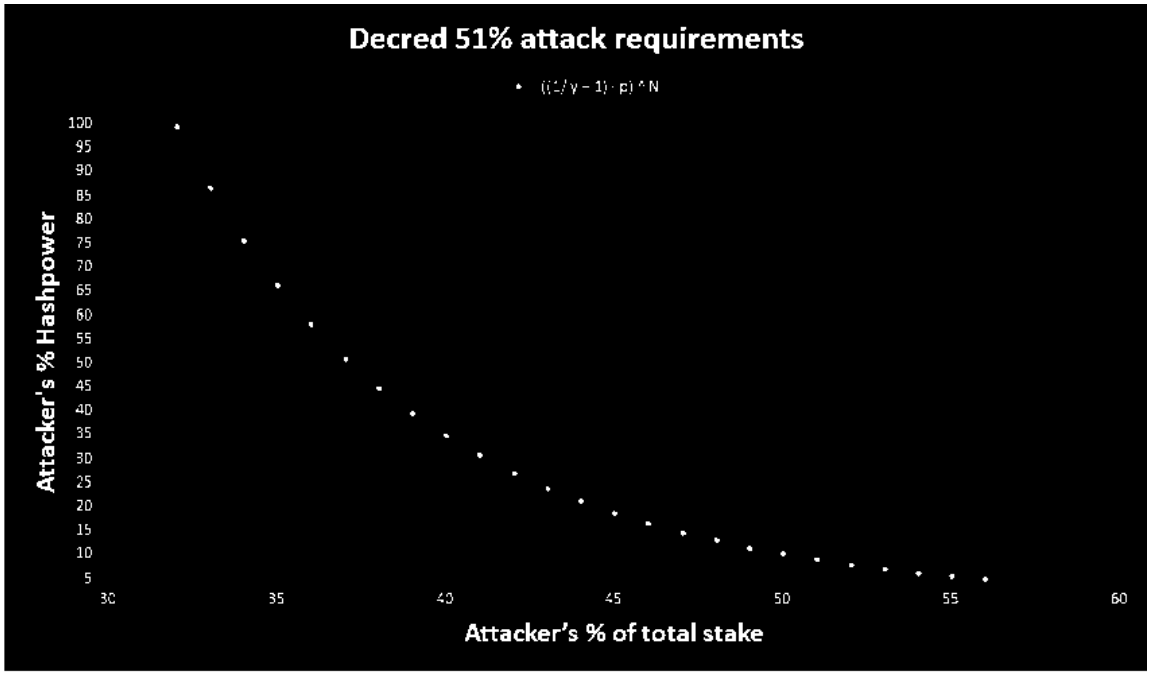


Figure 16 Representation of percentage of Hash rate and Total stake to start 51% attack [26]

2.15 Double-Spending Attack

The blockchain aided cryptocurrencies are gaining popularity and global acceptance these days. The cryptocurrencies have reformed the usage of digital currencies. A progressive transition towards the digital era is already occurring. The digital currencies offer so many benefits, but along with that comes the drawback and insecurities caused by attacks like the double-spending attack [27]. The double-spending attack is a potential flaw in the digital currencies when the single same digital token gets spent more than once. The

digital file which comes along with every digital token is duplicated for a double-spending attack to occur [27].

In a digital transaction, if the verification system goes missing or if it is not proper, then a double spending can happen.

2.15.1 Causes of Double-Spending Attack

2.15.1.1 The 51 % Attack

The 51% attack leads its way to a double-spending attack. The 51% attack involves a single entity or an organization taking over 50% of the hash rate or computational power of a blockchain network. When the attacker accomplishes the goal by taking over half of the hash rate, then inducing reverse transactions is the subsequent action, which causes the same digital token to be spent more than once. The attacker can halt the verification between a seller and payee. Thus, there is no proof of the digital token being spent, and this results in the double-spending attack. To conduct the 51 % attack is a significant investment on the attacker's side to destroy a blockchain network as some PoW based applications are massive in its architecture [28].

2.15.1.2 The Race Attack

The attackers aim at sending the same digital token swiftly to one or more addresses, and the buyer sometimes does not verify the transaction, in this case, there is a 50% probability that the token received by the merchant will get double-spent [28].

2.15.2 Solving the Double-Spending Attack

The blockchain technology has been novel in various aspects, and the blockchain was able to solve many obstructions that existed in conventional technologies. The consensus mechanism in blockchain aids in handling the security breaches and other problems faced by the cryptocurrency. The blockchain's consensus mechanism is the backbone of cryptocurrency as they come with features like a distributed ledger system and many more, which makes it an inevitable option. Any transaction that takes place in the blockchain network is documented digitally in chronological order and time-stamped for verification and future references. The tracking mechanism is in place, and the nodes verify any transaction that takes place in the platform before it goes into the universal ledger. Every ten minutes, a block, which is a chain of transactions, gets added to the ledger, and all the nodes in the network must have a copy of the block that goes into the ledger [28].

2.15.3 Double-Spending Attack Prevention

Assume a scenario when a person makes a cryptocurrency transaction, but if the person attempts to make the same transaction twice, the blockchain does not allow. This is

because when a transaction is initiated, it goes into a pool of unconfirmed transactions. Moreover, only the first completed transaction gets confirmation and verification from the miners. When the same token is attempted more than once, then the miner can identify that as a double spending, therefore, it does not get any verification or confirmation. Circumstances may arise if both the transactions are attempted at the same time, then the transaction which gets the highest confirmations from the miners will enter the blockchain. The merchant must at least wait for six confirmations after a transaction. It means that when a transaction is added to the blockchain, consecutively, six blocks must be added on top of the transaction. The transaction and the block will be mathematically connected with the previous one. Once the transaction enters the blockchain, it becomes difficult to alter the transaction. The concept of receiving six confirmations from the miners assures the transaction is not double spent [28].

2.15.4 Stealth Mining

When a legitimate miner finds a block, it is supposed to be broadcasted to all miners, so that they can verify it and add them to the blockchain. If a corrupt miner is a part of the network, the corrupt miner can create an offspring of the same blockchain and not broadcast the solutions of the block to other miners. Now two versions of the blockchains exist [29].

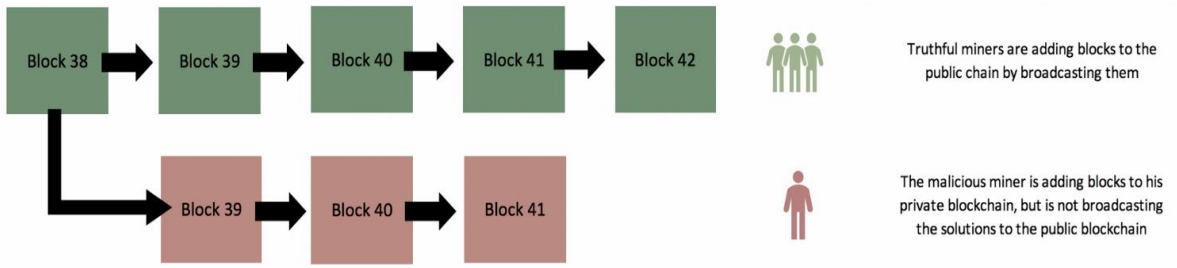


Figure 17 Representation of Legit block and Stealth block [29]

The corrupt miner can spend all his BTC on the original version of blockchain and keeps the offspring(unreal) version of blockchain hidden from the rest of the miners. The corrupt miner does not include any legitimate transactions on the violated version of blockchain; all the BTC in the violated blockchain are preserved [29]

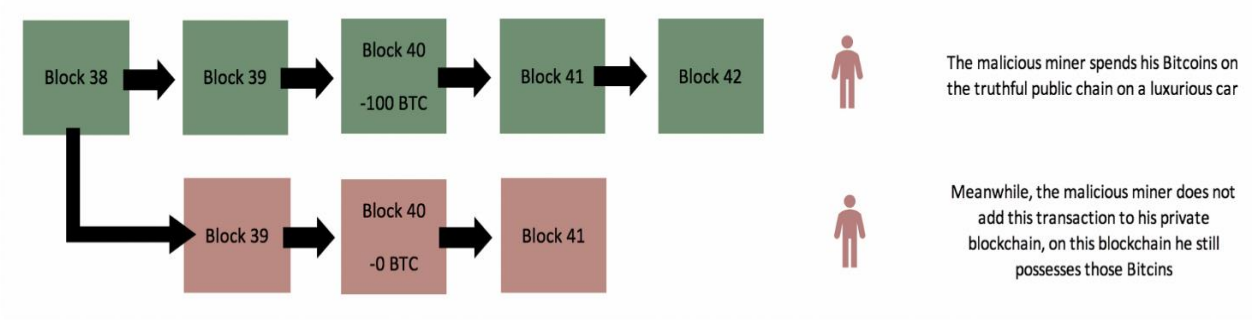


Figure 18 Block in Amber shows the Preserved BTC [29]

In the meantime, the corrupt miner verifies every block and adds to the isolated blockchain. The law of governance of blockchain is the longest chain gets the majority of miners adding their block to it. The longest chain in the network becomes the true version of blockchain as per the governance law [48].

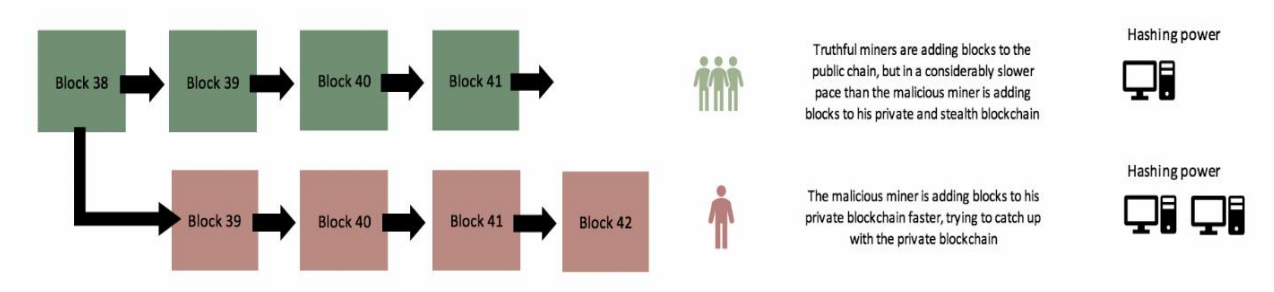


Figure 19 Building of the Stealth Blockchain [29]

The corrupt miner keeps adding the blocks to the chain faster than the legitimate miners adding blocks to the original version of the blockchain. When the isolated blockchain is long enough, it is broadcasted by the corrupt miner. The uncorrupt miners will switch to the unreal version of the chain due to the governing law of the blockchain.

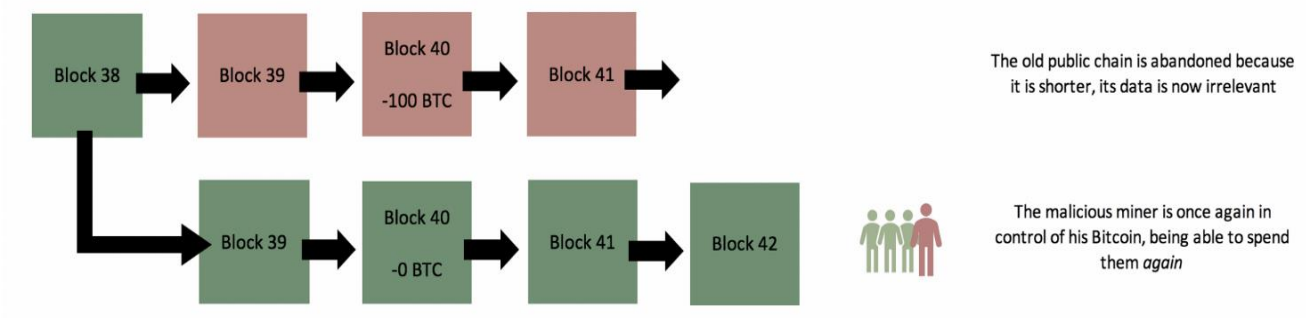


Figure 20 Corrupt Miner in Control of the BTC [29]

This precisely is a double-spending attack, which requires the attacker to own 50% of the hast rate or computational power of the entire network. The 50% computational power gives the ability to reverse the transactions, eventually leading to a double-spending attack [29].

3 DESCRIPTION OF CONSENSUS ALGORITHMS

3.1 Proof of Work:

The Proof of Work (PoW) is the first consensus algorithm to be found in the blockchain technology. Today many platforms use the Proof of Work PoW to validate all their transactions and move the relevant blocks to the network chain; they have also used to solve complex mathematical problems quickly. The distributed ledger stores all the blocks, which are a result of the confirmed transactions. However, individual nodes called the miners must take the utmost precaution when verifying and maintaining transactions, this process is called mining, and the individual who does this is called miners. The mathematical problems involve massive computational power, like the hash functions - where we find the output without knowing the input.

The Proof of Work is to serve as an economic measure to prevent denial of service attack or other security-related misuses. The Proof of Work allows only users who can compute a moderately designed function in order to gain access to the network. The Proof of Work consensus is designed with a prime-number based verification system, and this verification system depends upon the node to trace the number only used once (nonce). The verification system consists of 4 bytes field for every block, and the process of verification is done every 10 minutes to avoid any double-spending attacks or data abuse. A unique block for a set of transactions is created, selecting the hash of the previous block, and the Proof is hashed to produce a new block that meets the network requirements. The hash of the new block created by a node must have leading zero's, which is a requirement by the network. The Proof is initialized to zero. The hash is executed with the Proof, and the hash of the previous block and the transactions are utilized to create the hash of the new block. If the new hash does not have leading zero's, then the Proof is incremented. The process is repeated until the hash value meets the network's requirement, shown in Figure 5.

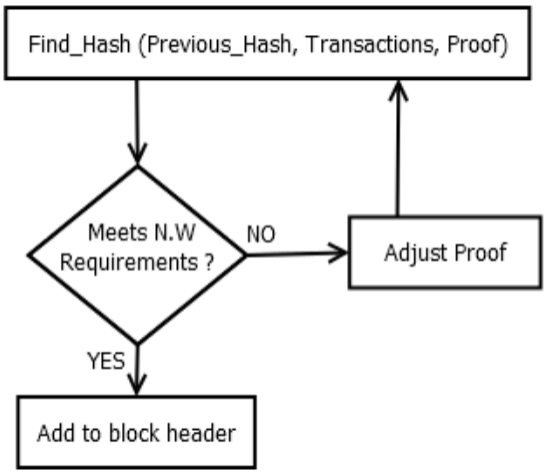


Figure 21 Proof of Work High-Level Flow Chart [30]

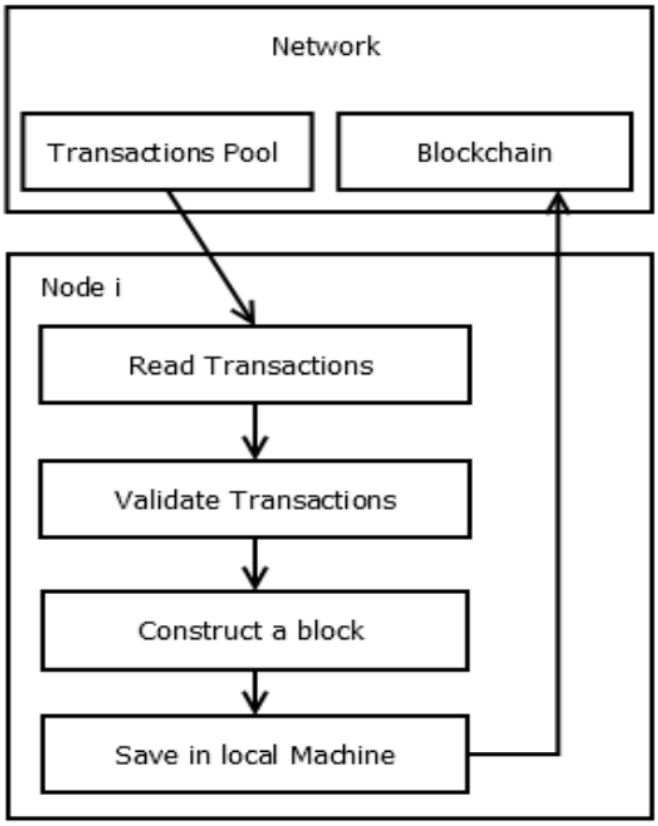


Figure 22 Proof of Work Node Level Flow Chart [30]

3.1.1 Proof of Work in Cryptocurrencies

The Proof of Work consensus is used in the Bitcoin network to avoid the double-spend. The general working of this network, the available Bitcoin nodes run by the participants, are stacked up in the form of a block (information). These blocks should be related to the previous one, and the whole procedure is repeated every ten minutes. The primary issue in the Bitcoin network is that too many miners are in the lookout for the same block. There are many complexities in choosing which block should be assigned to a participant or miners. This issue is resolved by using the Proof of Work consensus algorithm. A puzzle is formed mathematically with a high level of intricacy. The puzzle is a long hash value that needs to be solved by the miners, where the computer basically chooses each number. Typically, the hash value consists of different variables such as random numbers and letters. The puzzle is unique to each block of transaction. The first participant to crack the puzzle for a block is given that block of information to work with, another advantage in doing so is that the miner is provided with Bitcoins as a reward. After every transaction, the difficulty level of the puzzle increases. The more the transaction, the larger the hash

value. The complexity of the puzzle is indirectly connected to the number of participants in the network. The average time required to solve the puzzle would be around ten minutes. Here though the puzzle is challenging to solve, verifying the obtained answers is quite easy.

The process of creating the puzzle is basically finding the hash values of the block. The Bitcoin comprises of the hash values of the block with a set amount of consecutive zero's in them. The level of difficulty can be figured from the number of zero's in the hash values—more the number of the zero's, more the complexity of the puzzle. Though the Bitcoin network can create several blocks, it is not a preferred method. Mainly because creating a large number of blocks would result in slowing down the product and spamming it. Hence, overpopulating the blocks is not recommended in the blockchain network. The Proof of Work is used in creating each block, where a lot of computing work is done. This effort provides value for the product. The procedure of solving the puzzle, with much computational power, is known as mining. Usually, the duration to solve the puzzle is around ten minutes, but as the system's computing power improves with the number of puzzles solved, the time duration decreases. This reduction in time is compensated by increased complexity in every transaction. Hence the time duration does not decrease as transactions continue to happen.

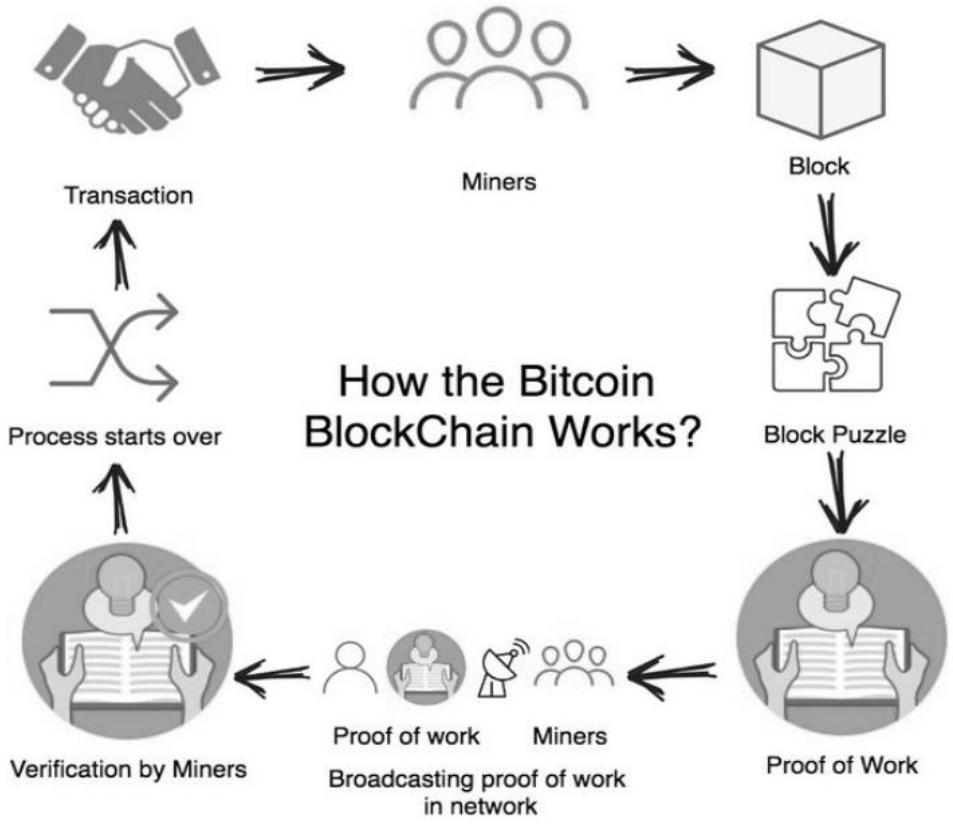


Figure 23 Working of Bitcoin Blockchain [31]

3.1.2 Proof of Work Mining using Hashing

Every miner faces a mathematical challenge when trying to add a block to the blockchain, as the hash output for the data in the block must be determined. The hash value is relatively hard to solve, but the resulting value can be easily verified. The hash function is nothing but a mathematical problem that requires the processing of data from a block through mathematical function, which results in an output with fixed length. The fixed length of the output improves the security of the network if an attacker wants to decrypt the hash value. The attacker may not be able to detect the length of the input based on the length of the output. Solving a hash involves a complex mathematical problem; mainly, it begins with the data available in the header of the block. The block's header contains previous block's hash, the hash of the Merkle Root, target hash, nonce, version, and time stamp. The miner concentrates on the nonce, which is a series of numbers. The nonce gets appended to the already hashed contents of the previous block. Now the total hashed value is verified to see if it is less than or equal to the targeted value; if it satisfies the requirements of the network, then the miner is duly rewarded, and the block is added to the blockchain. The number used only once (nonce) is a random string of numbers. The miner must examine the right string by conducting a trial and error method before determining the string should be used as the nonce. It is unlikely that the miner may come up with the right string on the first attempt itself [30].

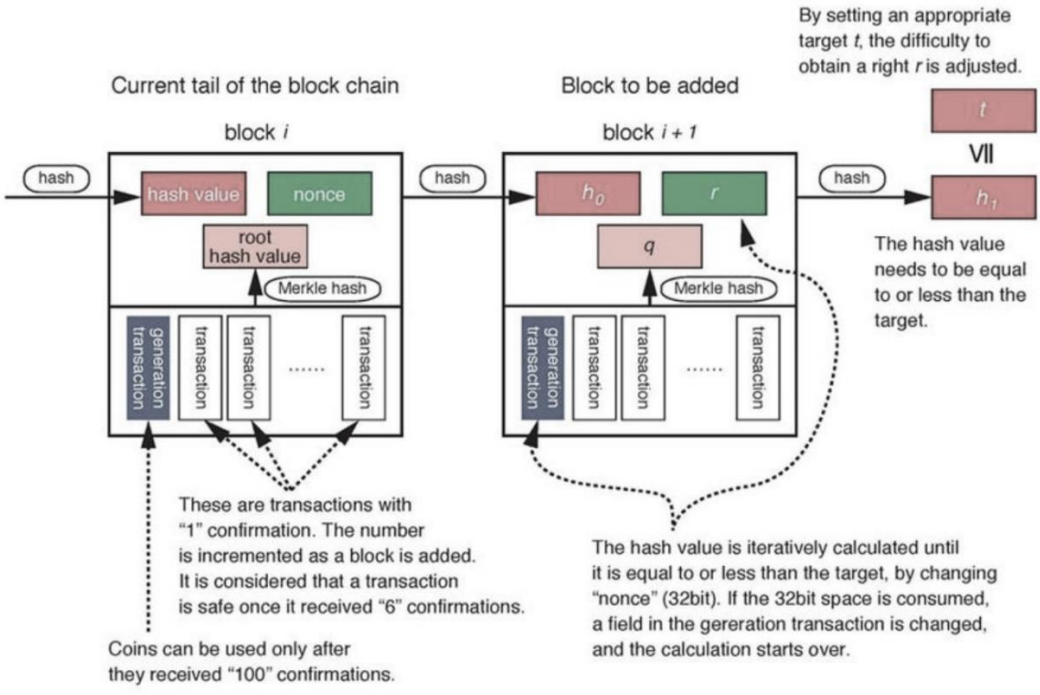


Figure 24 Proof of Work detailed schematic

3.1.3 Components of Proof of Work

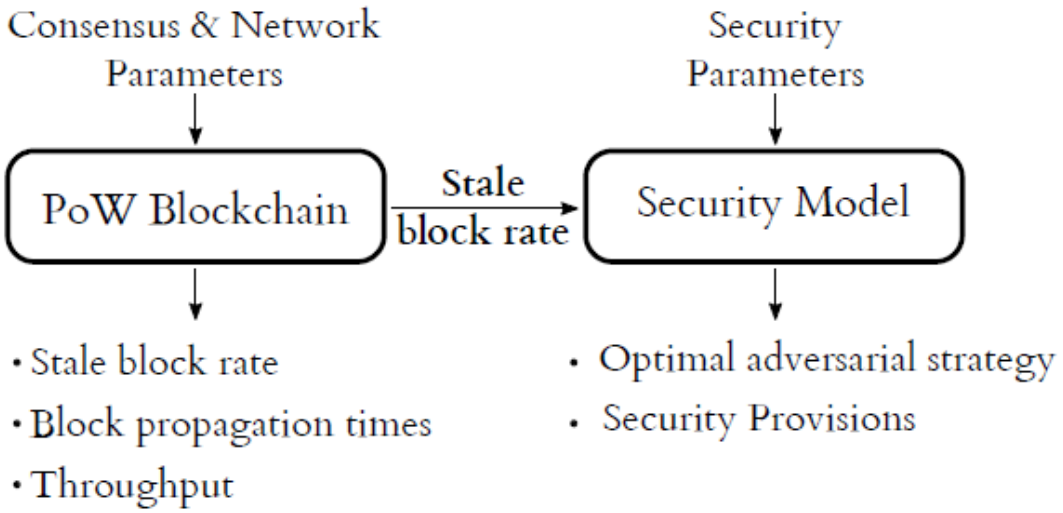


Figure 25 Components of PoW [32]

The above figure [29] contains (i) a blockchain instance and (ii) a security model for studying about the optimal adversaries strategies.

(i) One of the main components in the Proof of Work blockchain is the Proof of Work blockchain network. The Pow blockchain is used with other consensuses and network parameters like block propagation time, block size, average upload, and download speed [55]. The main output of the blockchain instance is the measured stale block rate, which is fed as an input into the security model [32].

(ii) The security model is designed to analyze the double-spending and self-mining attacks, which allows us to understand the optimal strategies implemented by adversaries [32].

Based on the findings with the security model, selfish mining does not implement a rational strategy every time, so we quantify the double-spending resilience of Proof of Work to capture rational adversaries. We objectively compare different Proof of Work blockchains with the required number of transaction confirmations. This finding helps the merchants to know ahead, the number of confirmations required for a specific value of a transaction to eliminate a double-spending attack.

Due to the small block rewards and high stale block rates of Ethereum compared to Bitcoin, Ethereum requires at least 37 confirmations with six block confirmations against the 30% adversary of total mining power. All these numbers are derived by comparing the security of Bitcoin. The findings also convey that, higher the block rewards in the blockchain network, more resilient is the network against adversaries.

By changing the block size and block intervals on the selfish mining and double spending, the security was surprisingly not compromised. The block size was set to average, i.e., 1 MB and block interval were reduced to one minute, the results proved that Proof of Work could attain effective throughput after 60 transactions per second without compromising on the security [32].

3.1.4 Node Identity Management

The public network allows any user to be a part of the network and contribute to it. Although the public based network allows participants, there are still some barriers to the entry of the network. In a public PoW network, a node requires specific computational power to be a part of the network. It is also made clear that just because a node is a part of the network does not mean that it can actively participate in the transactions [33].

The Proof of Work on a public network offers excellent data mutations after specific blocks are appended to the blockchain network. As said earlier, they have some constraints like the cost involved is high to be a critical node in the network, due to the scalability issue associated with the Proof of Work, the time taken for a transaction to get added to the network is prolonged. The public network is vast due to the permission less attribute that any user can be a part of the network that makes the network less private and confidential [34].

3.1.5 Data Model

The Proof of Work can either be transaction-based or account-based.
Transaction based is the Unspent Transaction Output Scheme, where all the unspent transactions are saved in a synchronized node for peer nodes in the network to verify a transaction. The account-based is typically the convention banking account model.

3.1.6 Communication model

The Proof of Work is set on an asynchronous model, and the asynchronous model has no upper bound delay of messages within the network. The asynchronous communication model makes the process of achieving consensus difficult in Proof of Work. The delivery of messages can take indefinite time to reach the intended node. The nodes can arbitrarily drop out of the network and rejoin. The reason why the Proof of Work opted for an asynchronous model is that it is a real-time consensus algorithm that does not wait to receive a response or any form of acknowledgment. The asynchronous system does not provide any response to the nodes about the delivery of data. The nodes tend to have a different view of the overall status of the blockchain network. Moreover, the nodes synchronize only periodically, so the nodes in an asynchronous PoW network seem to have a different opinion about the overall health of the network [35].

3.1.7 Electing Miners

The miners in the Proof of Work are elected based on the criteria, who finds the nonce value and produces the hash. Then the nonce is solved based on the standards based on the hash value.

3.1.8 Energy Consumption

The energy consumption is the amount of energy or electric power consumed by the blockchain network. There are no significant energy savings in Proof of work as there is high consumption of Electric power due to the computational power expected to meet the network's requirement [36].

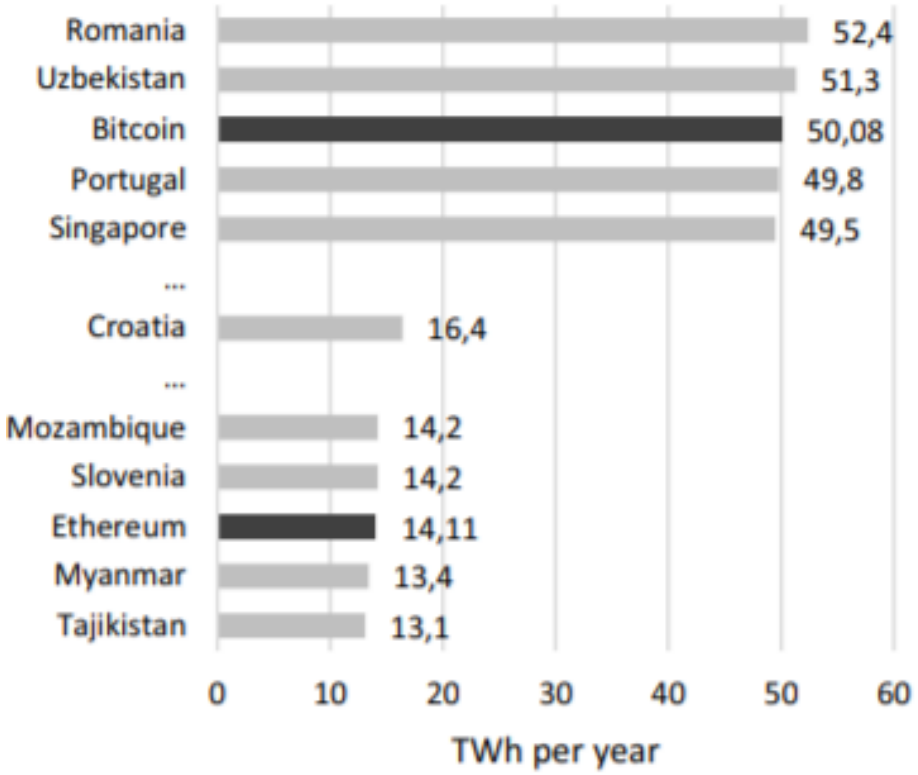


Figure 26 Energy consumption of PoW Based Bitcoin [37]

3.1.9 Tolerated Power of Adversary

The adversary of Proof of Work is less than 25% of the computing power, which is less compared to the highest adversary power recorded, which is 51%. The adversary model controls a part of the blockchain network; the role of it is to resist attack imposed by the opponent without causing any disturbance to the consensus. The results suggest that the adversary threshold be high in order to withstand any attack [36].

3.1.10 Transaction Fees

In Proof of Work, every transaction that happens in the network is charged, the Bitcoin model based on PoW has high transaction fees compared to the time when it began.

3.1.11 Block Reward

The Proof of Work offers block reward to qualified miners who first solve the puzzle. The complexity of the puzzle is decided based on the overall power of the network.

3.1.12 Communication Complexity

For the communication complexity of Proof of Work, we are going to analyze using Bitcoin. The analysis is developed over the work of Garay, which proves that the Bitcoin met three properties like Chain Quality, Common prefix, and Chain growth. The Proofs were explained based on a of a typical assumption, which states that the nodes produce blocks at a rate close enough to their hash power. The mathematical expression shows that the execution of k rounds in typical with the probability. The ε variable represents the measure of how close to the expected value of the block production. The proofs rely on the ε value, which is wholly conveyed in the honest majority assumption. This assumption conveys that ε is that depends on α and Δ/p. Where p is time required to create a block. The execution time longer is longer by $\Omega_{\alpha, \Delta/p(k)}$ times. The probability is given as

$$\text{Probability} = 1\text{-}e^{-\Theta\varepsilon(k)}$$

The system liveness is described in terms of a mathematical equation as $u >= (4k/1\text{-}\varepsilon) = \Theta_{\alpha,\Delta/p}(k)$.CQ, CP and CG are based on the normal execution and are proved. The execution is done using the assumptions, $kp = k_q$ should be greater than or equal to 2kf and $\tau = (1\text{-}\varepsilon)f$.

3.1.13 Verification speed and Throughput

The verification speed is greater than 100, and the throughput is less than 100.

3.1.14 Block Creation Speed in Proof of Work

The block creation speed is low in Proof of Work, as the PoW takes much time as the new block is created only after proper verification process and solving the hash value using the computation power.

3.1.15 Scalability

Scalability is the ability to meet the ongoing requirements. Scalability is an essential attribute in blockchain, along with decentralization and security. The scalability helps the distributed ledger system achieve an effective throughput equivalent to its counterparts. The biggest hurdle for scalability in the blockchain is the throughput of the network in any given situation should not be compromised. The Proof of Work implementations has admitted that scalability has been compromised [33].

3.1.15.1 Issues in Scaling Proof of Work

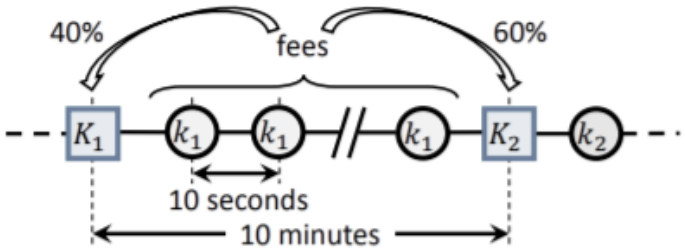
(i) The time taken by the nodes to verify a transaction and reach consensus reduces the scalability in PoW.

(ii) The mineable blockchain that originates from an original work component of Proof of Work that is the time taken to add each transaction to the blockchain network. Although the above-mentioned reasons may seem simple, they are quite deceptive in limiting the scalability of PoW [33].

3.1.15.2 Solutions to Improve Scalability

One method is to use the blockchain protocol Bitcoin-NG based on the Nakomoto consensus, divides the time into an epoch. Each epoch has a single leader, and the leader is responsible for arranging the transactions serially. This mechanism is supported by two blocks created by the protocol Bitcoin-NG (i) Keyblock (ii) microblock. The keyblock is generated by the miners from the Proof of Work mechanism, these blocks are responsible for electing a leader, and they do not contain any transaction. The leader has the capability to generate the microblock, which contains the transaction data, the process of transaction confirmation can progress continuously until the next leader is selected, thus reducing the transaction confirmation time, which improves scalability simultaneously [38].

Increasing the block size can improve the scalability in PoW. When the block size increases, it can mine more transactions as the block size accommodates more transactions than usual. When the number of transactions increases, the time taken by the blocks to reach the full status is achieved sooner [33].



Structure of Bitcoin-NG [39]

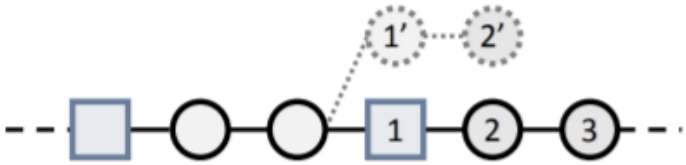


Figure 27 Microblock frequent Representation [39]

3.1.16 51% Attack

In Proof of Work, each node tries to find a nonce value to produce a hash, calculating the nonce value is complex, and it is a mathematical problem calculated based on the criteria of the hash value. Once the nonce value is found, a block is generated. This generated block is then advertised to the blockchain network, according to the Proof of Work consensus, the longest chain is always accepted by the peer nodes. Based on the consensus mechanism, a node is selected, which has the right to seal a block. This process is called mining. The node with a high computing rate will have the ability to calculate the nonce value faster. Sometimes these backfires if a selfish node that has high computational power or rate than the total computational rate of all nodes combined can compromise the network leading it to a 51% attack [40]. In ideal conditions, to conduct a 51% attack is quite expensive, considering the massive network and owning high computational power.

List of PoW 51% Attack Costs for Each Cryptocurrency

Name	Symbol	Market Cap	Algorithm	Hash Rate	1h Attack Cost	NiceHash-able
Bitcoin	BTC	$123.38 B	SHA-256	33,511 PH/s	$582,622	2%
Ethereum	ETH	$52.58 B	Ethash	216 TH/s	$364,099	3%
Bitcoin Cash	BCH	$15.79 B	SHA-256	4,013 PH/s	$69,773	13%
Litecoin	LTC	$6.47 B	Scrypt	309 TH/s	$65,298	7%
Monero	XMR	$2.51 B	CryptoNightV7	370 MH/s	$20,048	14%

Figure 28 51% Attack on some cryptocurrencies [41]

3.1.17 Double spending attack in Proof of Work

To analyze the security in Proof of Work, we shall assume a quantitative framework. The two factors which influence the double-spending attack are block size and block interval calibrated. The results show that, when the block reward is high, it will significantly prevent the double-spending attack. The block size set to 1MB and block interval to one minute shall not compromise the security of the blockchain network [36].

3.1.18 Byzantine Fault Tolerance and Crash Tolerance in Proof of Work

The BFT is the ability to tolerate failures in a blockchain network, which is 50 % in PoW. Potential failures in a network can be an attacker or intruder, dead nodes, corrupted data. These can prevent the system from reaching consensus. Crash tolerance is also 50%, meaning how far POW can endure if the whole blockchain network crashes due to an attack or system failure.

3.1.19 Summary of Metrics

METRIC	DESCRIPTION	PROOF OF WORK
NODE IDENTITY MANAGEMENT	The information/data regarding the transaction available for public use, private or consortium	Public
DATA MODEL	Handling of information by the blockchain. The different types are transaction-based, account-based and key-controlled	Transaction based and Account based
COMMUNICATION MODEL	The model through which information is passed through the network. The different types are synchronous, asynchronous and partially synchronous	Asynchronous
ELECTING MINER BASED ON	The working mechanism of how the block creator is selected.	Solving complex hash
ENERGY SAVING	The energy consumption during the whole process	No energy saving
TOLERATED POWER OF ADVERSARY	The percentage level required in the network power to attack the security of the system	Less than 25% computing power
TRANSACTION FEES	The fees generated whenever a new block is created	Provided for all miners
BLOCK REWARD	The coins obtained from creating a block	Provided for the first miner
VERIFICATION SPEED	Total duration required to complete all validation process.	Greater than 100 seconds
THROUGHPUT	Number of transactionss per second	Less than 100
BLOCK CREATION TIME	Time duration to obtain the confirmation of transaction	Low
SCALABILITY	Ability to expand the system by meeting the ongoing requirements	Strong
51% ATTACK	The attack done by the 51% network power holder	Yes, Occurs
DOUBLE SPENDING	The attack by duplicating the transaction for new block creation.	Yes, Occurs
BYZANTINE FAULT TOLERANCE	To resist certain level of failure in the node	50%

3.2 Proof of Stake

The main aim of using the Proof of stake algorithm is to use it as a distributed consensus. The distributed consensus is mainly used in a distributed network to achieve system reliability. The Proof of stake algorithm reduces the computational power and time taken by the Proof of work to allocate each block to a miner. Sunny King and Scott Nadal developed the very first Proof of stake in 2012. The basic idea was to stack the tokens instead of using computational power and reduce the work done by the miners in the network. The mechanism used for selecting the miners is the coin-age based selection, which is simply the product of the number of coins and the days for which it has been held. The basic rules in the coin-age based selection are as follows; the unspent coins wait for a time duration of 30 days after that they start to work on the next block. Once a block has been assigned to that miner/ when the block is signed, it must wait for another 30 days before stacking coins for the next block. The possibility of obtaining the next stake is reduced after 90 days to reduce the extensive collection of the stack to dominate the network. The more the number of coins and the period higher is the possibility to select. With this selection process, the leader is no longer chosen by computational power but by their stack. By doing so, there is a reduction in energy consumption when compared to the Proof of work. Another main disadvantage rectified in the Proof of stake mechanism is the time duration. Usually, the time duration is kept at meagre constant rates so that the network can be secure as the miners handle different blocks. This scenario is rectified in the Proof of stake mechanism because here, only one block is used, making the generation and confirmation speed higher than Proof of work. [42] [43]

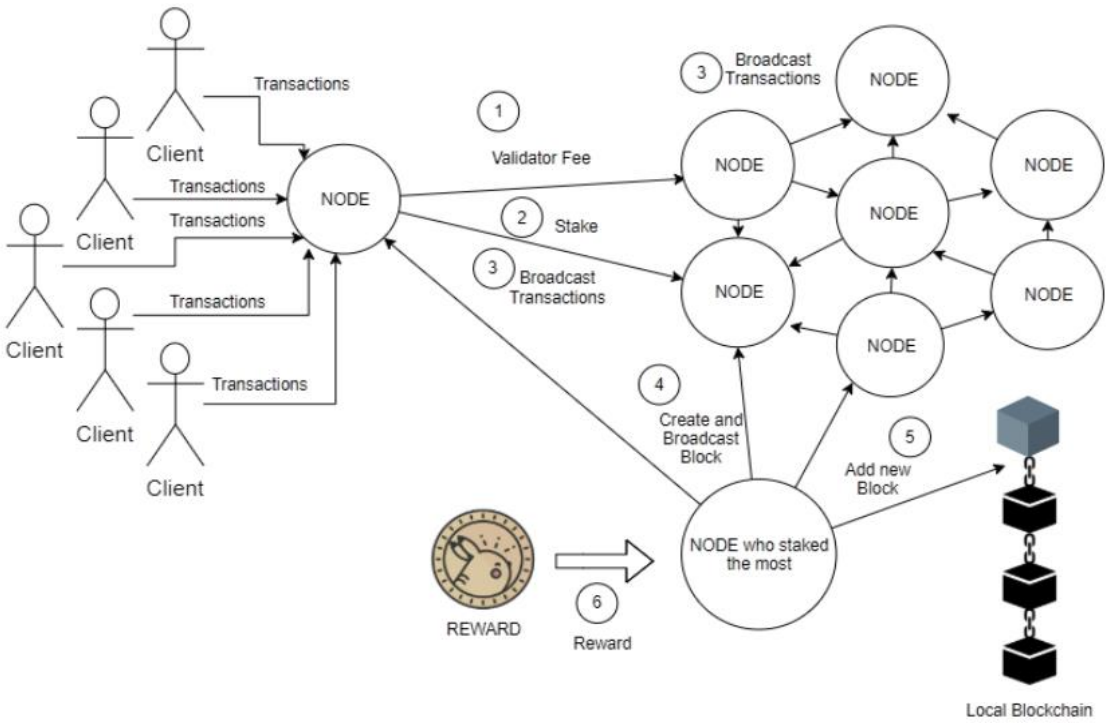


Figure 29 Basic structure of Proof of Stake [44]

The above figure [30], explains the basic structure of the Proof of Stake consensus algorithm mechanism. Initially, all the clients are connected through a node. This node in the network is the validator, and the validator fees are paid. The transaction process is the second step of this mechanism. When the transaction is completed, the validators can stake coins/cryptocurrencies to compete against one another. After the transaction, each node should display all the transactions to the clients. Then according to the algorithm, the highest stake is made the leader and becomes the block creator. The leader is selected through a combination of factors such as random selection (hash function), the time duration and the highest stake value. After all these considerations, the block creator is selected and connected to the chain. This information is also broadcasted to the whole network. The reward protocol is applied here, and the block creator is provided with the transaction fees as their reward. [44]

The Proof of work and Proof of stake is a blockchain consensus algorithm, the working mechanism is quite different from one another. Frequently cryptocurrencies such as bitcoin, Ethereum use the Proof of work algorithm. Ethereum plans on using the Proof of stake algorithm named Casper, to increase the security. Proof of stake works differently than Proof of work by stacking by the cryptocurrencies as collateral so that they can be verified transactions. The use of Casper will make the Ethereum a Proof of stake blockchain rather than Proof of work blockchain. The implementation of Casper includes two components Casper Correct by Construction (CBC) and Casper Friendly Finality Gadget (FFG).

The main reason for adopting a blockchain algorithm is to add new blocks to the network. This algorithm plays an essential role in determining where the block should be placed according to its capacity.

The working of Proof of stake does not allow any random miner to a block, instead have a specific mechanism for selecting the validator. The process is as follows, the miners are supposed to stack their tokens/balance, which is their collateral, and from the stack, the block creator is picked. The probability of getting selected is highly influenced by the stake amount. The miners whose stake is higher gets the block in the network. The main advantage of using the Proof of stake algorithm is no defrauding takes place. If the miners try to cheat or tamper the block, it will result in loss, the reason being they would risk their own stacked coins. Hence this consensus algorithm would increase the security of the network.

In Proof of Stake algorithm, as the miner is selected by the stack they provide, the stack is in the form of digital coins (cryptocurrencies). The miner is selected by stacking the coins and performing the mining performance so that a new block would be added to the network. Many Proof of Stake blockchain networks (Cardano, Tezos, Sp8de) use the FTS algorithm. The working of the FTS algorithm is as follows; the algorithm consists of a hash value, which is a random string created unique to each node. This hash value is taken as input, and the output is indexed by the FTS algorithm process in the form of a token index. The transaction history is made available, and the algorithm selects from the

transaction history that particular token and selects the miner/leader. The formula through which the particular node is selected is as follows,

$$p_i = \frac{s_i}{\sum_{j=1}^{N} s_j},$$

where Pi is the probability.

Si is the stack of the participant
N is the number of participants

One of the crucial advantages of Proof of stake other than reducing the computational power is the fast transaction confirmation speed compared to Proof of Work mechanism. This speed of the transaction is governed by two parameters, transaction throughput and the block confirmation time. The block confirmation time is the time between the moment the blockchain transaction is provided for the network to the moment it has been confirmed. In simpler words, it is the time taken for the participant to wait while the transaction is obtained and confirmed. Transaction throughput is defined as the number of transactions per second. The throughput transaction is a crucial parameter when there is a number of pending transactions to be performed by the network. [45]

$$Tx/s = \frac{Block_{size}}{Tx_{size} \times Block_{time}}.$$

For instance, if a particular blockchain network consists of a Txsize of 350 bytes with block size and block time of 1MB and 700 seconds, then the throughput time would be given as 4 transactions per second. The throughput time is vital in knowing how fast the transaction is confirmed in the network. Usually, in the bitcoin network, the average time required for the confirmation is around 1 hour. Another difference between the Proof of Stake and the Proof of Work is that PoS makes the block time lesser, and the block size much larger so that the efficiency increases. This particular mechanism makes the PoS have high throughput time. For instance, the Ethereum has a high throughput output of up to 875 Tx/s. There are cases where the transaction confirmation time is wholly reduced to one second. Thus particular cases can achieve immediate finality. There are also cases where the time is significantly increased because when there are multiple chains, the honest miners prefer the most extended chain rule resulting in delayed time for the confirmation.

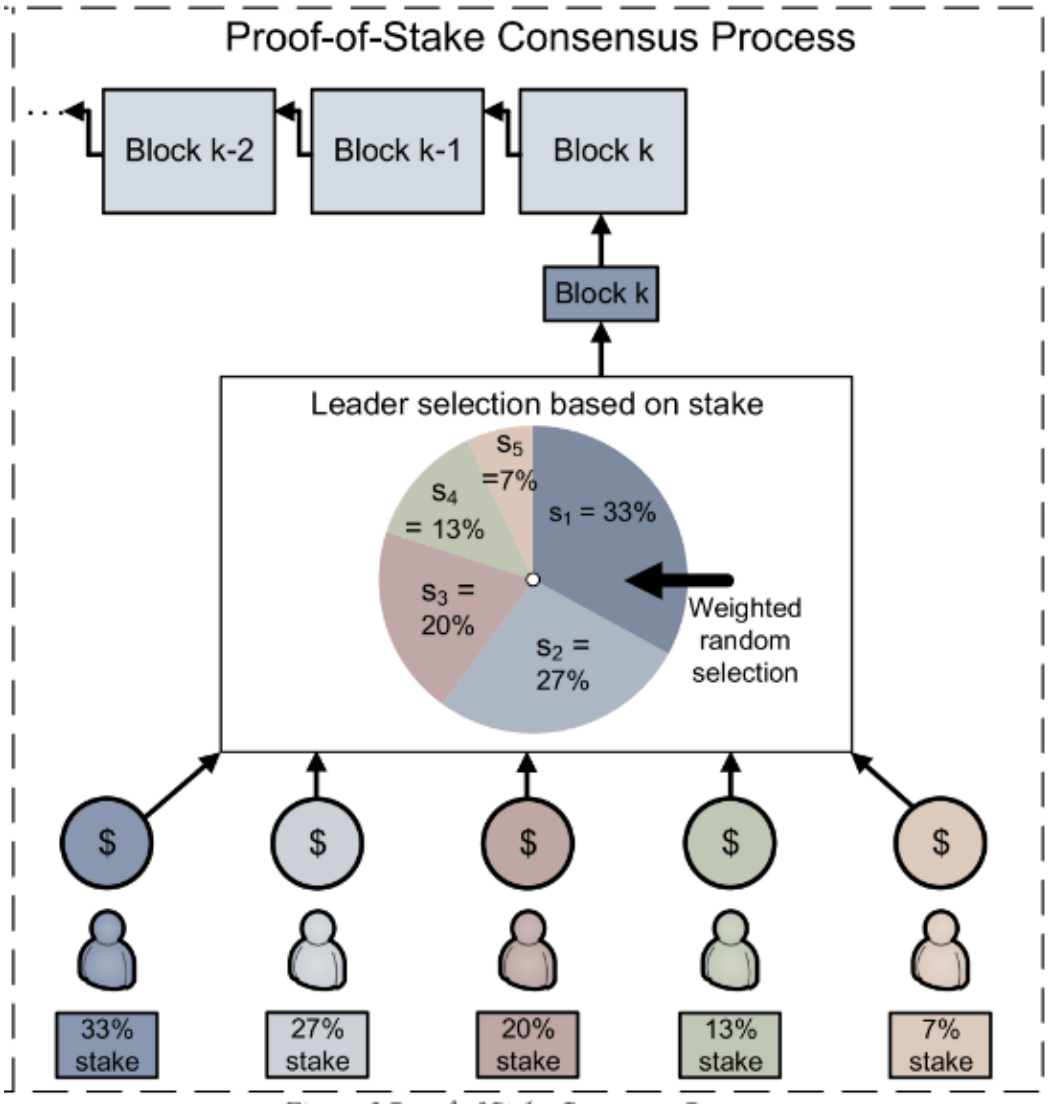


Figure 30 Proof of Stake consensus process [46]

The working process, for the above example, goes as follows, there are five miners taking part in the process. Each miner offers their stake and using the Proof of Stake consensus algorithm; the highest stake is made the leader followed by the rest of the miner/validators. Here the first participant offers 33%, the second offers 27%, the third offers 20%, fourth offers 13%, and the last participant offers 7%. The Proof of stake algorithm calculates the highest stake and the time duration as well before deciding the leader. When the weighted random selection is made, the miner is selected for each block in the order of the stake.

There are various parameters considered with the security of the Proof of Stake algorithm. The Proof of Stake consensus algorithm involves the necessity to send messages during the leader selection process. The leader selection process consists of voting, and the voters

need to send their votes to other miners. These messages need to pass accurately to other participants.
Hence network synchrony plays a vital role in keeping the network secure and reliable. The network synchrony is of two types synchronous and asynchronous. When the message reaches the target within a specific time limit is known as a synchronous network, and when the messages do not even reach the final destination is known as the asynchronous network. The Proof of Stake is a consensus algorithm that consists of both reward and penalty protocols. Similar to Proof of Work, the validators and the block creators are provided with rewards to incentivize their work and participation. The difference between the reward system in Proof of Work and Proof of Stake is that in Proof of Stake, the reward protocol changes from time to time. The reward system mainly depends on the total number of validators in the network at that particular time. The balance should be maintained in terms of reward. The rewards incentives should not be meagre, nor should it be too high. When the reward incentives are meagre, it will result in the reduction of participants, and when it is too high, it would inevitably reduce the value of the crypto asset. In the same instance, penalty protocols are also adopted. The primary reason for attacking the Proof of Stake network is that it is easy to create many blocks in this consensus algorithm. Therefore, the attacks which involve creating a large number of blocks should be penalized. When the consensus is disturbed by not running the required software during that time period is known as slashing. The Proof of Stake has a different set of rules for each action. The commonly used protocol is the loss of their stake. Mostly separate reputation system is attached to the mechanism, which would make the mechanism to identify the trustworthy validators quickly. [47]

3.2.1 Node Identity Management

When it comes to Proof based consensus algorithm, the node identity management is Public. All of these algorithms are used for network are public. The membership control is defined by type of the blockchain. The three different type of blockchain are private, public and consortium. This part is mostly defined at the design part of the network and differs from one application to another. In the Proof of stake working, after the transaction is verified, the aggregate data is broadcasted or displayed in a public ledger.

Usually, the blockchain network is a distributed, decentralized and public. The public blockchain is also referred as the permission less or unpermissioned network. This is mainly used in Proof of stake because of its process of displaying the transactions. This network allows anyone to take part in the block creation/validators and modify it. Every modified data is being updated and displayed to the public ledger. After every transaction and when each node is validated by the validators it is broadcasted. This transparent approach might result in security issues for certain scenarios. Here everyone can read the transaction data but only certain users can do the validation. [48]

3.2.2 Data Model

The account-based model works similarly to the standard banking model of balance management. The account-based model is used in the Ethereum Proof of Stake

mechanism. The mechanism is effortless and can be explained as follows, consider participant 1 has ten tokens, and participant 2 has ten tokens. When participant two wants to send the token to participant 1, it is subtracted from the former account resulting in zero balance in the account. Now participant 1 transactions 10 to participant 2, resulting in equal tokens for both participants. This mechanism is as simple as the example. The advantages of using the account-based model are easy tracking of the transactions, and it also prevents the double-spending attack. These benefits are achieved because of the centralized network mechanism for tracking the flow of the transaction.

In Ethereum Proof of stake, Private key-controlled user accounts and contract-code controlled accounts are used. These two accounts are crucial in determining why the account-based model is preferred over the Transaction-based model/UTXO model. The main reason for using the account-based model is the use of smart contracts; if the UTXO model is used, it will limit the use of smart contracts. The account-based model is much simpler because most of Ethereum applications are decentralized. Every account must have balance, code-storage space for other addresses. The process starts from the sender, and if the receiving end has the code to run, the process continues. The account gets debited if any changes are made, like adding another message by changing the internal storage. Hence these changes affect the whole system and all accounts as new accounts are added to the network.

The different advantages associated with the account-based model are as follows, this model increases the space available compared to the UTXO model because here, every transaction is associated with only one signature and reference, thus saving the space. This space-saving is crucial for the Ethereum because it is a sophisticated platform. The added advantages are simplicity and familiarity. The probable disadvantage in this model is its limitation in terms of scalability. This limitation becomes a bigger concern when it comes to a broader industry. [9]

3.2.3 Communication Model

The communication model is of three types synchronous, asynchronous and partially synchronous. All these models involve the sending and receiving of messages in the network. When there is a time duration associated with the network, then it is known as a synchronous network. A specific limit bounds the time limit. In the case of an asynchronous network, it is not bound by any time limit. In the case of a partially synchronous network, the system remains in the asynchronous state for a specific time limit, after which it changes into the synchronous network. The synchronous model is governed by an upper time limit bound and upper bound for the speed limit as well. In asynchronous network there is no upper bound on the speed and the time. It usually takes arbitrary duration of time to receive and respond to messages. As Proof of Stake mechanism uses an asynchronous network, it is difficult to achieve specific parameters described in the FLP impossible algorithm. In a consensus algorithm, it is difficult to achieve all three parameters of consistency, fault tolerance and availability. No network can achieve all three parameters at the same instance; it changes according to the applications. For instance, if it is a distributed network application, it is preferred to

consider safety over fault tolerance. Generally, both consistency and availability cannot be achieved at the same time because, during a specific period, some messages are intended to be dropped during the process. This concept is even described in the CAP theorem.
In the Proof of Stake, the total duration is divided into three categories, namely pro-pose, pre-vote and pre-commit. This division of period makes the mechanism a weak synchronous protocol (asynchronous). [48]

3.2.4 Electing Miners

In proof of stake, the miners are elected based on their stake. Apart from that, the two most essential parameters involved are the randomization (hash value) and the stake age. All these parameters together are involved in electing the miners. Generally, in this consensus algorithm, the blocks are forged rather than mined. Two methods of selection are done in PoS to select the leader. Randomized block selection and coin-age selection. The randomized block selection is concerned with the hash function, whereas the coin-age based selection is concerned with the stake of each participant. The validator whose stake is the highest and the node wealth is high would be made the block creator. Here the transaction fee is provided as the reward.

In the randomized block selection, the validators are selected based on the lowest hash value and with a higher stake value. In the coin-based selection, the process involves the product of the duration (No. of days) the coins have been held and the stake value. The combination of both is crucial in selecting the miners for the network. Once the block is allocated in order to obtain another block, the coin-age starts from zero so that domination by one significant stake can be avoided in the blockchain.

Many validation processes take place during transactions to ensure security before signing that particular block, which is added to the chain. Once the node decides not to participate in the mining process, all its stake and transaction fees are returned. However, it is not done immediately; it takes a certain amount of time so that no fraud takes place in the node. [49]

3.2.5 Energy Saving

In Proof of Stake, the energy consumption is not that high when compared to Proof of Work. This is mainly because, in Proof of Stake, no high computational power is required to solve resource-intensive puzzles. Hence it is more energy-efficient, and any cryptocurrencies involving this consensus algorithm are more reliable and efficient in the long run. As Ethereum has plans to use the Proof of Stake consensus algorithm, they plan to reduce the energy consumption by a significant amount. Ethereum's goal is to use smart contracts, which helps in reducing the energy consumed during this whole process. Typically, the energy consumed in cryptocurrency is way high (more energy required for gold mining), which will be considerably reduced in Proof of Stake mechanism. [49]

3.2.6 Tolerated Power of Adversary

Tolerated power of the adversary needs to be high in any consensus algorithm. Usually, a certain percentage of network power is used to attack the security of the network. This tolerance value should be high, and it is less than a 51% stake in the Proof of Stake, which is better compared to Proof of work and PBFT. The highest tolerated power of the adversary is around 51 %.

3.2.7 Transaction Fees

Transaction fees are paid to all the miners in the Proof of Stake consensus algorithm mechanism. In this algorithm, the transaction fee is in the form reward; it acts as a nodal reward. Here the transaction fees are collected whenever a block is created and is used to help incentivize the miner and make the blockchain growing. The transaction fees are also not provided separately; it is given as a part of the whole transaction process. Proof of Stake uses the transaction fee as a reward because it is obtained from the stake of the miners by building the coins/token for each block.

3.2.8 Block Reward

Block reward consists typically of the coins obtained from creating each block. It is given to the node after the transaction is completed. Block reward is not provided in the Proof of Stake consensus algorithm.

3.2.9 Communication Complexity

One of the networks which use Proof of Stake consensus algorithm is Ouroboros. It is implemented by Cardano for cryptocurrency. Here the process id executed by obtaining information about the previous broadcast of information for the current coin tossing. Since the working is not similar to Proof of Work, the randomization selection is generated for the leader selection. This process is complicated, but the use of a single leader at a particular instant helps in this scenario.
When it comes to the distribution of honest leaders and the attackers in the block tree, it can be used to prove the CP, CG and CQ properties. This process is done from the schedule of the block tree (encoding process). This encoding process is also known as analysis of characteristics strings. The output obtained displays a string of length k with probability neg(k). The system consists of at least one honest leader in each round of execution.

$$K = \Omega_{\alpha}(k)$$

$\tau = 1-\alpha$ (as the system has at least one honest leader in each round of execution) where the distribution string depends on α.

3.2.10 Verification Speed

Verification speed is the total duration required to compute all the validation process. The validation process plays a vital role in the mechanism because it prevents fraud by creating more blocks as it is simpler to create blocks in Proof of Stake consensus algorithm. The verification process is crucial at the receiving end as well, mainly because the generated hash value needs to be compared with the input hash value. The number of transactions is also indirectly connected to the verification speed. Hence at the beginning, the verification speed is comparatively higher to the subsequent transaction. The verification speed in the Proof of Stake consensus algorithm is less than 100 seconds, which is beneficial compared to the Proof of Work algorithm. When the verification speed is high, it ultimately reduces the entire transaction process as well.

3.2.11 Throughput

The throughput time is vital in knowing how fast the transaction is confirmed in the network. Transaction throughput is defined as the number of transactions per second. Throughput for Proof of Stake algorithm is less than 1000. It is the rate at which the transactions are completed in a specified time period. It is can also be represented as the total committed transactions divided by the total number of seconds at the number of committed nodes. The block time is lesser in Proof of Stake consensus algorithm, and the block size is much larger so that the efficiency increases. This particular mechanism makes the Proof of Stake have high throughput time. The blockchain work is a function of the throughput and the network size. Hence the throughput transaction is a crucial parameter when there is a number of pending transactions to be performed by the network. [45] [50]

$$Tx/s = \frac{Block_{size}}{Tx_{size} \times Block_{time}}.$$

3.2.12 Block Creation Time

The block confirmation time is the time between the moment the blockchain transaction is provided for the network to the moment it has been confirmed. In simpler words, it is the time taken for the participant to wait while the transaction is obtained and confirmed. Once the block is confirmed the block is created. The block creation speed is high in Proof of Stake compared to other consensus algorithms. In Proof of Work the block creation changes according to the difficulty level but that is not the case in Proof of stake hence its speed is much higher. [45]

3.2.13 Scalability

Scalability forms an essential role in the decentralized network. It refers to the ability to expand the system by meeting the ongoing requirements. A decentralized network needs to achieve the transactional throughput of an expanding network. The different solutions for enhancing the scalability of the network are by developing the consensus and the data structure, modifying the size of the block and developing second layer solutions.

When the block size is modified/increased, it will result in large capacity and reduce network congestion. Nevertheless, the increase in block size will also result in more transactions to take place in a shorter interval of time. The increase in the number of transactions will lead to an eventual delay in the confirmation of a transaction or even block them.

The process of obtaining throughput by horizontal scaling is known as sharding. The simultaneous transaction of multiple shards becomes more effective than processing a single transaction/mining at a time. Each shard consists of its block history and state information so that individual transactions can take place. Each shard is related to the main blockchain so that all the information is up to date. The problem arises when there are way too many shards that might require further scaling making the network congested. Thus, scalability increases with the network growth.

Proof of Stake facilitates sharding but has limitations when it comes to scalability regarding the throughput of the network. Proof of Stake allows scalability in the block confirmation time as it does not have any computational problems to solve but not in a significant manner. [33]

3.2.14 51% Attack

In Proof of Stake, it is not easy to make the 51% attack; it would require the attacker to own 51% of the tokens, which would be a considerable amount to obtain. Unlike Proof of Work, where 51% would mean computing the puzzle of the network. Moreover, in Proof of Stake, for a 51% attack, the attacker is compelled to buy 51% of the stake where the price increases as the tokens are bought. It is complicated to attack a Proof of Stake network as everything is public. Once the whole network knows that a particular address is buying many tokens, it is considered as a warning, and the attack is stopped even before it could happen. Even if the attack takes place, the value of it reduces in the network, making it a loss eventually. The tampering in the network would result in the loss of the attacker. Hence there is no benefit in attacking a Proof of Stake network.

Four common cases come under the 51% attack, such as finality reversion, where the finality guarantee is broke by finalizing another block. The next case is invalid chain finalization where unavailable blocks are finalized, then liveness denial and finally censorship. The third case is completed reduced in the Proof of Stake consensus algorithm. When the validators stop confirming the blocks, those node weights are reduced by removing them.

When a specific attacker address is identified, it is blacklisted, and it would be tough for the attacker to repurchase the tokens. Once the address is identified, all the stake is deleted, and then the value of the tokens is increased, making it even more complicated for the attacker to initiate another attack. [51]

3.2.15 Double Spending Attack

The Double spending attack is when the same digital token/coin is used to duplicate another transaction in the block creation. The literal meaning is spending the same money twice for different transactions. Most of the attackers would encounter double-spend at some point in their process. The attacker makes an initial transaction and then reverse it to complete another transaction. It can be easily identified if the transaction takes place in the same branch, but attackers usually do it in another branch by conflicting the initial transaction.

This attack would be possible only if the attacker holds the highest share, which would be way too expensive in the first place. This attack is considerably prevented in Proof of Stake consensus algorithm. The attack does not work on this algorithm because it is irrational for a high stake holder to waste all the resources in stake on every chain of the network. Similarly, it is useless in attacking and risking their investment. It would result in the attacker losing all the stake. [52]

3.2.16 Byzantine Fault Tolerance

Byzantine Fault tolerance is derived from the Byzantine General problem. It is the ability to resist a certain level of failure in the system. The system has a certain tolerance level to the failures in the nodes without affecting the whole network. The Byzantine fault tolerance level in Proof of Stake is around 50%. The tolerance level applies to situations where the messages are not correctly sent, or it takes time to send the messages. 50 % tolerances indicate that in the network, half of the validators should be honest for the system to work efficiently. The Byzantine fault tolerance is concerned with the security of the network. [53]

3.2.17 Summary of Metrics

METRIC	DESCRIPTION	PROOF OF STAKE
NODE IDENTITY MANAGEMENT	The information/data regarding the transaction available for public use, private or consortium	Public
DATA MODEL	Handling of information by the blockchain. The different types are transaction-based, account-based and key-controlled	Account-based
COMMUNICATION MODEL	The model through which information is passed through the network. The different types are synchronous, asynchronous and partially synchronous	Asynchronous
ELECTING MINER BASED ON	The working mechanism of how the block creator is selected.	Stake owned
ENERGY SAVING	The energy consumption during the whole process	Partial saving
TOLERATED POWER OF ADVERSARY	The percentage level required in the network power to attack the security of the system	Less than 51%
TRANSACTION FEES	The fees generated whenever a new block is created	Provided for all miners
BLOCK REWARD	The coins obtained from creating a block	Not provided
VERIFICATION SPEED	Total duration required to complete all validation process.	Less than 100 seconds
THROUGHPUT	Number of transactions per second	Less than 1000
BLOCK CREATION TIME	Time duration to obtain the confirmation of transaction	High
SCALABILITY	Ability to expand the system by meeting the ongoing requirements	Strong
51% ATTACK	The attack done by the 51% network power holder	Does not occur
DOUBLE SPENDING	The attack by duplicating the transaction for new block creation.	Does not occur
BYZANTINE FAULT TOLERANCE	To resist certain level of failure in the node	50%

3.3 Delegated Proof of Stake:

In the Delegated Proof of Stake, the delegates are elected by token owners to validate the blocks. The voting power lies in the number of tokens owned by the token owners, and the voting can be redirected at any time. The blocks are shuffled, and the block producers produce them. The block producers are also called delegates. The block producers create new blocks and get rewarded for it, but if they are not able to create a block on a specific time frame, then the stakeholders vote for a new block producer. Typically, in a Delegated Proof of Stake, there are twelve delegates . The Delegated Proof of Stake can be designed efficiently; the block producers are selected on a priority basis; by doing so, the consensus algorithm can produce efficient throughput, and it lowers the block production time. These are the major constraints found in the Proof of Work and Proof of Stake. The delegates in Delegated Proof of Stake, unlike Proof of Work or Proof of Stake, delegates collaborate with each other to create block rather than compete with them. The EOS in Delegated Proof Stake does not charge the nodes for the transaction; instead, they calculate the network attributes like CPU power, RAM utilization based on the stakes owned by the nodes [54].

The Delegated Proof of Stake allows anyone to be a participant of the network to govern the blockchain, and it symbolizes the idea of decentralization more realistically than other consensus algorithms. The DPoS is better is many ways than PoW, and PoS, like anyone with minimum coins, can take part in the network directly or indirectly- the users with low coin rate would receive the same benefits as quicker transactions, energy-efficiency, attack-proof. The DPoS coins are not suitable to be used in the private network as it would lessen the confidentially and privacy as anyone is allowed to be a part of the network, this can attract attackers to crash the system easily. The DPoS has the same challenges as PoS if used in a private network as in the voting privacy, balance privacy and stake have to be guaranteed. The idea of utilizing the DPoS in the private network can be an alternative to the conventional governing model to attain a consensus [9]. In reality, it impossible to attack a DPoS system, which involves the removal of delegates and backups, which are globally trusted [8].

Dan Larimer had primary reasons to design the Delegated Proof of Stake, as he felt the Bitcoin's Proof of Work seemed too much of computational power, consuming much energy and consumed too much of mining power. His idea was to achieve the throughput and performance in PoW using alternative solutions. His focus was mainly on designing a consensus which can compensate for the shortcomings of Proof of Work like speed efficiency and redundancy. The design of DPoS involved robustness, at the same time, still maintaining the decentralized and open infrastructure. The Delegated Proof of Stake is implemented in many platforms like Steem, EOS, Bitshares; many more applications prefer to use DPoS governance model for their architecture like Ark, Cardano, Lisk. To know more about the Delegated proof of Work, we have study about the block production technique and other functional attributes of the consensus [55].

3.3.1 Block Production In DPoS:

The Proof of Work and Proof of Stake seemed inefficient due to their high energy consumption and blockchain speed. The whole network participates in the transaction validation. The concept of a whole network involving in the transactions or block validation is necessary for censorship-resistant creation of neutral blocks. In Bitshares, delegates serve the role of timestamping the transactions and validating signatures.

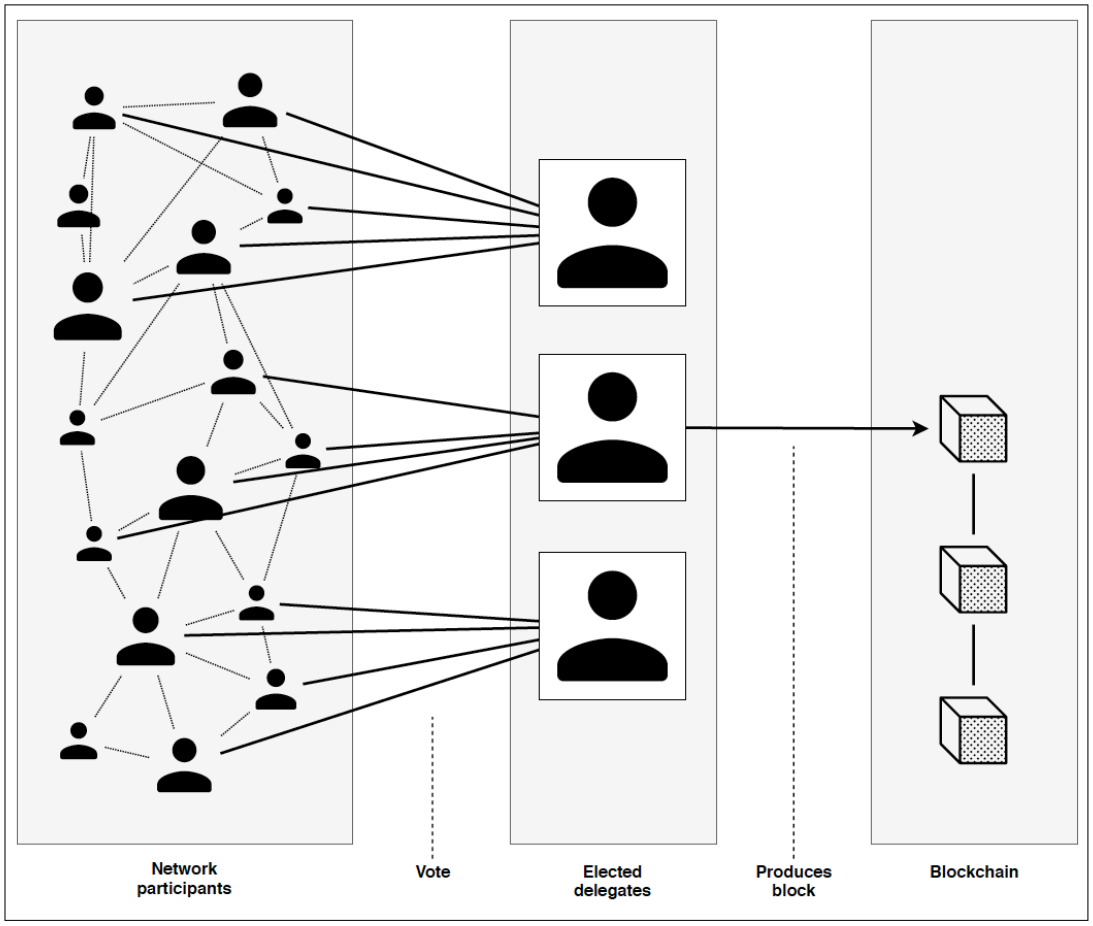


Figure 31 High level View of Generic Realization of DPoS consensus in a block chain [3]

The block producers are selected among the group of delegates during the block production hour. The delegates are elected by the token owners who cast their vote to the voter's stake in the blockchain network. The election process provides a chance for the token owners to commission a delegate to represent their assets that are present in the cold storage- it is called cold staking [3]. At the end of the election, the delegate who gathered the majority of votes is elected, the number of vacancy spots for a delegate is usually an odd number, but it differs from network to network. If the number of delegates in a DPoS network is on the higher side, then decentralization can be achieved with low performance. Generally, the number of delegates in an average DPoS network is lower than the nodes that are present in the PoW and PoS network; this is reason DPoS achieves

better centralization control within the blockchain network than its counterparts. The delegate selects a block to become a block producer. This block producing process can be compared to the round-robin effect. Within a specified time, the delegate must produce and block and hand over the block to the next block producer to forge the block, the round-time is directly linked to the blockchain's specific time, and this never changes [55]. The delegate during the process of producing a block creates the block in reference to the consensus node. By doing so, the block shall contain the current status of the network so that it can be validated and added to the blockchain. Especially this function of the delegates in producing block shows their importance in the Delegated Proof of Stake consensus.

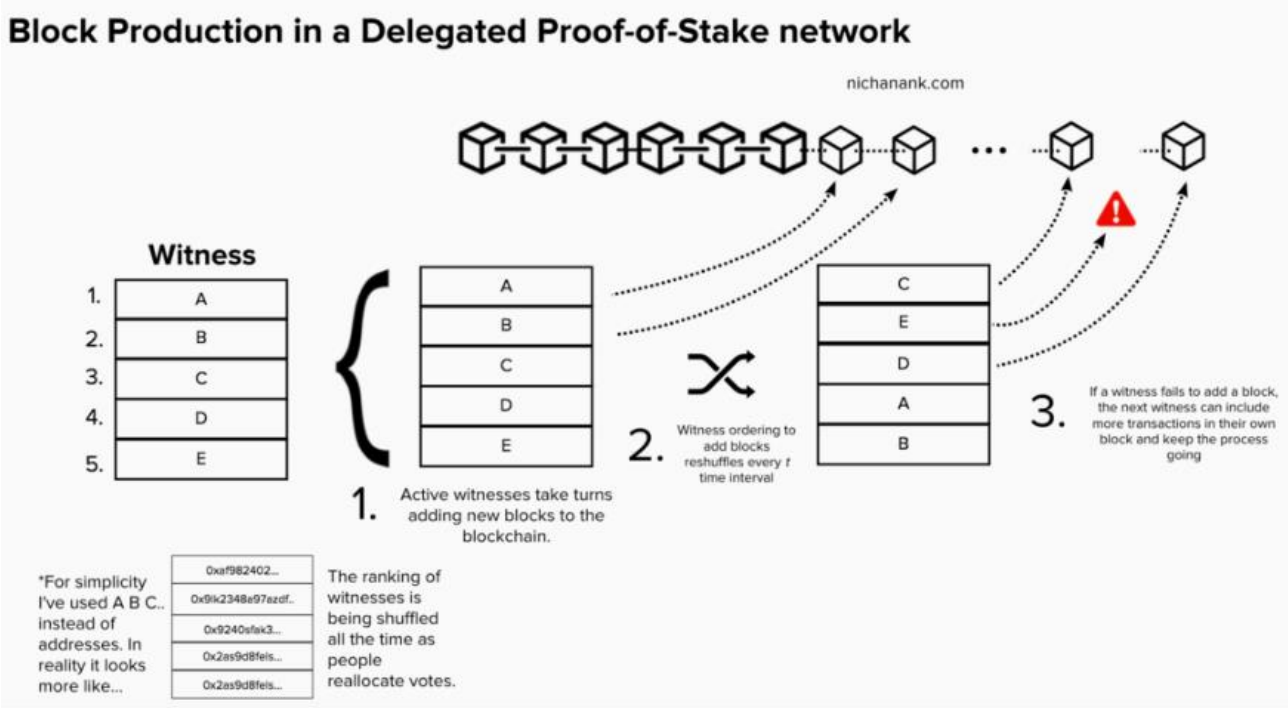


Figure 32 Representation of Block Production in DPoS [56]

3.3.2 Election Process & Block Reward

The election process in DPoS differs from network to network in terms of the opening time of the election, duration of the election, ending of the process depending on the terms and conditions of the network. The protocol that is followed to elect a delegate is the same across all the DPoS.The token owners cast their votes to elect a delegate to become a block producer. The token owner, after voting for a particular delegate, the vote is given to that delegate for a specific amount of time, and the voting process is tacitly reconducted if the voter did not change its vote [9]. This means that the voter can change their decision to vote for a different delegate during the time allotted to vote. Unlike the other consensuses like PoW and PoS, the Delegated Proof of Stake provides incentives to voters, the block reward given to a delegate for producing a successful block is shared

among the voters for their trust. Towards the end of each election, the next one begins, providing an opportunity for voters to reconsider their position in the voting. At the end of each round of the election, the network checks the status of the delegates and compares it with the previous round list. The delegate's role comes with great responsibility, as they are an integral part of the DPoS network; the delegate willing to become a block producer should have a well-proven record and positive reputation. Building a reputation can be done by inviting new members to the community, marketing, funding [56].

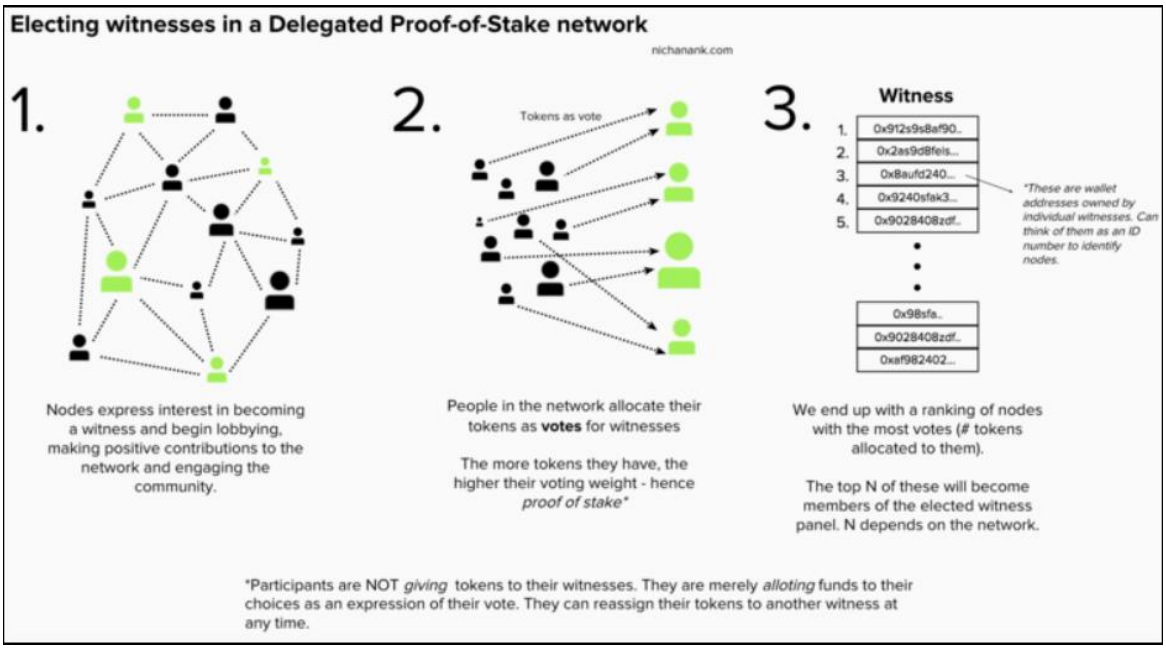


Figure 33 Electing a Delegate [57]

3.3.3 DPoS And Security

The Delegated Proof of Stake is required to provide a network that can protect its users from attacks and security breaches. The attacks in the DPoS are divided into two types (i) Attacks the originate within the network; bad actors disguising themselves as legitimate users. (ii) Security attacks from outside, which penetrates the network to destroy it.

(i) Attacks From Network Actors

Usually, every network tends to have a bad actor within acting as a genuine user. The bad actor will have gains by attacking the network. The voters within the network are an essential source to a DPoS network as they elect the delegates to become block producers, the voters have to go through the network security and checks and balances of the system. This is because the token owners have to elect the delegate on every round of the election process creates a virtuous circle, in which the delegates have incentive to represent their electorate.

The DPoS must have some mechanism to protect the network from bad actors like PoS networks. The staking system in PoS helps reduce the number of bad actors in the system. The PoW also has a staking mechanism that requires the block producer to prove its interest in the success of the network. The ultimate idea behind this is the significant stake is loaned by the actors in charge of the consensus. This method will ensure that the actors are incentivized to behave in a manner that would not cause any harm to the system. In the DDOS and double-spend attacks, the impacts the market value of the coin, initiating a loss from the collateral evaluation.

(ii) Attack from an outside source

The attack on the network can originate from outside as well. The famous 51% attack on the Proof of Work, acquiring 50% of the computational rate to build a faster chain of block by a selfish miner, is called a 51% attack. This attack could duplicate the transactions and lead to a double-spending attack. Though this attack is popularly spoken off but conducting such types of attacks on a vast network is a highly challenging task due to various reasons like the cost involved to attempt an attack, lack of liquidity in hash rate, or computational rate in the market. Robust the network, tougher it gets for the attacker. The 51% attack is even more difficult in the DPoS network, instead of hash rate, the attacker would stack coins to acquire 50% of the overall power of the network. There are reasons why a DPoS network can prevent a 51% attack, the DPoS network has raised its threshold of the stake from 51% to 67%, making it costly for the attacker. The DPoS has to tolerate the Byzantine Fault Tolerance attacks. For DPoS the attack is called DBFT (delegated BFT). In order to withstand the BDFT attack, the system needs to be more consistent and coherent in the overall system's rate. One implementation to overcome the DBFT attack is to make a node or group of nodes monitor the state of the blockchain consensus. The nodes act as a witness, and their data is verified with actual systems specs.

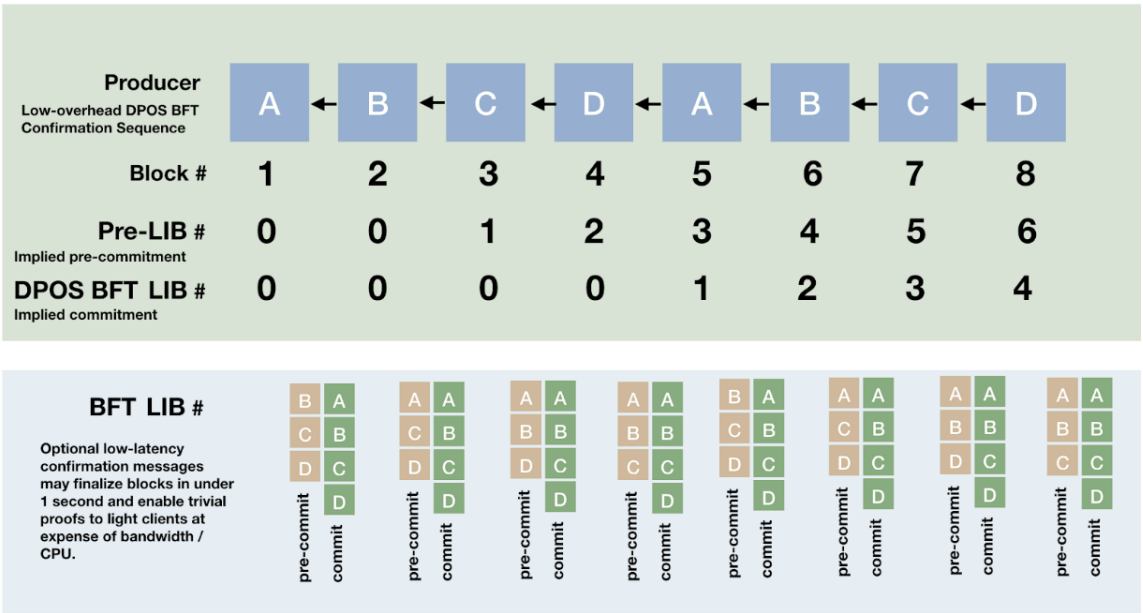


Figure 34 Delegated Byzantine Fault tolerance [58]

3.3.4 DPoS Mechanisms

Cryptocurrencies deploy various mechanisms in their platform, some of the most prominent cryptocurrencies that use the Delegated Proof of Stake as their mechanism are discussed.

3.3.4.1 EOS

The EOS is the first cryptocurrency to use DPoS as their consensus mechanism. It is a popularly known form of cryptocurrency and smart contract platform. Initially, the EOS was created in an Ethereum platform, later EOS was moved to a separate platform. The DPoS in EOS provides remarkable scalability properties, and the number of transactions in EOS per second is higher than Ethereum, it raised four billion USD in an ICO event, which is the highest so far. In a typical EOS, there are 21 validators, also known as the block producers (bp), the block producers are elected by the token owners in the DPoS to produce a block. The number of times a block producer is selected by the token holders to produce a block is proportional to the total number of votes received by the block producer from the token owners. Generally, in every DPoS, an initial supply is created, which is used towards the process of electing the 21 validators or Block producers. The initial supply is put into use to provide rewards to the block producers for creating a valid and successful block. These are steps taken to secure the network by incentivizing the block producers. In one way, this can reduce the security attacks inflicted on the network from the network itself (bad actors). EOS had an initial supply of 1 billion EOS tokens with annual inflation of 5%. 4% of the annual inflation is dedicated to R&D work and 1% for the rewards to the block producers. In EOS, the blocks are created in rounds, for every round, 21 blocks are created. At the beginning of each round, 21 block producers are elected, and then each block producer gets an opportunity to build a block in a pseudo-random fashion, note- these events take place within a particular round. When a block producer produces a block, it must be validated by the other block producers, and the new block must receive at least two thirds or more than two-thirds majority of block producer's validations to reach consensus. Once the validation is over, the block and its transaction are considered confirmed, the block cannot be forked [48].

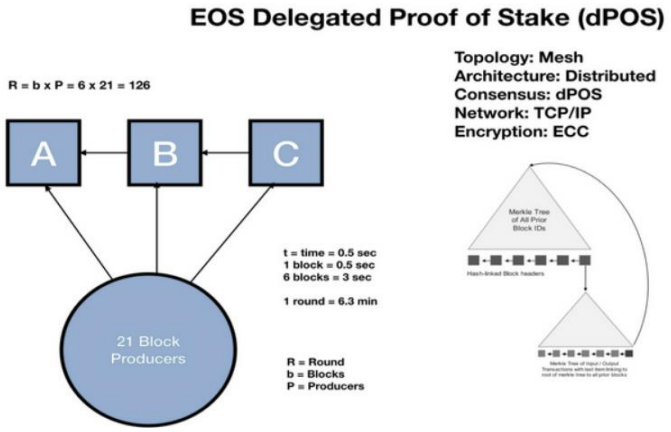


Figure 35 Creation of 21 Block Producers [59]

3.3.4.2 Tron

Tron is another widely used cryptocurrency, and it is a smart contract that has it is working very similarly to Ethereum and EOS. The initial supply of TRON is 99 billion TRON tokens represented as TRX. The consensus mechanism of TRON is it uses 27 validators called the super representatives who are elected by the TRX owners every six hours. The TRX holders must spare some TRX to vote for the super representative. This TRX tokens can be frozen back after three days of casting a vote. The super representatives create a block for every 3 seconds, and they are rewarded 32 TRX for valid ones. One striking feature of TRON is, it does not come with an inflation system, so the total supply of TRON will remain the same throughout its lifespan [59].

3.3.4.3 Tezo

The Tezo is also a cryptocurrency similar to EOS and Tron. It is a smart contract platform that uses a variant of DPoS. The initial supply of Tezo is 765 million XTZ, with inflation of 5.51% annually, the inflated amount is used for distributing block rewards.

The block reward in Tezo is 16 XTZ (XTZ- the currency of Tezo) and the block creation time is sixty seconds, Tezo's do not require a predefined number of stakeholders. Alternatively, the consensus mechanism suggests that a dynamic range of stakeholders who own a substantial amount of XTZ can be a stakeholder. This difference in the consensus mechanism makes Tezo different from EOS and Tron. This mechanism limits the large group of users forms not being able to participate in this particular consensus mechanism, to rectify this Tezo proposed an idea that any user holding XTZ can delegate it to anyone to accumulate the required amount to become a stakeholder. The stakeholder shall give the proportioned amount back to the delegated user from the block reward [59].

3.3.4.4 Ark

Ark is another DPoS based cryptocurrency platform. The Ark employs 51 delegates to create 51 blocks in every round. The block creation time is 8 seconds, and each round last for up to 408 seconds. The initial supply of Ark is 125 million, with inflation of 5.55%. The delegates receive 2 Ark (the currency of Ark) for every block creation. Like every other DPoS mechanism, the delegates are elected by the Ark owners, the weightage of the vote is proportional to the amount of Ark's owned by the voter [59].

3.3.4.5 Lisk

Lisk is a unique DPoS platform that develops the DA apps using javascript. The outstanding feature of Lisk from other DPoS platforms is the ability to collaborate and work with multiple blockchains called the side chains and with the central blockchain called the main chain. The side chain is deployed and maintained by an application

provider. The side chain is synced with the main chain; the presence of multiple side chains enables varied applications to work on it simultaneously without directing the burden to the main chain entirely. The maintenance of the mainchain should be done with the Lisk's DPoS consensus protocol. There are 101 delegates, Lisk utilizes these 101 delegates to create the blocks. Like any other DPoS mechanism, the delegates are elected by the LSK (Lisk currency) owners, and each owner will have 101 votes. The weight of the vote is proportional to the number of LSK owned by the voters. The process of electing a delegate happens before a round commences, there would be 101 block generation cycle in each round. During the round, a delegate is randomly selected to create a block. The block creation time is 10 seconds, and the block reward is 5 LSK. The initial supply of Lisk was 100 million, with an inflation rate of 5.65%, the initial supply of Lisk has gone up to 132 million [59].

Figure 36 Elements that supportthe DPoS link system [60]

3.3.5 Node Identity Management

The DPoS is a public based ledger system, which allows anyone to take part in the network, there are few variations in the DPoS based cryptocurrencies which are little different in allowing members to be a part of the network, example Tezo that has some terms and conditions on the amount of stake, but Tezo being a platform based on DPoS

has other ways to make the network more public. When it comes to any Proof based consensus algorithm, the node identity management is Public. All of these algorithms are used for the network are public. Membership control is defined by the type of blockchain. The three different types of blockchain are private, public, and consortium. This part is mostly defined at the design part of the network and differs from one application to another. Usually, the blockchain network is a distributed, decentralized, and public. The public blockchain is also referred to as the permission-less. The Delegated Proof of stake is usually referred to as the more decentralized public consensus compared to the Proof of Work and Proof of Stake. This network allows anyone to take part in the block creation/validators and modify it. Every modified data is being updated and displayed to the public ledger. After every transaction and when each node is validated by the validators, it is broadcasted. This transparent approach might result in security issues for specific scenarios.

3.3.6 Data Model

The DPoS can be both a transaction-based and account-based data model.The Transaction model is called the UTXO - Unspent Transaction Output scheme. The coins are stacked as unspent transaction output. There is always a spending criterion on the UTXO based models. The existing UTXO's are utilized during the transaction, and new UTXO's are created to replace them. The account-based model works similarly to the standard banking model of balance management. The DPoS can be used as an account-based model. The mechanism is effortless and can be explained as follows, consider participant 1 has ten tokens, and participant 2 has ten tokens. When participant two wants to send the token to participant 1, it is subtracted from the former account resulting in zero balance in the account. Now participant 1 transactions 10 to participant 2, resulting in equal tokens for both participants. This mechanism is as simple as the example. These benefits are achieved because of the centralized network mechanism for tracking the flow of the transaction. The DPoS would benefit more by using a UTXO or transaction-based model because it is more of a decentralized network. Though the account-based model has got its own advantages, it is suitable for consensus, which is inclined towards centralized infrastructure. Still, most cryptocurrencies in DPoS based platforms are smart contracts and require the data model to be an account-based one, and there is no hard and fast rule to be stuck with one particular data model. The choice must make depending on what the application requires and the type of users that it shall serve.

3.3.7 Communication Model

The DPoS communicates with its peer nodes in asynchronous communication. The asynchronous communication does not have an upper bound on message delay. One might think that the DPoS consensus mechanism requires a predetermined time set for the exchange of messages between the nodes as the DPoS's delegate election, block creation time are all time-sensitive, but one advantage in an asynchronous model is the node can leave the network anytime and rejoin whenever it wants. The communication model is of three types synchronous, asynchronous, and partially synchronous. All these models involve the sending and receiving of messages in the network. When there is a

time duration associated with the network, then it is known as a synchronous network—a specific limit bounds the time limit. In the case of an asynchronous network, it is not bound by any time limit. The partially synchronous network, the system remains in the asynchronous state for a specific time limit, after which it changes into the synchronous network. The synchronous model is governed by an upper time limit bound and upper bound for the speed limit as well. In the asynchronous network, there is no upper bound on the speed and the time. It usually takes an arbitrary duration of time to receive and respond to messages.

3.3.8 Electing Miners

Electing a delegate in DPoS is close enough with the Proof of Stake, both ate stake based. All platforms based on DPoS have a standard way of electing a delegate to create blocks, although there are some attributes that vary like the block creation time, election duration time, round time, initial supply and inflation rate. There are a varying number of validators in DPoS depending on the application, and the validators are elected by the token owners. The number of votes a delegate receives is proportional to the token owned by the voter. The elected delegates go through a process in rounds to create a block when a block is created, other delegates are required to validate the block to reach consensus. The delegate who created a valid block is rewarded with the block reward. In most of the DPoS based platforms, a minor stakeholder (owning fewer stakes) can also be a part of the network and select the delegate. While some cryptocurrency based on DPoS require the user to have enough stakes to be a part of the network but another option of delegating their stakes to another member, in order to accumulate the stakes is possible. If that delegate receives rewards, it is shared proportionately.

3.3.9 Energy Savings

In delegated Proof of Stake, the Energy consumption is not high as the consensus mechanism is based on stake based. The energy consumption in DPoS is negligible. The energy consumption is not that high when compared to Proof of Work, as this is mainly because, in DPoS, no high computational power is required to solve resource-intensive puzzles. Since the election process is all stake based, and there is no need to solve any hash value from a nonce, there is no need for high computational power. Hence it is more energy-efficient, and any cryptocurrencies involving this consensus algorithm are more reliable and efficient in the long run. The verdict is that the energy consumptions in DPoS are better than PoW, but still, it is considered partial on an overall basis.

3.3.10 Tolerated Power of Adversary

The tolerated power of an adversary is the ability to withstand when an opponent attacks the network. When DPoS is compared with PoW, PoW's security is better than DPoS as the number of validators in PoW is high, but the validators or delegates in DPoS is very less like 21 in EOS. In contrast, the tolerated power of the adversary is better in DPoS than in Pow. Tolerated power of the adversary needs to be high in any consensus

algorithm. Usually, a certain percentage of network power is used to attack the security of the network. This tolerance value should be high, and it is less than a 51% stake in the delegated Proof of Stake, which is better compared to Proof of work and PBFT. The highest tolerated power of the adversary is around 51 %.

3.3.11 Transaction Fee

A transaction fee is applicable to all the miners in the DPoS consensus algorithm mechanism. In this algorithm, the transaction fee is in the form reward; it acts as a nodal reward. Whenever a block is created, the delegates are incentivized for their trustworthiness. The transaction fees are also not provided independently; it is given as a part of the whole transaction process. DPoS uses the transaction fee as a reward because it is obtained from the stake of the miners by building the blocks for each network.

3.3.12 Block Rewards

The elected delegates are given block rewards when they create a block and get validated and added to the block. In some cryptocurrencies, based on DPoS, the initial supply is utilized to pool block rewards. This makes DPoS far better than PoS, as PoS does not involve any block rewards. Block rewards serve as an incentivized to the delegates for their trust and contributions towards the network. By doing block rewards, it can considerably prevent security attacks from within the network.

3.3.13 Communication Complexity

The communication complexity of DPoS is quite similar to the PoS as both are stake based consensus. Networks that use DPoS consensus algorithm is Ouroboros. It is implemented by Cardano for cryptocurrency. The selection of a leader is random in DPoS. The process-id is performed by obtaining the broadcast information for the coin tossing. The process is complicated, but if it outlined for a single leader, it helps. The distribution tree helps understand the properties of CG, CP, and CQ. This process is done from the schedule of the block tree (encoding process). This encoding process is also known as the analysis of characteristics strings. The output obtained displays a string of length k with probability neg(k). The system consists of at least one honest leader in each round of execution.

$K = \Omega\alpha(k)$
$\tau = 1-\alpha$ (as the system has at least one honest leader in each round of execution)
where the distribution string depends on α.

3.3.14 Verification speed

Verification speed is the total duration required to compute all the validation process. The verification speed is less than 100 seconds in DPoS. The blocks are verified by a specific

number of validators, so the process of verification is low due to two reasons; one is fewer transactions, and secondly, block production is less due to limited elected delegates who produce blocks. The number of transactions is indirectly connected to the verification speed. Hence at the beginning, the verification speed is comparatively higher to the subsequent transaction. The verification speed in the Delegated Proof of Stake consensus algorithm is less than 100 seconds, is considered beneficial compared to the Proof of Work algorithm. When the verification speed is high, it ultimately reduces the entire transaction process as well.

3.3.15 Throughput

The DPoS is considered to deliver high throughput, which in numbers is less than 1000. The throughput time is vital in knowing how fast the transaction is confirmed in the network. Transaction throughput is defined as the number of transactions per second. It is the rate at which the transactions are completed in a specified time period. It is can also be represented as the total committed transactions divided by the total number of seconds at the number of committed nodes. Just like Proof of Stake, the block time is lesser in the Delegated Proof of Stake consensus algorithm, and the block size is much larger so that the efficiency increases. This particular mechanism makes the DPoS have high throughput time. The blockchain work is a function of the throughput and the network size. Hence the throughput transaction is a crucial parameter when there is a number of pending transactions to be performed by the network. [45] [50]

3.3.16 Block creation time

The Block creation time in DPoS is high, as the verification process is low, it does not take the block to get validated and reach consensus. The block creation time is the time between the moment the blockchain transaction is provided for the network to the moment it has been confirmed. In simpler words, it is the time taken for the participant to wait while the transaction is obtained and confirmed. Once the block is confirmed, the block is created. The block creation speed is high in DPoS and PoS compared to another consensus algorithm. In Proof of Work, the block creation changes according to the difficulty level, but that is not the case in Proof of Stake; hence its speed is much higher. [45]

3.3.17 Scalability

The scalability in DPoS is strong. Scalability forms an essential role in the decentralized network. It refers to the ability to expand the system by meeting the ongoing requirements. A decentralized network needs to achieve the transactional throughput of an expanding network. The different solutions for enhancing the scalability of the network are by developing the consensus and the data structure, modifying the size of the block, and developing second layer solutions. When the block size is modified/increased, it will result in large capacity and reduce network congestion. Nevertheless, the increase in block size will also result in more transactions to take place in a shorter interval of time.

The increase in the number of transactions will lead to an eventual delay in the confirmation of a transaction or even block them. The scalability increases with network growth. DPoS allows scalability in the block confirmation time as it does not have any computational problems to solve but not in a significant manner. [33]

3.3.18 51% Attack

The 51% attack is not possible in DPoS, as it would require the attacker to own 51% of the tokens, which would be a considerable amount to obtain, unlike Proof of Work, where 51% would mean computing the puzzle of the network. Moreover, in DPoS, for a 51% attack, the attacker is compelled to buy 51% of the stake where the price increases as the tokens are bought. It is complicated to attack a DPoS network as everything is public. Once the whole network knows that a particular address is buying many tokens, it is considered as a warning, and the attack is stopped even before it could happen. Even if the attack takes place, the value of it reduces in the network, making it a loss eventually. The tampering in the network would result in the loss of the attacker. Hence there is no benefit in attacking a DPoS network.

3.3.19 Double spending attack

It is evident that if a 51% attack is not possible in DPoS, then the double-spending has no chance to occur. The Double spending attack is when the same digital token/coin is used to duplicate another transaction in the block creation. The literal meaning is spending the same money twice for different transactions. Most of the attackers would encounter double-spend at some point in their process. The attacker makes an initial transaction and then reverse it to complete another transaction. It can be easily identified if the transaction takes place in the same branch, but attackers usually do it in another branch by conflicting the initial transaction. This attack would be possible only if the attacker holds the highest share, which would be way too expensive in the first place. This attack is considerably prevented in the DPoS consensus algorithm. The attack does not work on this algorithm because it is irrational for a high stakeholder to waste all the resources in stake on every chain of the network. Similarly, it is useless in attacking and risking their investment. It would result in the attacker losing all the stake. [52]

3.3.20 Byzantine Fault Tolerance

The DPoS is 50% tolerant of the Byzantine fault Tolerance. Byzantine Fault tolerance is derived from the Byzantine General problem. It is the ability to resist a certain level of failure in the system. The system has a certain tolerance level to the failures in the nodes without affecting the whole network. The tolerance level applies to situations where the messages are not correctly sent, or it takes time to send the messages. 50 % tolerances indicate that in the network, half of the validators should be honest for the system to work efficiently. The Byzantine fault tolerance is concerned with the security of the network. [53]

3.3.21 Summary of Metrics

METRIC	DESCRIPTION	DPoS
NODE IDENTITY MANAGEMENT	The information/data regarding the transaction available for public use, private or consortium	Public
DATA MODEL	Handling of information by the blockchain. The different types are transaction-based, account-based and key-controlled	Account based
COMMUNICATION MODEL	The model through which information is passed through the network. The different types are synchronous, asynchronous and partially synchronous	Asynchronous
ELECTING MINER BASED ON	The working mechanism of how the block creator is selected.	Stake owned
ENERGY SAVING	The energy consumption during the whole process	Partial saving
TOLERATED POWER OF ADVERSARY	The percentage level required in the network power to attack the security of the system	Less than 51% validators
TRANSACTION FEES	The fees generated whenever a new block is created	Provided for all witnesses
BLOCK REWARD	The coins obtained from creating a block	Provided for elected witnesses
VERIFICATION SPEED	Total duration required to complete all validation process.	Less than 100 seconds
THROUGHPUT	Number of transactions per second	Less than 1000
BLOCK CREATION TIME	Time duration to obtain the confirmation of transaction	High
SCALABILITY	Ability to expand the system by meeting the ongoing requirements	Strong
51% ATTACK	The attack done by the 51% network power holder	Does not occur
DOUBLE SPENDING	The attack by duplicating the transaction for new block creation.	Does not occur
BYZANTINE FAULT TOLERANCE	To resist certain level of failure in the node	50%

3.4 Practical Byzantine Fault Tolerance

Practical Byzantine Fault Tolerance (PBFT) is an asynchronous consensus algorithm. This algorithm was primarily used in distributed computing and eventually found its way into the blockchain consensus algorithm family. Like many other consensus algorithms, PBFT protects from system failures by a collective decision-making process that includes both healthy and faulty modes. The Byzantine army problem can best explain the PBFT algorithm. Byzantine army generals camped across the enemy territory can communicate with one another only via messenger. Some generals may be traitors and decide to prevent the loyal generals from reaching on an attack plan. The objective now is that loyal generals must decide on an attack plan and doing so would need a majority of their distributed army to attack at the same time. The generals must devise an algorithm that should guarantee that all generals agree on the same attack plan, and also a small number of treacherous generals would not cause the plan to fail. This loyal and traitor general analogy is to be applied for healthy and faulty nodes in the distributed system.

There are two types of node failures. One is where the node stops operating and fails, and another is a random-node failure. Some of the random node failures are:

- Return result failures
- Incorrect result failure
- Misleading result failure (malicious)
- Inconsistent result failure (sending different results to different nodes)

3.4.1 PBFT Working:

PBFT provides a state machine replication that could work when faulty nodes are part of the system. PBFT has a leader node and secondary nodes. When there is a primary node failure, any eligible secondary node in that system could become the primary node. Using the majority rule, a consensus is reached by all the healthy nodes. PBFT works on the basis that the maximum number of faulty nodes should not be higher than or equal to one-third of the participating nodes in the system.

PBFT utilizes message authentication codes (MACs) for authenticating all the messages it passes between the nodes. MACs use symmetric cryptography, and they can be computed three times faster than traditional signatures. Therefore, the PBFT is relatively faster in the vote-based category of consensus algorithms.

3.4.1.1 Normal Operation

There are four phases: pre-prepare, prepare, commit, and reply. The pre-prepare and prepare are atomic requests sent in the same view even when the leader node becomes malicious. The prepare and commit phases assure that committed requests are across all views. In the below figure, replica 0 is the leader node and replica 3 is a malicious node.

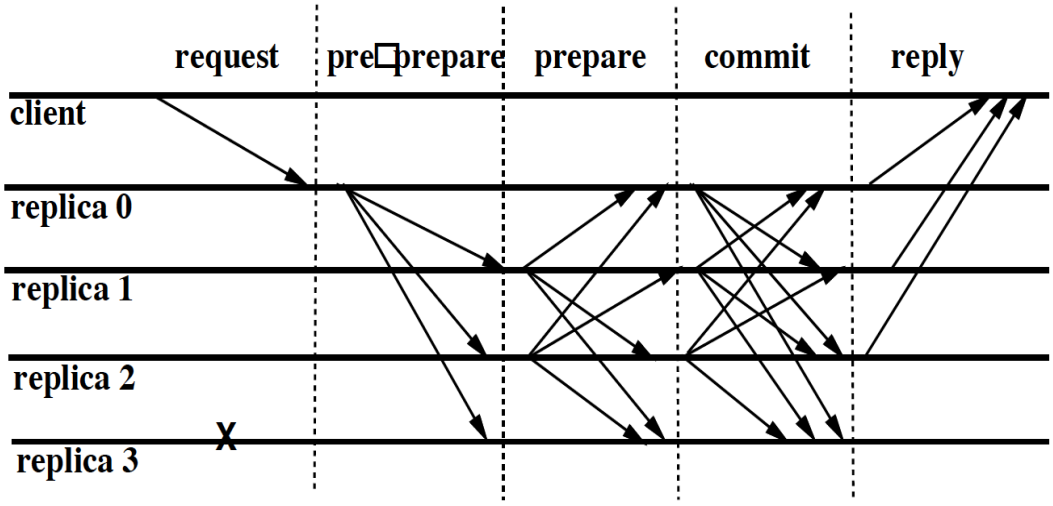


Figure 37PBFT Replica Communication [61]

All of the replica has a service state and an integer value to provide the replica's current state or view.

When the leader (replica 0) receives a service request from the client, a sequence number is allocated to the request. A pre-prepare message containing the sequence number is atomically multicasted. The prepare and commit messages sent in the other phases also contain the sequence number. The replicas only accept one of the messages if it is in a view where it can verify the veracity of the message. Also, a backup accepts the pre-prepare message only if it doesn't contain any other view and sequence number. The backup then further multicasts that message to all replicas and adds it to the log.

Now all the replicas accept the messages until it possesses a quorum certificate, which has the sequence number, view, and request number. Quorum is defined as the minimum number of nodes needed for the network to run properly and make valid decisions[4]. Quorum comprises of the honest nodes. Once the replica prepares the request, it will have the prepared certificate. After this event, the replicas concur to order for requests in the same view. This arrangement guarantees that it is impossible to get prepared certificates in the same view, sequence number, and for different requests.

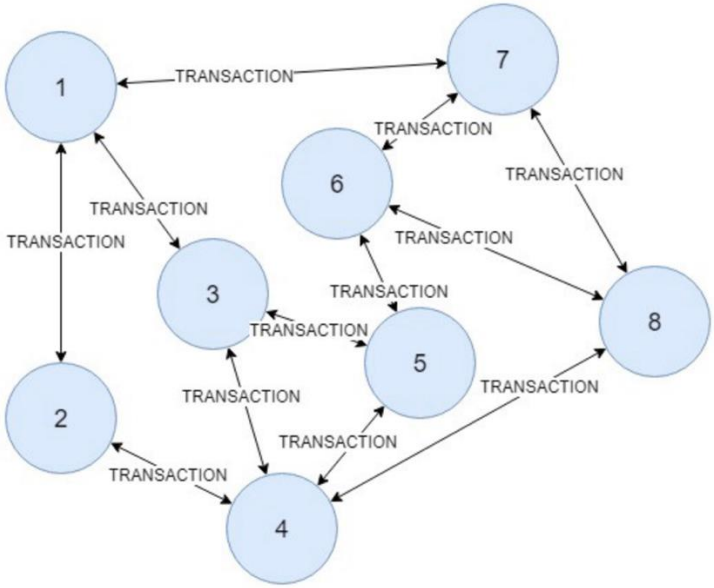
In the commit phase, all the replicas atomically multicasts the prepared certificate and logs it. Every replica receives messages until it possesses the quorum certificate with 2f+1 commit message. The final message is called the committed certificate. Once the request is committed, there is a guarantee that request originated from a quorum. The request is addressed successfully when ‘f+1’ replies from different nodes are sent to the client that are same, where m is the maximum number of faulty nodes allowed.

3.4.1.2 View Changes

The view change protocol is responsible for the PBFT's ability to exchange certificates between the replicas. It has the same communication pattern as traditional BFT except that backups send acknowledgments to the primary for every view-change message they receive from a different backup. These confirmations are used to prove the authenticity of the view-change protocol messages in the new-view certificate [62].

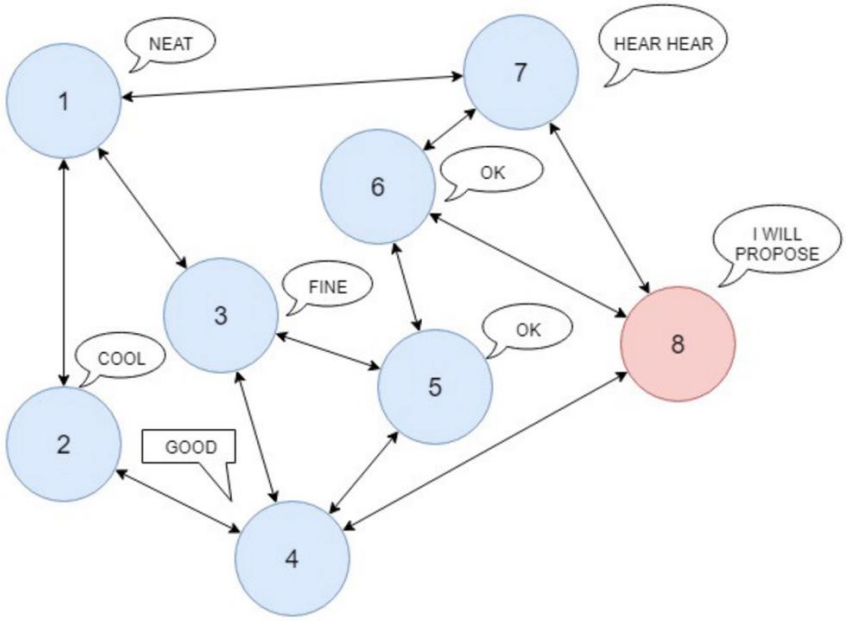
The idea behind the view change protocol is for all non-malicious replicas to cooperate and reconstruct weak certificates that corresponds to any prepared certificate that might have been accumulated by some non-faulty replica in a previous view. This is achieved by having replicas include information on view-change messages about pre-prepare, prepare, and checkpoint messages that they have transmitted in the past.

This section will illustrate the algorithm diagrammatically for better understanding:



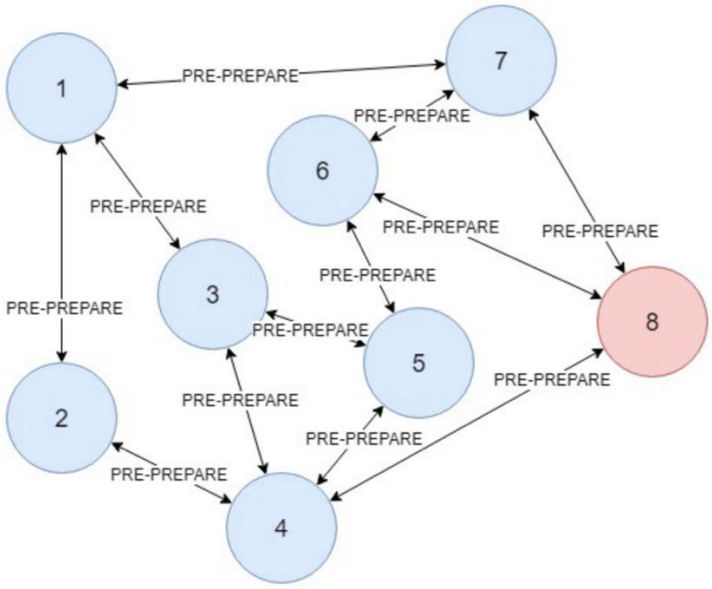
Before a new round begins, transactions are broadcasted among nodes so that all the nodes have the same transactions in their pool. After a sufficient number of transactions in their pool, these nodes start a new round.

Figure 38 PBFT Illustration 1 [61]



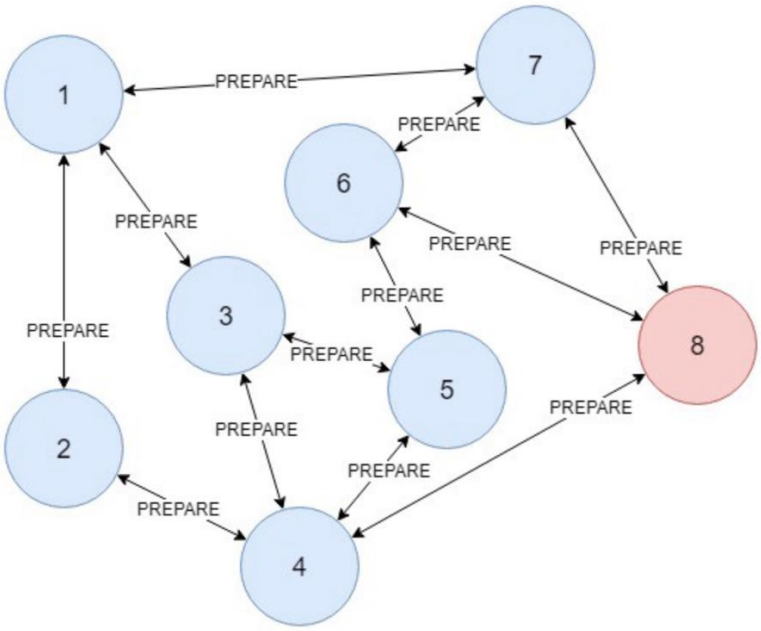
A proposer is chosen in a round-robin fashion. Node 8 becomes the proposer and rest of the nodes agree upon on it. The proposer sends a PRE-PREPARE message and each node enters PRE-PREPARED state.

Figure 39 PBFT Illustration 2 [61]



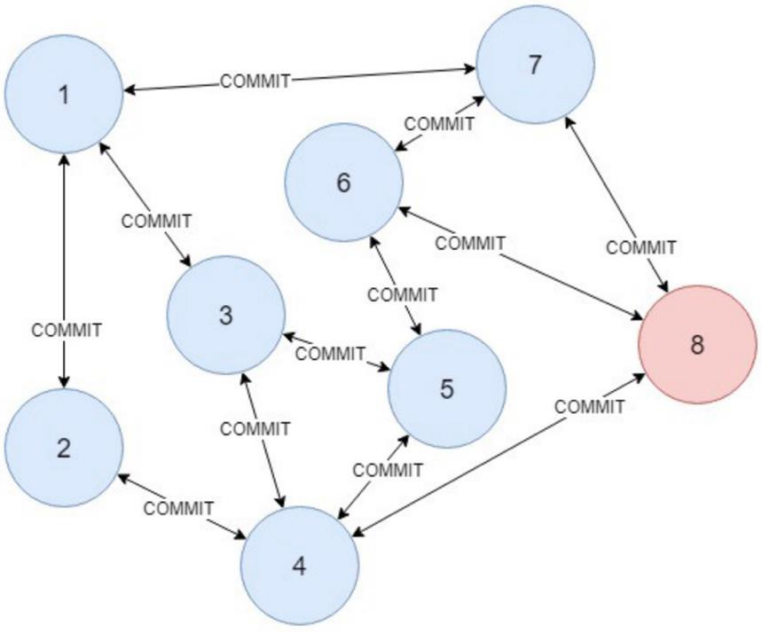
The proposer broadcasted a PRE-PREPARE message, which contains a proposed block. The rest of the nodes broadcast this message to other nodes.

Figure 40 PBFT Illustration 3 [61]



Each node sends a PREPARE message if they agree upon the proposed block. After 2F+1 such messages, nodes change state to PREPARED.

Figure 41 PBFT Illustration 4 [61]



Prepared nodes send COMMIT messages to each other, upon 2F+1 commits, nodes move to COMMIT state and add the block to the chain. After adding the block they move to the FINAL COMMITTED state.

Figure 42 PBFT Illustration 5 [61]

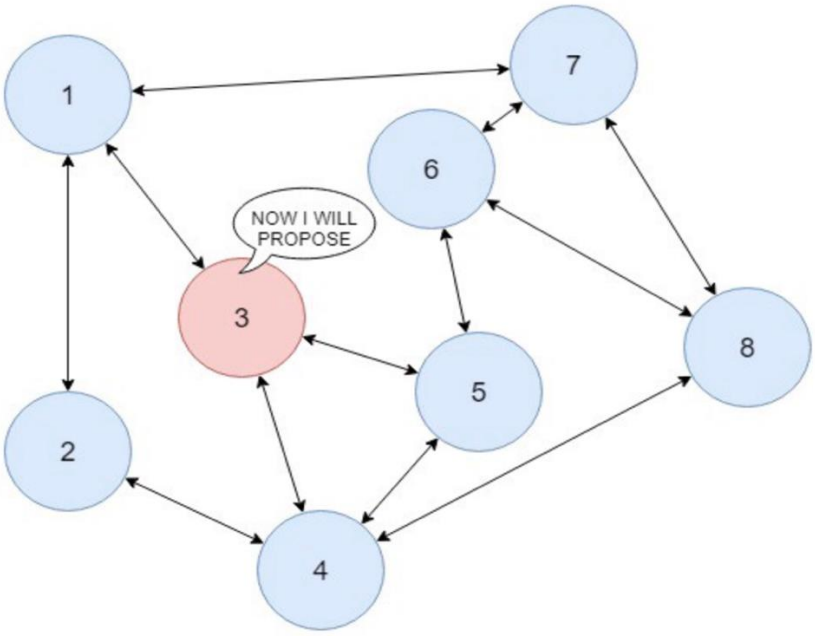


Figure 43 PBFT Illustration 6 [61]

3.4.2 PBFT - Mathematical Proof:

As mentioned earlier in PBFT, a system containing N nodes could tolerate F number of faulty nodes provided N=3F+1.

Every decision in a Practical Byzantine Fault Tolerant system requires 2F+1 node approval, where F is the number of faulty nodes. The below calculation proves the above proposition, which are corollary of one another.

3.4.2.1 Liveness

Liveness is the ability of the distributed system to operate even in the presence of some errors. In the blockchain world, the system will continue to append new blocks even if some of the nodes fail.

3.4.2.2 Safety

Safety here in the distributed systems means the ability of the system to converge to a single decision. In a distributed system, node may fork into two or multiple forks. Safety of that distributed system will ensure that the network will finally emerge with a single decision across all the honest nodes even in the presence of faulty nodes.

Proof: Let us assume N nodes in a network, with f number of fault nodes, the Quorum size Q required for guarateeing Liveness and Safety is as per below working.

As defined earlier, Quorum is defined as the minimum number of nodes needed for the network to run properly and make valid decisions**.**

Liveness
For a network to avoid stalling, there must exist at-least one non-faulty node.
so, for liveness:

$$Q \leq N - f$$

Safety
A majority should be present to avoid forking (network splitting into multiple decisions). For a honest majority, the Quorum size is expected to be greater than half the number of total nodes.

Thus, for Safety:

$$Q > N/2$$

$$2Q - N > 0$$

By combing both the conditions we get,

$$N + f < 2Q < 2(N - f)$$

$$N + f < 2N - 2f$$

$$3f < N$$

$$N > 3f$$

$$\text{If } N = 3f + 1,$$

$$\text{Then } 2Q > 4f + 1$$

Or

$$Q > 2f + 1/2$$

therefore, for all byzantine failures

$$Q_{min} = 3f + 1$$

3.4.2.3 Advantages of PBFT:

i. **Energy efficiency**: The biggest advantage is that PBFT can attain distributed consensus without performing complex mathematical computations (like in proof-based algorithms). Zilliqa, a cryptocurrency utilizes PBFT in conjunction with Proof of Work-like complex hash solving rounds for every 100 blocks.

ii. **Transaction finality**: All the transactions do not require several confirmations after they have been committed and agreed upon. This is in stark contrast to the cases in PoW mechanism where every node individually has to verify the transactions before appending a new block to the blockchain where acknowledgment can take up to an hour depending upon the number of entities confirming the new block.

3.4.3 Node Identity Management

PBFT is a permissioned blockchains are private and typically operated by invitations with known identities, which means there is already an established trust between parties. This arrangement mitigates the requirement for a trustless environment like the proof-based algorithms and also allows the network to enjoy the benefits of the PBFT.

3.4.4 Data Model

Unlike proof-based algorithms, PBFT is not transaction or account-based setup. PBFT works based on key value model. The keys are specifically called Message Authentication Code (MAC). MACs could be computed three times in magnitude faster than digital signatures.

For example, a 200MHz Pentium Pro could take only 43 millisecond to generate a 1024-bit mod RSA signature of a Message Digest algorithm (MD5) and 0.6ms to validate the signature [61], whereas it takes only 10.3 s to calculate the MAC of a 64-byte message on the same hardware in the implementation. There could be other public key cryptosystems that generate signatures faster, e.g., elliptic curve public-key cryptosystems, but signature validation is slower [61] and in PBFT algorithm every signature is validated many times.

3.4.5 Communication Model

PBFT follows an asynchronous communication model. Despite following asynchronous setup, PBFT ensures safety and liveness. It is worth noticing that the algorithm does not depend on synchrony to give safety and also the resilience of this algorithm is nominal. 3f+1 is the minimum number of replicas that allow an asynchronous system to provide the safety and liveness properties when up to f replicas are malicious or faulty [61].

PBFT ensures requests are ordered consistently across all views. In fact, PBFT is the first state machine replication algorithm that can withstand byzantine faults in asynchronous systems.

3.4.6 Energy consumption:

All of the surveyed literature repeatedly state one statement that PBFT network with a reasonable number of replicas and broadcast transmit power results in significant energy savings. There is no research, or it is difficult to compute a number in terms of energy consumption like what is commonly published for proof-based algorithms. One important trend worth noticing is that the IoT market sees the application of the wireless PBFT as a viable algorithm owing to its relatively low power consumption. Also, as it is evident, based on other algorithm comparisons that electrical power consumption is significantly less for algorithms that does not perform mining operations.

3.4.7 Tolerated Power of Adversary

Though PBFT enjoys other benefits in terms speed, security and energy consumption, its tolerated power of adversary is 33.3%. PBFT can tolerate lesser than one third of the faulty or malicious nodes. This compared to other proof-based algorithms is significantly less except for Proof of work which has 25%.

3.4.8 Transaction fees

Since PBFT is a private block chain, there are no, or very low transaction fees involved in using the consensus algorithm. Typically, an enterprise using PBFT will not have any transaction fees associated with its usage but a semi private PBFT could have low transaction fees.

3.4.9 Block Reward

As mentioned in the previous section, PBFT is a private or permissioned that typically depend on enterprise resources. PBFT algorithm does not reward the client for making a request for consensus.

3.4.10 Communication Complexity

This complexity is calculated for a normal case operation of the PBFT, except that primary nodes remain unchanged at each command (i.e. block). For each new request from the client, there is a primary node that will produce the quorum and implemented throughout all nodes. To optimize bandwidth, only the Message Authentication Codes (MAC) is included in the messages. This requires that b bytes of the request to be sent to all n nodes, at a Reliable Total Order Broadcast cost of n^2 bytes, making through all of the PBFT phases. Assuming a fault-free execution, $(bn+n^2)\ \Theta\ (1)$ is the communication complexity and $\Delta\Theta\ (1)$ latency.

Note that the PBFT, is considered only for the normal case operation above. However, since the primary is arbitrarily designated, it could be faulty. In our case, the properties of the Reliable Total Order broadcast will ensure safety, but termination could be prevented. This is why, after a time-out, it will be the turn of the next designated primary to proceed for a new broadcast. The choice for the new primary is in a fixed round-robin mechanism, all $\alpha n = \Theta\alpha\ (n)$ malicious nodes may be leaders first and delay the block commit by the same factor. Therefore, the worst-case communication cost for PBFT is $(bn^2+n^3)\ \Theta\alpha\ (1)$ bits and $\Delta n\Theta\alpha\ (1)$ for latency.

3.4.11 Verification Speed

As discussed on the data model section, the verification speed of PBFT is one of the fastest, based on the research data provided for a typical number of nodes, the verification speed is less than 10s for a typical vote based on algorithm.

3.4.12 Throughput

Based on PBFT performance modeling done by the researchers, the throughput is found to be close to 2000 T/S (Transactions per Second). In general, the private blockchains have high throughput compared to public proof based blockchains.

3.4.13 Scalability

As mentioned in earlier sections, scalability refers to the amount of the transactions that could be processed concurrently and the block size that can be created by the nodes. PBFT is not scalable, since the communication overhead will increase, this is because every node must communicate to every other participating node to maintain the network security, which can rapidly grow into a high communication cost.

3.4.14 Sybil Attack

The traditional PBFT model is very vulnerable to a Sybil attack one participant can create and manipulate a large number of nodes in the system, thus compromising the entire

network. However, improvements to traditional PBFT such as combining with Proof of Work algorithms for every 100 blocks have greatly reduced the sybil attack vulnerability.

3.4.15 51% attack & Double spending

PBFT algorithm is not prone to 51% attacks. Since PBFT is a permissioned algorithm there is no reason for majority of the participants to control 51% percent of the network.

3.4.16 Summary of Metrics

METRIC	DESCRIPTION	PBFT
NODE IDENTITY MANAGEMENT	The information/data regarding the transaction available for public use, private or consortium	Private
DATA MODEL	Handling of information by the blockchain. The different types are transaction-based, account-based and key-controlled	Key value controlled
COMMUNICATION MODEL	The model through which information is passed through the network. The different types are synchronous, asynchronous and partially synchronous	Asynchronous
ELECTING MINER BASED ON	The working mechanism of how the block creator is selected.	Mathematical computation
ENERGY SAVING	The energy consumption during the whole process	Considerable energy saving
TOLERATED POWER OF ADVERSARY	The percentage level required in the network power to attack the security of the system	Less than 33.33% replicas
TRANSACTION FEES	The fees generated whenever a new block is created	No transaction is provided
BLOCK REWARD	The coins obtained from creating a block	No block reward is provided
VERIFICATION SPEED	Total duration required to complete all validation process.	Less than 10 seconds
THROUGHPUT	Number of transactions per second	Less than 2000
BLOCK CREATION TIME	Time duration to obtain the confirmation of transaction	High
SCALABILITY	Ability to expand the system by meeting the ongoing requirements	Weak
51% ATTACK	The attack done by the 51% network power holder	Does not occur
DOUBLE SPENDING	The attack by duplicating the transaction for new block creation.	Does not occur

BYZANTINE FAULT TOLERANCE	To resist certain level of failure in the node	33%

3.5 Proof of Activity

Proof of Activity is considered as a Hybrid model in the blockchain consensus algorithm. This algorithm is formed by the combination of Proof of Work and Proof of Stake consensus algorithm. The idea of Proof of Activity was initially proposed by four authors in a paper in 2014. The main aim of this consensus algorithm is to include the advantage of both Proofs of Work and Proof of Stake and limit the constraints associated with those consensus algorithms. Proof of Activity improves the security of the network and also has reduced storage space. All the transaction are appropriately done, and all the information regarding the transaction are recorded for future use. Large amount of energy consumption is done in the first part of the process where Proof of Work consensus algorithm is used compared to the second part of the process where Proof of Stake algorithm is used. Rewards are obtained in this mechanism as well along with the chance of getting selected as the block creators.

One of the main advantages of using Proof of Activity is it reduces attack in the network. It monopolizes the whole process making the system difficult to attack. Even if the attack takes place, it would be way too expensive for the attacker. For the attack to take place, a large part of the stack should be owned by the attacker, which in turn requires a large number of currencies. Attack handling is similar to the Proof of Stake mechanism. If the attacker is identified, all the coins related to the attack is deleted. The Proof of Activity algorithm makes the transaction genuine by reducing attacks and also helps the validators/participants to reach consensus. Here the process initially begins similar Proof of Work where a puzzle is solved using much computational power. The puzzle is a hash value, which consists of consecutive zero at the beginning. The number of zeros in the hash value is directly linked to the complexity of the puzzle. As a new block is created and added to the network, the complexity level also increases with it. One of the main reasons for using the Proof of Work consensus algorithm is to avoid the problem called the "tragedy of commons" where the security of the system is at risk as the miners/validator begin to think about their benefits and act against the security of the network. This potential risk will become evident once the incentives start to reduce, and the only form of reward is the transaction fees (comparable to Proof of Stake).

Here the block time of the Proof of Work algorithm is reduced, and then the Proof of Stake algorithm is added to the mechanism. The Proof of Stake consensus algorithm involves the creation of a block based on the stake. It is selected by the stake wealth, stake age/node age and the randomization using the hash function. In this consensus algorithm, the participants or the validators are selected by using the FTS algorithm. Unlike the traditional Proof of Stake consensus algorithm where one signature is required for the block creation here, multiple signatures are required. The process executes in a way such that it does not matter whether the validator participates or not, the stake means all the coins the validator has. This procedure makes the algorithm require several signs to create

a block. The process is started by the miners but is completed by the stakeholders in the network.
The limitations in both Proof of Work and Proof of Stake are proposed to reduce by using the Proof of Activity algorithm. It is basically used to combine the advantages of both consensus algorithm. The main issue associated with the Proof of Work consensus algorithm is the high computational power required to solve the hash function, then the amount of work required to be added to the network to validate the transaction, which cannot be done with less amount of work contribution—finally, the expensive hardware device used for the mining process. On the other hand, Proof of Stake consensus algorithm has the advantage of reduced energy consumption and uses less hardware as it does not require any computational work to be done.

The security of a network plays an essential role in any consensus algorithm. Though Proof of Work is susceptible to a 51% attack, it is considerably reduce using Proof of Activity algorithm. Unlike other algorithm, Proof of Activity consensus algorithm requires the signature of multiple participants before the creation of a block to add to the chain. The attackers would not gain any investment in such cases because, in order to attack the network, they might want enormous investment, which is highly expensive. When it comes to Proof of Stake number of assets needs to be blocked, but this is avoided in Proof of Activity where the funds are not all blocked. Thus, the security constraints are overcome by combining the benefits of PoW and PoS algorithm. [63] [64]

3.5.1 Mechanism

The Proof of Activity mechanism is as follows.

i. The process begins from the mining process (similar to Proof of Work). With the use of computational power, the randomly generated hash value is solved. The hash function complexity increases as more blocks are created.

ii. The new block obtained/mined is now directed to operate with the Proof of Stake consensus algorithm. The block created would contain a reward for the first validator (address) and header.

iii. As mentioned earlier, in Proof of Activity, multiple signs are required to create and confirm a block. The block created is validated by using the header information obtained in the earlier stage. For the multiple sign, a random group of participants are selected for validation.

iv. Similar to the Proof of Stake consensus algorithm here, the coin-age based selection is made for the validators. The coin-age based selection involves taking into consideration the product node wealth and the node age. The validator with the highest node wealth, which has it held for the most prolonged duration, is prioritized to sign

the block. The complete block confirmation is acquired when a group validator signs the block. Several validators are selected for this process.

v. The newly created block is added to the blockchain and notified/broadcasted to the whole network. All the transaction information is recorded in the network for identifying any frauds or attacks.

vi. If the selected validators for a block creation are not participating in the mechanism, then the process moves to another block. The new block would be the next winning block selected through Proof of stake mechanism of coin-age based selection. For this new block, another set of validators are selected to sign the block based on the cryptocurrency availability. Even if the block created does not have the minimum required number of signers but achieves the status of a complete block, the process still continues, unlike the previous scenario.

vii. All the validators, including the first, would be provided with rewards in the form of transaction fees for each block created. [65]

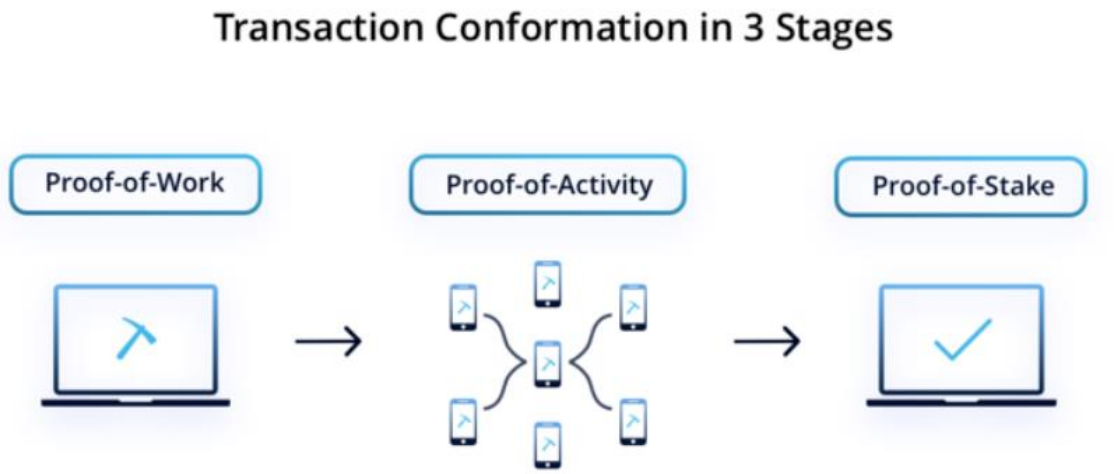


Figure 44 Proof of Activity [66]

In the Proof of Activity, the process begins with the miner preparing to solve the hash value using more computational power. An empty block with header, index and code for the section is created. The header of the empty block should match the complexity level. If the complexity standards are achieved, then it is sent to the network for further steps. Here the entire process is not done similar to the Proof of Work, hence the entire block transaction is not done using the Proof of Work consensus algorithm. The next step is to sign the empty bock through the PoS algorithm. Here the miners are called the stakeholders. The topmost part of the block is a distributed network where all information

regarding the transaction is shared with the miners/validators and making the system less volatile to risk. All the tokens in the process are averaged in the network to compensate for the computational power.
The various properties of the Proof of Activity consensus algorithm are as follows; the data protection part is the initial part of the process where information about the previously validated blocks is present for reference. It does not include information about all the blocks in the network. This feature reduces network congestion in terms of data storage. The main reason for permanent data exchange in Proof of Activity consensus algorithm is to minimize the network congestion when information is downloaded regarding the network. Another feature of this algorithm is that the network can be controlled by many validators/stakeholders because a single member cannot control the whole network; it would be impossible to do so.

The project Enecuum uses the Proof of Activity algorithm. Here the high level of distribution is achieved by providing all the participants/validators with the right for confirmation in the block transaction. Here the miners find empty blocks of different sizes and compute the puzzle. The participants who complete the computation immediately proceed to the next level of the transaction for confirmation. Usually, the confirmation is done by 64 random node owners. The next step is the PoS mechanism, where the participant with the highest node wealth and the time duration becomes the block creator. Various validations are executed with the information provided by the network, after which the confirmation is sent to the participant. This project is not yet widely used, it is in its initial stage of mining, after which various audits need to take place before the next step is executed.

For instance, if a certain number of participants are not present during the transaction(offline), then further mining process takes place, and different validators are selected as stakeholders. After this process, the complexity of the process is increased. It increases the difficulty level as new blocks are created. The communication in the Proof of Activity is two levels of execution. This makes the block creation process of the Proof of Activity consensus algorithm more involved in the process rather than just the Proof of Work algorithm.

The stakeholders are selected based on a specific selection process; for this to happen, the Proof of Work part of the mechanism should be able to generate new addresses instantly. By doing so, the cryptocurrencies can be split among the different address. The Proof of Work mechanism is crucial in making the system convergent. A convergent network would be decentralized and also achieve a specific result (winning branch) without introducing any failures in the system.

The mechanism of Proof of Activity allows utilizing the 160- bit security. When the need arises, and the value of output is higher than a certain threshold, it can be used to increase the block size by 160 bits. It is crucial that the participants in the network should be online to receive reward/incentives rather than Proof of Stake where, when the participants are not online, it would lead to double-spending attacks. This disadvantage is reduced in the Proof of Activity consensus algorithm. This algorithm has less overhead when it comes

to communication in the network as it does not have checkpoint blocks to increase the reliability of the network. The Proof of Activity process consists of only reward protocols, not any punishment/penalty protocols when the participants in the network go offline, which are available in other consensus algorithms. When it comes to the Proof of Stake algorithm, this penalty is included, which makes most of the stakeholders to take part in the process. The simple definition is that reward cannot be obtained when investments are not made. Though it is an advantage in Proof of Stake when it comes to Proof of Activity, it is similar to PoW, where the obtained rewards would be utilized to compensate for the hardware cost and computational power. The penalty protocol in Proof of Activity is the loss of the coins.

When it comes to reward protocols, the Proof of Activity uses a different mechanism to the Proof of Work. Here many online nodes can be created at the same make the transactions more and decreasing the reward/incentives. Hence some of the stakeholders re-transmit all the information to obtain the rewards. At the same time, some stakeholders value their stake more than the rewards and hence do not re-transmit their data. The Proof of Activity consensus algorithm can provide the reward to specific nodes that re-transmit with transaction fees.

3.5.2 Node Identity Management

Proof of Activity is a Public node identity management. The public node identity management is also known as permission less network. Here all the transaction information is available in the public ledger. The information about the past transaction are broadcasted but not all the participant information of the network. The data protection part is the initial stage of the transaction process and shows the information regarding the previous blocks. This limited information reduces the network congestion in the system when data is to be downloaded during transaction. When it comes to Proof based consensus algorithm, the node identity management is Public. In the Proof of Activity working, after the transaction is verified, the aggregate data is broadcasted or displayed in a public ledger. All of these algorithms are used for the network are public. The membership control is defined by type of the blockchain. The three different type of blockchain are private, public and consortium. This part is mostly defined at the design part of the network and differs from one application to another.

This network allows anyone to take part in the block creation/validators and modify it. Every modified data is being updated and displayed to the public ledger. After every transaction and when each node is validated by the validators it is broadcasted. This transparent approach might result in security issues for certain scenarios. Usually, the blockchain network is a distributed, decentralized and public. The public blockchain is also referred as the permission less or unpermissioned network. This is mainly used in Proof of Activity because of its process of displaying the transactions. Here everyone can read the transaction data but only certain users can do the validation.

3.5.3 Data Model

The data model used in the Proof of Activity is account based model. As it is a hybrid model combining the mechanism of Proof of Work in the initial stage and the Proof of Stake mechanism in the final stage of the process, influences the network model to be account based. Here the final transaction confirmation is based on the stake the validator presents, hence follows the same mechanism as Proof of Stake. The account-based model works similarly to the standard banking model of balance management. The mechanism is effortless and can be explained as follows, consider participant 1 has ten tokens, and participant 2 has ten tokens. When participant two wants to send the token to participant 1, it is subtracted from the former account resulting in zero balance in the account. Now participant 1 transactions 10 to participant 2, resulting in equal tokens for both participants. This mechanism is as simple as the example. The account-based model is used in the Proof of Activity mechanism.

As Proof of Activity mechanism is still not use, detailed explanation about its data model specifically is not yet available. But the working is similar to Proof of Stake examples as it is hybrid model of consensus algorithm. The account-based model is much simpler because most of applications are decentralized/distributed. In Ethereum Proof of stake, Private key-controlled user accounts and contract-code controlled accounts are used. These two accounts are crucial in determining why the account-based model is preferred over the Transaction-based model. The main reason for using the account-based model is the use of smart contracts; if the transaction-based model is used, it will limit the use of smart contracts. When the Proof of Work consensus is used in the initial stage of the process it occupies certain amount space due to its computational workload, only the remaining space is allocated for the rest of the transaction process. The account gets debited if any changes are made, like adding another message by changing the internal storage. These changes should be limited for the extensive use of the data storage space. Every account must have balance, code-storage space for other addresses. The process starts from the sender, and if the receiving end has the code to run, the process continues. The changes affect the whole system and all accounts as new accounts are added to the network.

The advantages of using the account-based model are easy tracking of the transactions, and it also prevents the double-spending attack. These benefits are achieved because of the centralized network mechanism for tracking the flow of the transaction. The different advantages associated with the account-based model are as follows, this model increases the space available compared to the transaction model. Here as more signatures are required for one block creation it occupies more space but less than the transaction-based model. This space-saving is crucial for the Proof of Activity mechanism because it is a sophisticated platform. The added advantages are simplicity and familiarity. The probable disadvantage in this model is its limitation in terms of scalability. This limitation becomes a bigger concern when it comes to a broader industry. [9]

3.5.4 Communication Model

The communication model adopted in the Proof of Activity is asynchronous communication model. The hybrid model involves a complicated platform and when the messages are sent to each node it might take an arbitrary amount of time to receive the data about the confirmation of the transaction. This make the communication model an asynchronous model. The communication model is of three types synchronous, asynchronous and partially synchronous. A specific limit bounds the time. In the case of an asynchronous network, it is not bound by any time limit. All these models involve the sending and receiving of messages in the network. In asynchronous network there is no upper bound on the speed and the time. It usually takes arbitrary duration of time to receive and respond to messages. When there is a time duration associated with the network, then it is known as a synchronous network. The synchronous model is governed by an upper time limit bound and upper bound for the speed limit as well. In the case of a partially synchronous network, the system remains in the asynchronous state for a specific time limit, after which it changes into the synchronous network. In the Proof of Activity, the total duration is divided into three categories, namely pro-pose, pre-vote and pre-commit. This division of period makes the mechanism a weak synchronous protocol (asynchronous). [48]

As Proof of Activity mechanism uses an asynchronous network, it is difficult to achieve specific parameters described in the FLP impossible algorithm. No network can achieve all three parameters at the same instance; it changes according to the applications. In a consensus algorithm, it is difficult to achieve all three parameters of consistency, fault tolerance and availability. For instance, if it is a distributed network application, it is preferred to consider safety over fault tolerance. The concept described in the CAP theorem is about the consistency and availability. Generally, both consistency and availability cannot be achieved at the same time because, during a specific period, some messages are intended to be dropped during the process.

3.5.5 Electing Miners

In Proof of Activity, the miners are elected based on the hash function solving capability. The two most essential parameters involved are the randomization (hash value) and the stake age. All these parameters together are involved in electing the miners. But the main part is by solving the hash function for which the complexity increases as number of transaction increases. Generally, in this consensus algorithm, the blocks mined. Two methods of selection are done in Proof of Activity to select the leader. Randomized block selection and coin-age selection. The randomized block selection is concerned with the hash function, whereas the coin-age based selection is concerned with the stake of each participant. The randomized selection is done at the initial part of the process (similar to Proof of Work) and the coin-based selection is done at the later stage (similar to Proof of Stake). The validator whose stake is the highest and the node wealth is high would be made the block creator after solving the complicated hash function. Here the transaction fee is provided as the reward.

In the randomized block selection, the validators are selected based on the capability to solve the hash value and with a higher stake value. The hash value consists of consecutive 0's in the the input string. The number of 0's is indirectly linked to the complexity of the problem/puzzle. In the coin-based selection, the process involves the product of the duration (No. of days) the coins have been held and the stake value. The combination of both is crucial in selecting the miners for the network.

Once the block is allocated in order to obtain another block, the coin-age starts from zero so that domination by one significant stake can be avoided in the blockchain. Many validation processes take place during transactions to ensure security before signing that particular block, which is added to the chain. Once the node decides not to participate in the mining process, all its stake and transaction fees are returned. However, it is not done immediately; it takes a certain amount of time so that no fraud takes place in the node. [49]

3.5.6 Energy Saving

Typically, the energy consumed in cryptocurrency is way high (more energy required for gold mining), which will be considerably reduced in Proof of Activity mechanism. In Proof of Activity, the energy consumption is not that high when compared to Proof of Work. The Energy consumption in the Proof of Activity is not that high but still the use of mechanism similar to the Proof of Work in the initial stages requires energy. This energy consumption is due to the high computational power required to solve the hash function. This part of the energy consumption cannot be reduced as it is required to meet the network requirements. This is mainly because, in Proof of Activity, only the initial stage involves the need for high computational power to solve resource-intensive puzzles. Hence it is more energy-efficient, and any cryptocurrencies involving this consensus algorithm are more reliable and efficient in the long run. Here when this consensus algorithm is used it would reduce the energy consumption by a significant amount. The use of smart contracts helps in reducing the energy consumed during this whole process. Thus, there is partial energy saving in the Proof of Activity consensus algorithm. [49]

3.5.7 Tolerated Power of Adversary

The tolerated power of the adversary for the Proof of Activity consensus algorithm is 50% of the online stake. Tolerated power of the adversary needs to be high in any consensus algorithm. Usually, a certain percentage of network power is used to attack the security of the network. This tolerance value should be high, and it is around 50% online stake in the Proof of Activity, which is better compared to Proof of work and PBFT. The highest tolerated power of the adversary is around 51 %. The tolerance level is variable in the Proof of stack and delegated Proof of Stake but it is fixed in Proof of Activity and it is very good level of tolerance compared to other consensus algorithm.

3.5.8 Transaction Fees:

Transaction fees are paid to all the miners in the Proof of Activity consensus algorithm mechanism. Apart from all the miners the stakeholders selected for the later stage of the process are also provided with the transaction fees. These stakeholders are selected randomly so that each block could has sign from various validators. These stakeholders are considered the lucky stakeholders. Proof of Activity uses the transaction fee as a reward because it is obtained from the stake of the miners by building the coins/token for each block. In this algorithm, the transaction fee is in the form reward; it acts as a nodal reward. Here the transaction fees are collected whenever a block is created and is used to help incentivize the miner and make the blockchain growing. The transaction fees are also not provided separately; it is given as a part of the whole transaction process.

3.5.9 Block Reward

Block reward is not provided in this consensus algorithm because it uses the mechanism of Proof of Stake, which does not provide block rewards. Block reward consists typically of the coins obtained from creating each block. It is given to the node after the transaction is completed. Instead of the block reward the incentive is compensated in the transaction fees.

3.5.10 Verification Speed

Verification speed for Proof of Activity is not yet found because this hybrid mechanism is not yet applicable in the industry. The validation process plays a vital role in the mechanism because it prevents fraud by creating more blocks as it is simpler to create blocks in Proof of Activity consensus algorithm. Verification speed is the total duration required to compute all the validation process. The verification process is crucial at the receiving end as well, mainly because the generated hash value needs to be compared with the input hash value. The number of transactions is also indirectly connected to the verification speed. Hence at the beginning, the verification speed is comparatively higher to the subsequent transaction. The verification speed in the Proof of Stake consensus algorithm is less than 100 seconds and the verification speed in the Proof of Work is greater than 100 seconds. Among these two algorithm Proof of Stake is beneficial compared to the Proof of Work algorithm. As Proof of Activity is hybrid model combining the above two mechanism it combines the advantages, making the verification speed better. When the verification speed is high, it ultimately reduces the entire transaction process as well.

3.5.11 Throughput

Throughput for Proof of Activity algorithm is not yet found as the concept is not completely developed for industry use. It is can also be represented as the total committed transactions divided by the total number of seconds at the number of committed nodes. The throughput time is vital in knowing how fast the transaction is confirmed in the network. Transaction throughput is defined as the number of transactions per second. It is the rate at which the transactions are completed in a specified time period. The block time is lesser in Proof of

Stake consensus algorithm, and the block size is much larger so that the efficiency increases. This particular mechanism makes the Proof of Stake have high throughput time. But in case of Proof of Work the throughput is very low. The combination of both results in the throughput of Proof of Activity consensus algorithm. The blockchain work is a function of the throughput and the network size. Hence the throughput transaction is a crucial parameter when there is a number of pending transactions to be performed by the network. [45] [50]

$$Tx/s = \frac{Block_{size}}{Tx_{size} \times Block_{time}}.$$

3.5.12 Block Creation Time

The block creation time for Proof of Activity is high. The block confirmation time is the time between the moment the blockchain transaction is provided for the network to the moment it has been confirmed. In simpler words, it is the time taken for the participant to wait while the transaction is obtained and confirmed. Once the block is confirmed the block is created. The block creation speed is high in Proof of Stake compared to other consensus algorithms. In Proof of Work the block creation changes according to the difficulty level but that is not the case in Proof of stake hence its speed is much higher resulting in the high value altogether for Proof of Activity. [45]

3.5.13 Scalability

Scalability forms an essential role in the decentralized network. It refers to the ability to expand the system by meeting the ongoing requirements. A decentralized network needs to achieve the transactional throughput of an expanding network. The different solutions for enhancing the scalability of the network are by developing the consensus and the data structure, modifying the size of the block and developing second layer solutions.

The process of obtaining throughput by horizontal scaling is known as sharding. The simultaneous transaction of multiple shards becomes more effective than processing a single transaction/mining at a time. Each shard consists of its block history and state information so that individual transactions can take place. Each shard is related to the main blockchain so that all the information is up to date. The problem arises when there are way too many shards that might require further scaling making the network congested. Thus scalability increases with the network growth.

When the block size is modified/increased, it will result in large capacity and reduce network congestion. Nevertheless, the increase in block size will also result in more transactions to take place in a shorter interval of time. The increase in the number of transactions will lead to an eventual delay in the confirmation of a transaction or even block them.

Proof of Activity is strong as it includes the scalability of both Proof of Work and Proof of Stake. Proof of Stake facilitates sharding but has limitations when it comes to scalability

regarding the throughput of the network. Proof of Stake allows scalability in the block confirmation time as it does not have any computational problems to solve but not in a significant manner. On the other hand scalability in Proof of work is constrained by the network requirements. [33]

3.5.14 51% Attack

Unlike Proof of Work, where 51% would mean computing the puzzle of the network it is complicated to attack Proof of Activity and would result in loss. In Proof of Activity, it is not easy to make the 51% attack; it would require the attacker to own 51% of the tokens, which would be a considerable amount to obtain. Moreover, in Proof of Activity, for a 51% attack, the attacker is compelled to buy 51% of the stake where the price increases as the tokens are bought. Once the whole network knows that a particular address is buying many tokens, it is considered as a warning, and the attack is stopped even before it could happen. It is complicated to attack a Proof of Stake network as everything is public. Even if the attack takes place, the value of it reduces in the network, making it a loss eventually. The tampering in the network would result in the loss of the attacker. Hence there is no benefit in attacking a Proof of Activity network. When a specific attacker address is identified, it is blacklisted, and it would be tough for the attacker to repurchase the tokens. Once the address is identified, all the stake is deleted, and then the value of the tokens is increased, making it even more complicated for the attacker to initiate another attack.

It is not benefitable to attack the Proof of Activity algorithm even when four common cases of attack are considered. The four common cases that comes under the 51% attack are finality reversion, where the finality guarantee is broke by finalizing another block. The next case is invalid chain finalization where unavailable blocks are finalized, then liveness denial and finally censorship. The third case is completed reduced in the Proof of Activity consensus algorithm. When the validators stop confirming the blocks, those node weights are reduced by removing them. [51]

3.5.15 Double Spending Attack

Double spending attack does not occur in the Proof of Activity consensus algorithm. The Double spending attack is when the same digital token/coin is used to duplicate another transaction in the block creation. The literal meaning is spending the same money twice for different transactions. Most of the attackers would encounter double-spend at some point in their process. The attacker makes an initial transaction and then reverse it to complete another transaction. It can be easily identified if the transaction takes place in the same branch, but attackers usually do it in another branch by conflicting the initial transaction.
This attack would be possible only if the attacker holds the highest share, which would be way too expensive in the first place. This attack is considerably prevented in Proof of Stake consensus algorithm resulting in the prevention of this attack in Proof of Activity. The attack does not work on this algorithm because it is irrational for a high-stake holder to waste all the resources in stake on every chain of the network. Similarly, it is useless

in attacking and risking their investment. It would result in the attacker losing all the stake. [52]

3.5.16 Byzantine Fault Tolerance:

The Byzantine fault tolerance is not yet found for the Proof of Activity algorithm because the time duration is not known in this case which is crucial in finding the Byzantine fault tolerance. Byzantine Fault tolerance is derived from the Byzantine General problem. It is the ability to resist a certain level of failure in the system. The system has a certain tolerance level to the failures in the nodes without affecting the whole network. The Byzantine fault tolerance level in Proof of Stake is around 50% and the Byzantine fault tolerance for Proof of Work is 50%. The tolerance level applies to situations where the messages are not correctly sent, or it takes time to send the messages. 50 % tolerances indicate that in the network, half of the validators should be honest for the system to work efficiently. The Byzantine fault tolerance is concerned with the security of the network. [53]

3.5.17 Summary of Metrics

METRIC	DESCRIPTION	PROOF OF ACTIVITY
NODE IDENTITY MANAGEMENT	The information/data regarding the transaction available for public use, private or consortium	Public
DATA MODEL	Handling of information by the blockchain. The different types are transaction-based, account-based and key-controlled	Account-based
COMMUNICATION MODEL	The model through which information is passed through the network. The different types are synchronous, asynchronous and partially synchronous	Asynchronous
ELECTING MINER BASED ON	The working mechanism of how the block creator is selected.	Solving complex hash function
ENERGY SAVING	The energy consumption during the whole process	Partial saving
TOLERATED POWER OF ADVERSARY	The percentage level required in the network power to attack the security of the system	50% of online stake
TRANSACTION FEES	The fees generated whenever a new block is created	Provided for all miners and lucky stakeholders
BLOCK REWARD	The coins obtained from creating a block	No
VERIFICATION SPEED	Total duration required to complete all validation process.	Not found
THROUGHPUT	Number of transactions per second	Not found
BLOCK CREATION TIME	Time duration to obtain the confirmation of transaction	High
SCALABILITY	Ability to expand the system by meeting the ongoing requirements	Strong
51% ATTACK	The attack done by the 51% network power holder	Does not occur
DOUBLE SPENDING	The attack by duplicating the transaction for new block creation.	Does not occur
BYZANTINE FAULT TOLERANCE	To resist certain level of failure in the node	Not found

3.6 Other Proof Based Consensus Algorithms

3.6.1 Proof of Importance

The proof of importance is a further development of proof of stake. The Proof of Importance is a novel algorithm that assigns ratings to account's importance in a network based on the network's theory design. It came into limelight due to the NEM. In the Proof of Importance, a node is added to the blockchain network depending on the harvesting mechanism. The more harvesting is done on a node, the more chances for the node to get qualified to be a part of the network chain. The node gets a transaction fee that the validator can collect as a reward, and this is done in exchange for the harvesting. To be qualified to harvest, the user must have at least 10,000 XEM tokens. The Proof of Importance was able to fill the pitfalls found in the Proof of Stake, in the Proof of Stake, the richer gets more share of money while the validators are not rewarded well. For example, if a wealthy member owns 20% cryptocurrency, then they can mine all the 20% blocks from the blockchain network. The Proof of Stake was promising to wealthy users who own majority stakes [67].

The function of the PoI is similar to the PoS, the nodes need to vest money to be eligible for creating blocks, and the selection process for the nodes that creates the block also depends on the node that has more coins. In Proof of stake, the score is the total vested amount, but in Proof of Importance, the score includes more variables.

3.6.1.1Characteristics of Proof of Importance

3.6.1.1.1 Vesting

The vesting or harvesting is the most interesting feature in blockchain consensus algorithms. In order to qualify to harvest, a member needs to have at least 10,000 XEM, and the PoI score must display the history of harvest, the algorithm also considers the time period the member has the coins.

3.6.1.1.2 Transaction Partnership

The Proof of Importance algorithm will reward users who make transactions with other NEM account holders. The algorithm will consider the two as partners; at the same time, it can also find if there are any fraudulent activities during the course of the partnership.

3.6.1.1.3 Scoring system

The transactions by the users influence significantly on the PoI scores. The PoI algorithm scores are based on the transactions that happen in the last 30 days. This means the transactions have to be frequent and substantial for the scores to good.

3.6.1.2 Node identity management

The Proof of Importance uses consortium-based node management. The consortium is semi-private based node management, which has controlled user groups. Typically, the consortium combines both the features of public and private chains. The consortium would be more suitable in multi-level organizations, which is under a common industry and requires a common ground to depends on information. When it comes to any Proof based consensus algorithm, the node identity management is usually Public, but PoI's node management is different due to the mechanism of the algorithm.

3.6.1.3 Data model

The data model in PoI can either be transaction-based or account-based. The Transaction model is called the UTXO - Unspent Transaction Output scheme. The coins are stacked as unspent transaction output. There is always a spending criterion on the UTXO based models. The existing UTXO's are utilized during the transaction, and new UTXO's are created to replace them. The account-based model works similarly to the standard banking model of balance management.

3.6.1.4 Communication model

The communication model in PoI is asynchronous.The Proof of Importance communicates with its peer nodes in asynchronous communication. The asynchronous communication does not have an upper bound on message delay.

3.6.1.5 Electing miners

In Proof of Importance, electing miners is of high priority, which is based on the node which has a greater number of coins, just like the PoS consensus mechanism. The nodes need to vest more coins to be eligible to create blocks, and a node is selected to create block proportional to the node's score.

3.6.1.6 Energy savings

The energy savings in Proof of Importance is high as it does not involve any mining activity. The electrical power in consensus algorithms is conserved, which does not have the mining mechanism.

3.6.1.7 Tolerated power of the adversary

The tolerated power of an adversary is the ability to withstand when an opponent attacks the network. The tolerated power of adversary in Proof of Importance is less than 50% of importance.

3.6.1.8 Transaction Fee

The transaction fee in Proof of Importance is given to transaction partners. The transaction fees awarded to the validator of the node for creating the block as block rewards.

3.6.1.9 Block Creation Time

The block creation speed in Proof of Importance is high as the verification speed of the creating blocks is quick, and the consensus is reached.

3.6.1.10 Scalability

The scalability in Proof of Importance is high; the scalability features in the PoI are similar to PoS. The increase in the number of transactions will lead to an eventual delay in the confirmation of a transaction or even block them. The scalability increases with network growth. PoI allows scalability in the block confirmation time as it does not have any computational problems to solve but not in a significant manner.

3.6.1.11 51% Attack

The 51% attack is not possible in PoI, as PoI does not involve computational power. The node with a high computing rate will have the ability to calculate the nonce value faster. Sometimes these backfires if a selfish node that has high computational power or rate than the total computational rate of all nodes combined can compromise the network leading it to a 51% attack [40]. In ideal conditions, to conduct a 51% attack is quite expensive, considering the massive network and owning high computational power. The Proof of Importance does not use computational power.

3.6.1.12 Double spending attack

There are no chances of a double-spending attack possible in PoI. If a consensus algorithm has no chance of a 51% attack, then the double-spending attack can be ruled out.

Note- There are some metrics that are not compared as they may be yet to be explored, or the consensus algorithm may not have been exposed to such attributes. The PoI has some metrics which are yet to be found like block reward, verification speed, throughput, Byzantine Fault Tolerance.

3.6.1.13 Summary of Metrics

METRIC	DESCRIPTION	PoI
NODE IDENTITY MANAGEMENT	The information/data regarding the transaction available for public use, private or consortium	Consortium
DATA MODEL	Handling of information by the blockchain. The different types are transaction-based, account-based and key-controlled	Transaction-based and Account-based
COMMUNICATION MODEL	The model through which information is passed through the network. The different types are synchronous, asynchronous and partially synchronous	Asynchronous
ELECTING MINER BASED ON	The working mechanism of how the block creator is selected.	Priority wise (High)
ENERGY SAVING	The energy consumption during the whole process	Considerable energy saving
TOLERATED POWER OF ADVERSARY	The percentage level required in the network power to attack the security of the system	Less than 50% importance
TRANSACTION FEES	The fees generated whenever a new block is created	Provided for all transaction partners
BLOCK REWARD	The coins obtained from creating a block	No Block reward is provided
VERIFICATION SPEED	Total duration required to complete all validation process.	Not found
THROUGHPUT	Number of transactions per second	Not found
BLOCK CREATION TIME	Time duration to obtain the confirmation of transaction	High
SCALABILITY	Ability to expand the system by meeting the ongoing requirements	Strong
51% ATTACK	The attack done by the 51% network power holder	Does not occur
DOUBLE SPENDING	The attack by duplicating the transaction for new block creation.	Does not occur
BYZANTINE FAULT TOLERANCE	To resist certain level of failure in the node	Not found

3.6.2 Proof of Luck

The Proof of Luck is consensus's mechanism is all the participants select a random number, and the one who picked the highest random number wins, that block that won is selected and added to the blockchain network. The PoL consensus is based on Intel's SGX model. As the random number selection happens in the SGX environment, it cannot be forged.

The node which selects the random lucky number is chosen, and the block created by the node goes into the chain. Each node will create blocks and assign a random number from 0 to 1 to the block header. This algorithm has high immunity to the double-spending attack as the probability of the attacker picking the lucky number to bring down the network is impossible. The Proof of Luck experiences similar problems to the Proof of Work like the node will lose the chance to lucky if the clock is not synchronized to the network's clock. The PoL requires high processing power as it has to run several numbers before reaching the lucky number [8].

3.6.2.1 Node identity management

The Proof of Luck needs to verify the identify the nodes that participate in the mining process; hence PoL uses consortium networks due to the reason that tracing the nodes in a public network will become complicated. The Proof of Luck uses consortium-based node management. The consortium is semi-private based node management, which has controlled user groups. Typically, the consortium combines both the features of public and private chains. The consortium would be more suitable in multi-level organizations, which is under a common industry and requires a common ground to depends on information [8] .

3.6.2.2 Data Model

The data model in Proof of Luck can either be transaction-based or account-based. The account-based model works similarly to the standard banking model of balance management. The Transaction model is called the UTXO - Unspent Transaction Output scheme. There is always a spending criterion on the UTXO based models.

3.6.2.3 Communication Model

The communication model in PoI is asynchronous. The Proof of Importance communicates with its peer nodes in asynchronous communication. The asynchronous communication does not have an upper bound on message delay.

3.6.2.4 Electing Miners

The Proof of Luck, the nodes select a random number, and the node with the highest number is elected to produce the block, which can be added to the chain. The nodes in

PoL like PoW require high computational power to run several attempts to reach a lucky number [8].

3.6.2.5 Energy Savings

The energy savings in Proof of Luck is high as it does not involve any mining activity like PoI PBFTconsensus. The electrical power in consensus algorithms is conserved, which does not have the mining mechanism. Although, as mentioned earlier, the computation power is higher, like Proof of Work, to reach the lucky number.

3.6.2.6 Tolerated power of the adversary

The tolerated power of an adversary in Proof of Luck is less than 50% of the processing power. The design of the Proof of luck consensus is such that, the attacker to find the random luck value is a big hurdle to crash the network.

3.6.2.7 Block Rewards

The block reward is offered to the node, which picked the highest random luck value and produced the block that is added to the chain.

Verification speed - The verification time of Proof of Luck is greater than 15 seconds, still better than the verification speed of PoW.

3.6.2.8 Block Creation Time

The block creation speed in Proof of Luck is high, and the verification process is quick due to high computational power.

3.6.2.9 Scalability

The scalability in Proof of Luck is high. It measures the number of transactions a consensus mechanism can process at a time and size of the block created by the node.

3.6.2.10 51% Attack

The 51% attack is not possible in Proof of Luck, as PoL does not involve computational power. The node with a high computing rate will have the ability to calculate the nonce value faster. Sometimes these backfires if a selfish node that has high computational power or rate than the total computational rate of all nodes combined can compromise the network leading it to a 51% attack. In ideal conditions, to conduct a 51% attack is quite expensive, considering the massive network and owning high computational power.

3.6.2.11 Double spending attack

There are no chances of a double-spending attack possible in Proof of Luck. If a consensus algorithm has no chance of a 51% attack, then the double-spending attack can be ruled out.

Note- There are some metrics that are not compared as they may be yet to be explored, or the consensus algorithm may not have been exposed to such attributes. The PoI has some metrics which are yet to be found like block reward, transaction fee, throughput, Byzantine Fault Tolerance.

4 ALGORITHM COMPARISON ANALYSIS

We quickly analyze the aspects that have an impact on the selection and usage of consensus algorithms in this Section.

The report identifies the algorithms that are private and public type blockchains. All of the proof-based algorithms mentioned in this report are suitable for public blockchain networks, whereas vote-based blockchain algorithm is very much suited for private usage (e.g., PBFT in an enterprise setup).

From a mining efficiency perspective, PoW is high in power consumption because of the complex puzzle the user needs to solve for rewards. PoW will be a less desirable algorithm in the future for a new blockchain network, whereas PoS, DPoS, and PoA type algorithms are likely to dominate the new and upcoming public blockchain networks. The mining process in PBFT is based on the exchange of messages between nodes, which could create overheads in the private network, but modified PBFT type algorithms will start to dominate the private blockchain networks.

Energy saving is critical in the present world where everyone is concerned about sustainability, as you may have noticed in the report at various instances, PoW results in consuming large amounts of energy which makes that algorithm as the "most energy-consuming algorithm." There are reports surveyed for this project, which clearly shows developers and consumers are increasingly interested in somewhat hybrid proof-based algorithms such as proof of stake and DPoS. For the permissioned blockchain, electrical power is saved considerably because of the nodes not performing mining operations as in proof-based algorithms.

As for the incentive, private blockchains are typically reliant on enterprise resources; therefore, no rewards are provided to miners. For the public blockchain networks, miners are provided with incentives for what they spend on computing and electrical power so that it will encourage more participants.

Performance of the blockchains are elucidated utilizing verification speed, throughput, block creation speed and scalability. Private blockchains are better in performance when compared to public blockchains. On public blockchains, Proof of Work is the slowest one compared to the other proof-based algorithms. When it comes to scalability, public blockchains fare well compared to PBFT.

The security level of consensus algorithms is measured using two criteria a.) Tolerated power of adversary and b.) Exposure likelihood.

The worst tolerated power of the adversary was found to be PoW algorithm followed by the PBFT algorithm. All other proof-based algorithms were relatively better than the above two. As far as the other threat exposures listed in the comparison table, most of the algorithms listed are successful in warding off 51% and double-spending attacks.

Related to the above, there was the question of selecting which cryptocurrencies to include in the analysis chart, but the cryptocurrency market is fluctuating, causing the top performer's list to vary year-round. Furthermore, there is no ranking to the equivalent of a marketability contest rather than comparisons being based on objective measures.

5 CONCLUSION AND FUTURE WORK

With large and small enterprises moving towards blockchain based solution for their business needs, the time has never been perfect before for a consensus algorithm comparison project. In this report, an analysis of popular consensus algorithms was presented, and the metrics used for their comparison were clearly explained. For any future work related to evaluating various consensus algorithms, this report will serve as a primer so that the reader doesn't have to reinvent the wheel and focus more on quantitative areas which the report may not have addressed. Based on the explanation provided for the inner workings of the consensus algorithms, the reader might be able to design suitable experiments for further qualitative evaluation of the algorithms.

6 REFERENCES

[1] N. C. a. M. M. Yousaf., " "Consensus Algorithms in Blockchain: Comparative Analysis, Challenges and Opportunities,"," in *12th International Conference on Open Source Systems and Technologies (ICOSST), Lahore, Pakistan,*, 2018 .

[2] K. Qin and A. Gervais, ""An overview of blockchain scalability, interoperability and sustainability,"," in *Imperial College, London,* , 2019.

[3]] C. Shumo and W. Sophia, ""The Curses of Blockchain Decentralization,"," ArXiv, Seattle, 2018.

[4] S. W. Shumo Chu, "curses of Blockchain Decentralization University of Washington and Epichain.io," [Online]. Available: https://arxiv.org/pdf/1810.02937.pdf.

[5] C. Shumo and W. Sophia, "The Curses of Blockchain Decentralization," ArXiv, Seattle, 2018.

[6] "Block chain Types," Dec 2019. [Online]. Available: https://www.researchgate.net/publication/338159337_Blockchain_Types.

[7] "Permissioned and Permission less Block chain: A compressive guide," 13, NOV , 2019. [Online]. Available: https://www.blockchain-council.org/blockchain/permissioned-and-permissionless-blockchains-a-comprehensive-guide/.

[8] F. A. A. Shikah J. Alsunaidi, "A survey of consensus algorithms for block chain technology," [Online]. Available: https://ieeexplore-ieee-org.login.ezproxy.library.ualberta.ca/stamp/stamp.jsp?tp=&arnumber=8716424.

[9] B. Curran, "Comparing Bitcoin and Ethereum: UTXO vs Account based transaction model," July 23, 2018. [Online]. Available: https://blockonomi.com/utxo-vs-account-based-transaction-models/.

[10] J. Clifford, "Intro to Blockchain: UTXO vs Account Based," September 20, 2019. [Online]. Available: https://medium.com/@jcliff/intro-to-blockchain-utxo-vs-account-based-89b9a01cd4f5.

[11] T. McCallum, "Diving into Ethereum's World state," February 10, 2018. [Online]. Available: https://medium.com/cybermiles/diving-into-ethereums-world-state-c893102030ed.

[12] "How Ethereum works," March 30, 2017. [Online]. Available: https://www.coindesk.com/learn/ethereum-101/how-ethereum-works.

[13] " Bitcoin Energy Consumption Index. Digiconomist," [Online]. Available: https://digiconomist.net/bitcoin-energy-consumption.

[14] L. C. Ryan cole, "Modelling of the Enrgy consumption of blockchain consensus algorithms," 2018. [Online]. Available: https://ieeexplore-ieee-org.login.ezproxy.library.ualberta.ca/stamp/stamp.jsp?tp=&arnumber=8726725.

[15] C. Stoll, "The carbon footprint of bitcoin," 17, july 2019. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S2542435119302557.

[16] M. P. D.-. A.-s. Rachid Guerraoui, "Blockchain protocols: The adversary is in the details," [Online]. Available: https://pdfs.semanticscholar.org/044e/59d5dcf4e24fc104e0cb784ebf3b0f3f1fd2.pdf.

[17] y.-y.-t. Jen-yeu chen, "Distributed intrusion detection of Byzantine attacks in wireless networks with random linedar network coding," march 2013. [Online]. Available: https://www.researchgate.net/publication/235892644_Distributed_Intrusion_Detection_of_Byzantine_Attacks_in_Wireless_Networks_with_Random_Linear_Network_Coding.

[18] "What is the blockchain fee," [Online]. Available: https://wirexapp.com/help/article/what-is-the-blockchain-fee-0078.

[19] M. j. m. s. M. A. C. Md sadek ferdous, "Blockchain consensus algorithms: A survey".

[20] E. B.-H. D. L. a. G. M. Antoine Durand, "Asymptotic Performance Analysis of Blockchain," February 12,2019.

[21] "Blockchain performance Throughput and Scalability," [Online]. Available: https://www.himss.org/blockchain-performance-throughput-and-scalability\.

[22] C. W. Q. h. Shenglin Wang, "Corking by Forking:Vulnerability anlaysis of blockchain".

[23] "What is a 51% attack?," [Online]. Available: https://www.binance.vision/security/what-is-a-51-percent-attack.

[24] A. K. S. Kevin Jonathan, "security issues and vulnedrabilities on a block chain system:review," [Online]. Available: https://ieeexplore-ieee-org.login.ezproxy.library.ualberta.ca/stamp/stamp.jsp?tp=&arnumber=9034659.

[25] G. l. h. k. y. g. f. Congcong ye, "Analysis of security in blockchain:Casestudy in 51%-attack detecting," 2018. [Online]. Available: https://ieeexplore-ieee-org.login.ezproxy.library.ualberta.ca/stamp/stamp.jsp?tp=&arnumber=8563187.

[26] Z. Zia, "Apples to apples, Decred is 20x more expensive to attack than bitcoin," June 2, 2018 . [Online]. Available: https://blog.usejournal.com/apples-to-apples-decred-is-20x-more-expensive-to-attack-than-bitcoin-68bafeb4546f .

[27] "Double - Spending," [Online]. Available: https://en.wikipedia.org/wiki/Double-spending.

[28] T. K. sharma, "Hoe blockchain is solving the problem of Double spending in the finance sector?," November 3, 2018. [Online]. Available: https://www.blockchain-council.org/blockchain/how-blockchain-is-solving-the-problem-of-double-spending-in-the-finance-sector/.

[29] J. s, "Blockchain explained: How a 51% attack works(double spend attack)," May 5, 2018. [Online]. Available: https://blog.goodaudience.com/what-is-a-51-attack-or-double-spend-attack-aa108db63474.

[30] J. Frankenfield, "Blockchain-Hash," August 15, 2019. [Online]. Available: https://www.investopedia.com/terms/h/hash.asp.

[31] H. S. Suman Ghimire, "A survey on bitcoin crypto currency and its mining," November, 2018. [Online]. Available: https://www.researchgate.net/publication/331040157_A_Survey_on_Bitcoin_Cryptocurrency_and_its_Mining .

[32] G. O. K. W. V. G. Arthur Gervais, "On the security and performance of Proof of Work blockchains".

[33] "Scalability Overview," [Online]. Available: https://spec-rationality.com/scalability-overview/.

[34] N. gaski, "Public vs Private Blockchain: what you need to know," March 13, 2019. [Online]. Available: https://kaleido.io/public-vs-private-blockchains-what-you-need-to-know/.

[35] "A review of asynchronous and semi synchronous blockchain," [Online]. Available: https://coinsavage.com/content/2018/09/a-review-of-asynchronous-and-semi-synchronous-blockchains/.

[36] M. M. Y. Natalia Chaudhry, "Consensus Algorithm in Blockchain: Compartive analysis , challenges and oppurtunities," [Online]. Available: https://ieeexplore-ieee-org.login.ezproxy.library.ualberta.ca/stamp/stamp.jsp?tp=&arnumber=8632190.

[37] B. M. L.M. Bach, "Compartive analysis blockchain consensus algorithm," [Online]. Available: https://ieeexplore-ieee-org.login.ezproxy.library.ualberta.ca/stamp/stamp.jsp?tp=&arnumber=8400278.

[38] H. H. Z. Z. J. B. Qiheng Zhou, "Solutions to Scalability of Blockchain: A survey," 2016.

[39] A. E. G. E. G. S. IItay Eyal, "Bitcoin-NG: A scalable block chain protocol," [Online]. Available: https://arxiv.org/pdf/1510.02037.pdf.

[40] Y. c. X. C. Xinle Yang, "Effective scheme against 51% attack on Proof of Work Block chain with History Weighted information," [Online]. Available: https://ieeexplore-ieee-org.login.ezproxy.library.ualberta.ca/stamp/stamp.jsp?tp=&arnumber=8946277.

[41] "PoW 51% attack cost," [Online]. Available: https://bitcoinexchangeguide.com/list-of-pow-51-proof-of-work-mining-attack-costs-for-each-cryptocurrency/.

[42] "Proof of Stake," [Online]. Available: https://en.wikipedia.org/wiki/Proof_of_stake.

[43] "History and Evolution of Proof of Stake," [Online]. Available: https://cointelegraph.com/news/the-history-and-evolution-of-proof-of-stake-.

[44] K. Kullar, "Implementing Proof of Stake Part-1," January 25, 2019. [Online]. Available: https://medium.com/coinmonks/implementing-proof-of-stake-e26fa5fb8716.

[45] J. Ma, "Confirmation Time," [Online]. Available: https://www.binance.vision/glossary/confirmation-time.

[46] C. &. D. T. H. &. N. D. &. N. D. &. N. H. &. D. E. Nguyen, " Proof-of-Stake Consensus Mechanisms for Future Blockchain Networks: Fundamentals, Applications and Opportunities," in *PP. 1-1. 10.1109/ACC*, 2019.

[47] "What is Proof of Stake and how does it work? Ultimate coin stakeing guide," [Online]. Available: https://blokt.com/guides/proof-of-stake.

[48] M. S. &. C. M. &. H. M. &. C. A. Ferdous, " Blockchain Consensus Algorithms: A Survey.," 2020.

[49] "Proof of stake explained," [Online]. Available: https://www.binance.vision/blockchain/proof-of-stake-explained.

[50] "Hyperledger Blockchain Performance Metrics," [Online]. Available: https://www.hyperledger.org/resources/publications/blockchain-performance-metrics#transaction-throughput.

[51] D. Bharel, "How Proof of Stake renders a 51% attack unlikely and unapplealing," September 19,2018. [Online]. Available: https://blog.qtum.org/how-proof-of-stake-renders-a-51-attack-unlikely-and-unappealing-ddebdc91a569.

[52] "A survey on Long range attacks for Proof of Stake Protocols," January 20, 2019. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8653269.

[53] "Proof of stake- Byzantine Fault tolerance," [Online]. Available: https://github.com/ethereum/wiki/wiki/Proof-of-Stake-FAQ#what-would-the-equivalent-of-a-51%25-attack-against-casper-look-like.

[54] Z. H. P. B. a. B. G. G. Chaumont, "DPoPS: Delegated Proof-of-Private-Stake, a DPoS implementation under X-Cash, a Monero based hybrid-privacy coin,," X-Network, Paris, 2019.

[55] G. Chaumont, Z. Hildreth, P. Bugnot and B. Giraux, "DPoPS: Delegated Proof-of-Private-Stake, a DPoS implementation under X-Cash, a Monero based hybrid-privacy coin," X-Network, Paris, 2019.

[56] N. Kesonpat, "Consensus algorithm," June 9, 2018. [Online]. Available: https://www.nichanank.com/blog/2018/6/4/consensus-algorithms-pos-dpos.

[57] "DPoS," [Online]. Available: https://en.bitcoinwiki.org/wiki/DPoS.

[58] "DPoS BFT - Pipelined Byzantine Fault tolerance," [Online]. Available: https://eos.io/news/dpos-bft-pipelined-byzantine-fault-tolerance/.

[59] "EoS DPoS (delegated Proof of Stake)," [Online]. Available: https://steemit.com/cryptocurrency/@vtce/eos-dpos-delegated-proof-of-stake.

[60] C. Adams, "Understanding LISk," December 6, 2017. [Online]. Available: https://www.investinblockchain.com/lisk-delegated-proof-of-stake/.

[61] M. Castro, "Practical Byzantine Fault Tolerance," Cambridge, MA, 2001.

[62] M. Castro, ""Practical Byzantine Fault Tolerance," Massachusetts Institute of Technology," Cambridge, MA,, 2001.

[63] S. Walters, "Proof of Activity explained: A hybrid consensus algorithm," April 6, 2018. [Online]. Available: https://www.coinbureau.com/blockchain/proof-of-activity-explained-hybrid-consensus-algorithm/.

[64] "Reaching consensus using Proof of Activity Algorithm," [Online]. Available: https://blockspoint.com/articles/technologies/proof-of-activity.

[65] S. Dhar, "What is Proof of Activity," October 22, 2019. [Online]. Available: https://theblockchaincafe.com/what-is-proof-of-activity-poa/.

[66] "Thee heart of Eneccum_Proof of Activity is a promising work algorithm," [Online]. Available: https://bigcoinvietnam.com/trai-tim-cua-enecuum-proof-of-activity-mot-thuat-toan-cong-viec-day-hua-hen.

[67] H. Anwar, "Consensus algorithm:The root of the blockchain technology," August 25,2018. [Online]. Available: https://101blockchains.com/consensus-algorithms-blockchain/#prettyPhoto.

[68] K. Qin and A. Gervais, "An overview of blockchain scalability, interoperability and sustainability," Imperial College, London, 2019.

[69] C. M. a. D. P. ". S. A. i. B. u. D. B. M. o. M. P. Swathi, in *10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)pp. 1-6*, Kanpur, India, 2019.

[70] N. D. G. B. a. J. S. D. ". A. o. a. P.-o.-S. B. 2. 2. I. C. o. E. o. C. C. S. (. M. V. 2. p. 1.-2. W. Y. Maung Maung Thin, " "Formal Analysis of a Proof-of-Stake Blockchain," 2018 23rd International Conference on Engineering of Complex Computer Systems (ICECCS)," in *pp. 197-200*, Melbourne, VIC, 2018.

[71] H. M. A. Aljassas and S. Sasi, ""Performance Evaluation of Proof-of-Work and Collatz Conjecture Consensus Algorithms," 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS," in *pp. 1-6*, Riyadh, Saudi Arabia,, 2019.

[72] "Implementing PBFT in Blockchain," [Online]. Available: https://medium.com/coinmonks/implementing-pbft-in-blockchain-12368c6c9548.

[73] A. A. S. J. M. S. S. A. N. a. T. A. T. Ali Syed, ""A Comparative Analysis of Blockchain Architecture and its Applications: Problems and Recommendations," in IEEE Acces," in *vol 7,pp. 176838-176869, 201*.

END OF DOCUMENT