sustainability

MDPI

Article

Modeling and Analysis of Proof-Based Strategies for Distributed Consensus in Blockchain-Based Peer-to-Peer Networks

Majed Abdullah Alrowaily [1,*], Mansoor Alghamdi [1,*], Ibrahim Alkhazi [2], Ahmad B. Hassanat [3], Musab Mutasim Saeed Arbab [4] and Charles Z. Liu [5]

[1] Department of Computer Science, Applied College, University of Tabuk, Tabuk 47512, Saudi Arabia
[2] College of Computers & Information Technology, University of Tabuk, Tabuk 47512, Saudi Arabia
[3] Faculty of Information Technology, Mutah University, Karak 61710, Jordan
[4] Department of Administrative Sciences, Applied College, University of Tabuk, Tabuk 47512, Saudi Arabia
[5] School of Electrical and Information Engineering, University of Sydney, Sydney, NSW 2008, Australia
* Correspondence: malrowaily@ut.edu.sa (M.A.A.); malghamdi@ut.edu.sa (M.A.)

Abstract: Blockchain technology has a wide range of applicability in the fields of transportation infrastructure construction and maintenance, transportation big data analysis and application, expressway toll collection, and logistics. The core technology lies in the distributed, decentralized, immutable, and programmable features brought about by consensus. This paper studies the dynamic analytical modeling of Proof-Based Consensus (PBC) strategies in blockchain systems, focusing on basic strategies, including Proof of Work (PoW), Proof of Stake (PoS), Proof of Authority (PoA), and Proof of Luck (PoL), which can be extended to other PBC models. We focus on modeling these typical strategies and discuss their solution characteristics in terms of algorithmic mechanisms and principles. The relevant results can be used for quantitative analysis and evaluation of distributed consensus based on the model.

Keywords: proof-based strategy; distributed consensus; Byzantine general problem; blockchain; peer-to-peer network

check for updates

Citation: Alrowaily, M.A.; Alghamdi, M.; Alkhazi, I.; Hassanat, A.B.; Arbab, M.M.S.; Liu, C.Z. Modeling and Analysis of Proof-Based Strategies for Distributed Consensus in Blockchain-Based Peer-to-Peer Networks. *Sustainability* **2023**, *15*, 1478. https://doi.org/10.3390/su15021478

Academic Editor: Huaming Chen

Received: 20 November 2022
Revised: 25 December 2022
Accepted: 3 January 2023
Published: 12 January 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Blockchain has been increasingly applied in many industries, including e-business [1–3], finance and banking [4–6], smart grid [7], medicare [8–10], and other administration fields [11–13], which is revolutionizing the traditional industries. In addition, the application of blockchain within the education sector is expected to make improvements to academic transcripts, credentials, digital libraries, and student records. Community validation is essential to guarantee the synchronization of ledgers replicated across commercial transactions in a blockchain [14], and the cornerstone of its application lies in that blockchain technologies enable an information system with the distributed account of block chaining, supporting the sharing of single fact versions across different complex ecosystems and processes, which brings many new opportunities for novel types of business [15–18].

The strength of blockchain-based systems is mainly brought by global consensus [19–21] and tamperproof traceability [22,23], in which being tamperproof is also guaranteed by the global consensus because it increases the cost of fraud behaviors. The blockchain allows all nodes in the network to have a timestamp on each block to record as a global consensus confirmation (e.g., [24]), indicating that the information was written with the global consensus, thus enabling applications, such as e-business, to be more trustworthy while also being flexible due to decentralization [18,25,26]. Blockchain technology is also playing an increasingly important role in the transportation industry. Blockchain-based smart contracts can provide shippers and carriers with inexpensive as well as secure [27]

contracts that are not dependent on third parties [28]. Transportation companies using blockchain technology can effectively track fleet or vehicle performance history and provide shippers and carriers with more transparent [29] vehicle data. For time-sensitive shipments, such as pharmaceuticals and food, that require cold storage, blockchain technology can provide better cargo tracking [30].

Another key feature of blockchain systems is decentralized and secure storage [31,32]. The essence of this is the distributed tamper-evident nature of storing data on the blockchain with consensus authentication, which makes it difficult for a single entity to modify or delete data on the ledger without the consensus of other nodes in the network [33,34]. To achieve secure storage on blockchains, it is important to ensure that the consensus protocol used by the network is robust and able to withstand attacks. Many blockchain networks use proof-based consensus protocols, which require nodes in the network to provide evidence or "proof" to support the validity of their proposed transactions or blocks [35]. Such proofs can take various forms, depending on the specific protocol being used.

The blockchain-based consensus enables P2P network systems to perform decentralized authentication. Via decentralized authentication and distributed exchange network, blockchain can perform a fast operation for international cross-border trading and commercial transactions, and with the autonomous smart contract approval, automatic commercial authentication is very useful for business tracking [36–38], especially for the application of supply chain management in trading with logistics.

Despite the tamperproof feature of the blockchain, business security is an issue for blockchain-based systems. The worries are reasonable because although many cryptographic technologies (e.g., PKI and Hash [39–42]) are used in blockchain, cryptographic elements can construct an insecure system accidentally with an unintentional design. For example, even though many users use the blockchain wallet function [43,44], many of them share the key by e-mail due to the convenience of the password.

The similar situation may happen in the nodes of the network as well. One of the typical attacks is the Eclipse attack [45], in which the key phase of this attack lies in communication as nodes in the blockchain remain in constant communication for updating and processing data. When attackers handle passwords to access the authorization of the nodes, they would have higher probability to take over the communications of one or many of the nodes. Thus, the attacker can make the victim node accept false data. The attack is mainly reflected in the tampering of the records and the confusion of information. When the records between the blocks have different results, but there are times when they are added to the blockchain, deviation in the record will occur.

It is necessary to enhance the security and trustworthy network communication of the P2P network. One of the ways to solve this problem is the implementation of consensus protocols and processes. To implement this reinforcement against fraud or unexpected faults or errors, the consensus among the members of the community can further diminish the risk since the cost of fraud will increase, and errors can be filtered or fixed by the majority when information is handled by more members. Because of the importance of community validation in commercial transactions synchronization [14], it is critical to build and maintain a healthy business community with a robust online exchanging mechanism in blockchain-based operation.

Because consensus protocol is significant for the processing of the whole blockchain-based services, we focus on the consensus process in the blockchain-based consensus strategies with modeling and analysis, principles, and solutions.

2. Preliminary

2.1. Problem Statement

Distributed consensus has become a highly dynamic field [21,46,47], and one of the typical problems that the consensus in blockchain-based peer-to-peer (P2P) networks is mainly aiming at solving can be modeled as an equivalent Byzantine Generals Problem (BGP) [19,48,49]. The consensus of BGP is a matter of agreement, in which the generals of

the Byzantine Empire must decide unanimously whether to attack an enemy. Considering that these generals are geographically separated, and there are traitors in the generals, the traitor can act arbitrarily to achieve the following goals: (1) to deceive certain generals to take offensive actions; (2) to promote a decision that is not agreed by all generals, such as when the generals do not want to attack to promote offensive operations; (3) to confuse certain generals so that they cannot make a correct decision. If the traitor achieves one of these goals, the outcome of any attack is doomed to failure, and only a fully agreed effort can be won.

Byzantine assumptions are models of the real world, and computers and networks may exhibit unpredictable behavior due to hardware errors, network congestion or disconnection, and malicious attacks [50–52]. Byzantine fault-tolerant agreements must deal with these failures, and these agreements must also meet the specifications of the problem to be solved.

BGP problems can be modeled as a process of message communication among a group with n members, which can be formulated as $A = BGP(m, n)$, where A refers to the agreement with the true or false result, n refers to the total number of all members in the group, and m refers to the number of traitors in the group. In a binary system, we can model $true = 1$, $false = 0$ as the algebra model of the BGP. Matching with the BGP, generals in the group are modeled as those nodes which transmit messages as transmitters, while adjutants or lieutenants are modeled as the nodes which receive messages as receivers. If there is no interference between generals, no traitor blocks send a message (truncation) and falsify the message (forgery). Then, each general can send their message to every other general without error. If there are traitors in the group, how the group can still reach the correct truth is the core task of the solution to this problem.

In a practical P2P network, this type of traitor can be from not only fraud behaviors but also the faults in node or network systems, such as communication, network delay, and errors. Therefore, to guarantee the reliability of the system with fault tolerance, we need to design a certain protocol to operate among the nodes to filter the errors or faults out for correct consensus. Meanwhile, practical blockchain-based consensus also requires fairness in equity among the members regardless of a gap in the capabilities of each node, verifiability that it is easy for the participant to verify the result and investment that the input and output of the election should be proportional. Therefore, how to solve BGP while meeting these requirements lies at the core of the design of strategies.

2.2. Proof-Based Methods

For BGP and similar problems equivalent to BGP, in blockchain-based systems, there is a type of solution named Proof-Based Consensus (PBC), and the typical examples include Proof of Work (PoW [53–55]), Proof of Stake (PoS [56,57]), Proof of Authority (PoA [58]), and so on.

PoW serves as one of the most fundamental strategies for distributed consensus with many applications in blockchain-based networks [54,59,60]. Based on network propagation, different block sizes, block generation intervals, information dissemination mechanisms, and other factors, a quantitative framework is proposed in [55] to analyze the security and performance of various consensus and network parameters of a PoW blockchain. Meaning further study of trade-offs between performance and safety regulations. PoW provides an important solution to the BGP crux for distributed consensus by the hash puzzle for this proof, which is hard to perform but easy to check. Although it serves as one of the most mature solutions in blockchain applications, this strategy consumes a lot of computing power for only hash puzzles along with its poor performance scalability [61], which renders it to be considered a lowly efficient solution.

To save computing power and energy, some other algorithms are proposed to improve the mechanism and efficiency, such as Proof of Burn (PoB [62]) and Proof of Stake (PoS [63,64]). PoB is an incineration certification mechanism. It is a commitment to vote for who has the leadership of the network by burning the tokens in their hands. PoB is

a method of distributed consensus. Although essentially a variant of PoW, PoB's proof is to win the right to speak through existing value consumption rather than mining to generate value.

PoS [63,64] is another one of the strategies proposed to improve PoW with many corresponding applications as well [56,65]. PoS can improve scalability while reducing computing power consumption with flexible options to organize the services [66,67]. However, the discussions and suspensions of doubt about the PoS protocol still exist because of security and mechanism reliability issues [68,69]. Considering the attacks (e.g., nothing at stake attack [70–72], long-range attacks [73–75], etc.), one type of work on this issue is to improve the security of the PoS-based network [57,76], which provides an idea of the solution of applying Trusted Execution Environments (TEEs) to enforce security. Meanwhile, the security and reliability can also be improved by, for example, using a trusted beacon, authorized endorsers, and common prefix [77]; therefore, the protocol can be organized, and the reliability can be enhanced by flexibly applying various supplementary options.

In order to further save energy and increase the speed to adapt to high-traffic network transactions, PoS can also be extended to, for example, delegated PoS (DPoS [78,79]) and Proof of Activity (PoAc [80]), in which DPoS allows each person holding a bit stock to vote to simplify the network into a connection consisting of super nodes or pools generated after authorization, thereby improving efficiency by reducing network complexity, while PoAc extended PoW with stakeholders to enable the network the PoS-based economic incentives in charge of securing the network.

Different from PoS, Proof of Authority (PoA [81]) also provides a solution to improve the performance of distributed consensus in a blockchain with the consideration of security and efficiency, in which the miner was selected as the certifier of the block by virtue of the personal credit rather than the cryptocurrency of the mortgage or computing power. Therefore, the PoA blockchain is protected by a trusted authentication node under a scalable structure. However, it acts potentially as a centralization. To enforce the decentralization, combining PoW with PoA can be a solution [82] for the balance of efficiency and decentralization.

To improve the performance of decentralization with more fairness and security, Proof of Luck (PoL [83,84]) provides a lottery-like mechanism with specific hardware based on, for example, SGX (Software Guard Extensions [85–87]) to guarantee security and fairness. Similarly, Proof of Elapsed Time (PoET [88,89]) also applied SGX to guarantee fairness, in which each node obtains a random wait time (elapsed time) through trusted operations based on the security hardware (SGX) from an enclave before releasing the block data, and the node with the shortest waiting time will be the first to publish the block data. The advantage of PoET is that participation is low. Because more people can easily join and then achieve decentralization. For all participants, it is easier to verify that the leader is legally elected. The cost of the controller's electoral process is proportional to the value gained from it. Although the cost of PoET is low, specific hardware must be used, so the agreement will not apply to the public blockchain until it is adopted on a large scale.

Proof of Capacity or Proof of Space (PoC or PoS [90–92]) replaces the hash puzzle's computing power by actually using the hard disk space, thus improving the inefficiency and waste of PoW. The network will eventually choose which miner will be the node of the packaged block, and the probability depends on the percentage of storage space that the miner takes out of the entire network. One of the benefits of using this space-proven algorithm is that it can equalize the ability to mine. To further improve security, Proof of Space and Time (PoCT [93,94]) introduces a layer of time proofing mechanism based on PoC, using a dedicated time server to verify the time, and detect the time of the server in the interval between each two mining periods, so that in every two blocks, a time workload is added between them to prevent large computing resource owners from disrupting the network by repeating PoC.

2.3. Byzantine Fault-Tolerant Methods

There are also many other protocols that are available to solve BGP, such as PBFT [51,52], RPCA [95,96], Distributed-Acyclic-Graph-Based Consensus (DAG/Tangle [97]), Federated Byzantine Agreement (FBA [98,99]), etc., which are different from PBC methods. These methods can be named Byzantine Fault-Tolerant (BFT) methods. The PBC method mainly emphasizes the proof, while the BFT methods stem from the flow, structure, and order of communication under a BGP puzzle.

According to the problem modeling of BGP, it can be seen that the model is usually characterized by the elasticity m, which represents the number of error processes the algorithm can handle. The greater the m is, the more robust the protocol is. It can be seen that m has an upper bound, and the rough upper bound is $m < n$. However, how to find the exact tolerance bound, $\bar{m}$ that $m \leq \bar{m} \leq n$ with a higher upper bound, is the crux that most protocols are directed at.

One of the most famous protocols is oral messages (OM) [100–102], which is mainly about message delivery with the consistent delivery of verbal messages. If there are m treason generals, the total number of generals must be $3m + 1$ or more, which is the protocol for converging to a decisive agreement.

The OM algorithm is widely used in consensus process design. The solution provided by the OM protocol for the Byzantine general problem can actually be modeled as a probability problem determined by the number of people. The fault tolerance of the OM can be formulated as:

$$R_{OM} = \frac{m}{3m + 1} \tag{1}$$

It can be seen that the upper bound of the fault tolerance is around 33%, while ideally, the upper bound is 50%. Therefore, based on the OM, many blockchain consensus processes are designed based on or impacted by its theory to pursue the ideal upper bound of fault tolerance. As a practical implementation of the BFT, PBFT can be used to implement fault-tolerant distributed systems in practice. In the case of PBFT, each node sends its proposed changes to the state of the system to the other nodes, which then verify the proposed changes and reach a consensus on whether to accept or reject them. This process ensures that the system remains consistent and correct, even in the presence of faulty or malicious nodes.

PBFT is designed to be resilient to Byzantine failures, which are errors or malicious attacks that can cause nodes in a network to behave in an arbitrary or unpredictable manner. PBFT uses a combination of cryptography and communication protocols to ensure that nodes in a network can reach consensus even in the presence of a Byzantine failure. High performance. PBFT is typically faster and more efficient than other consensus protocols, such as Proof of Work (PoW) or Proof of Stake (PoS), because it does not require nodes to perform expensive computational tasks or wait for a certain number of blocks to be mined. This makes PBFT ideal for systems that require high levels of scalability and throughput. In terms of simplifying governance, PBFT does not require a complex governance structure because the nodes in the network can reach a consensus through direct communication with each other. This can make PBFT-based systems easier to manage and operate than systems that rely on more complex consensus protocols.

PBFT is typically more centralized than other consensus protocols because it relies on a small number of nodes to reach consensus and validate transactions. This can make PBFT-based systems less decentralized and less resilient to malicious attacks or failures. Limited scalability. While PBFT is typically faster and more efficient than other consensus protocols, it still has limited scalability because the number of nodes in the network and transaction rates can affect system performance. Complex implementation. PBFT can be more complex to implement and maintain than other consensus protocols, as it requires the use of advanced cryptography and communication protocols.

Compared to BFT, PBFT is more efficient as it uses a specific protocol to reach a consensus among the nodes in the system and requires less information to be exchanged

among them, which makes PBFT more efficient in terms of communication overhead and reduces the time it takes to reach a consensus in the system.

To achieve the ideal fault tolerance, PBC can be used as a complementary option because the introduction of the proof enables systems to break through the threshold of traitors constrained by communication. It can be seen that most of the PBC mechanisms are derived or expanded from some fundamental PBC; for example, PoC can be seen as an expansion of PoW by replacing work of computing power with capacity for storage, also deriving PoCT; PoET mechanism is similar to PoW, but consumes much less power based on specific technology SGX (Software Guard Extensions [85–87]), which is similar to PoL; PoAc extends the PoW via PoS [80] while the PoB mechanism generates new blocks by combining the idea of PoW for mining but with lower computing power and PoS for claiming the stake by burning the coin to allocate the opportunity to perform the work.

Considering the diverse extensions derived from the same sources, in this paper, we mainly focus on the typical proof-based consensus, including PoW, PoS, PoA, and PoL, and discuss these fundamental strategies based on their principles and mechanisms. To narrow the issues down, the primary issues are concentrated on that how likely the whole network will take a value as a consensus, modeling and analyzing the characteristics impacted by the principles and mechanism.

3. Proof-Based Consensus

3.1. Proof of Work

3.1.1. Principle and Mechanism

The Proof of Work (PoW) consensus mechanism, which was first introduced by Bitcoin [54], is widely used in blockchain network systems. PoW mainly solves two outstanding problems: (1) how to protect both sides of the transaction; (2) how to avoid the same currency being used multiple times. In response to these two problems, PoW uses the principle of public key cryptography to ensure the confidentiality of parties, where the public key can be used as the accepted address of the blockchain system, and the private key is used to confirm the transfer payment of the currency in the account.

In the blockchain-based P2P network system, the concurrency and verification of messages are important in communication [55,103,104]. Associated with BGP, the sender of each message acts as a general. However, if multiple generals simultaneously launch news, it will inevitably lead to system confusion, which makes it difficult to reach an agreement. In response to this problem, PoW introduced the cost of sending information, i.e., workload, in a system, so the workload consumed by the work determines which node can propagate information over a period of time.

PoW is a foundation of the secure consensus process in blockchain-based P2P networks, in which consensus ensures that the miner is only able to inspect the transactions in the new block and add them to the blockchain by mining that involves a lot of hashing (calculation) attempts, so more computing power means more attempts per second [54]. Since this calculation is essentially a random collision test with very low probability, it requires a large amount of computing resources to be consumed by the payment node, thus discouraging the attack or fraud behaviors [105].

PoW-based blockchains also introduce a signature mechanism, i.e., when a node sends a unified attack message, each node receives the initiator's message and must sign and seal to confirm their respective identities. Meanwhile, to control the same fair competition opportunity for each node in the whole network, the PoW network will adjust the difficulty of solving the hash problem based on the real-time workload statistics. The difficulty level is dynamically adjusted by the PoW protocol. The adjustment of this difficulty is based on the time benchmark.

3.1.2. Modeling and Analysis

Due to the workload, difficulty, and data fingerprint, the cost of attack is increased by cost control barriers. It is impossible to forge a record unless when over 50% of nodes or

computing power is controlled by attackers, but such forgery will cost a lot of energy and time, thus eliminating the possibility of fraud in terms of cost.

In PoW, the nodes in the network find the random number through the workload competition and win the utterance right to modify the blockchain. It can be seen that the blockchain network can be modeled as a system measured by computing power c_i for each node i. Therefore, the BGP problem is solved by PoW with the computing-power-based workload and hash data fingerprint protected by asymmetric encryption. Because data fingerprints and asymmetric encryption strategies play a steady role in the network, the only variable in the system lies in the computing power. Due to the almost brute-force approach, participants with a higher hash rate (computing power) during the search process will be able to make more attempts per unit time, which is more likely to have higher opportunities to become a block winner (leader or command sender) who can receive rewards. Since each member gets an opportunity based on computing power in the network, the probability $P\{v_i\}$ that member node i wins and becomes a leader in the blockchain network with n members and change the consensus value as v_i can be expressed as:

$$P\{v_i\} = \frac{c_i}{\sum_k^n c_k} \tag{2}$$

If the distribution of the computing power in the blockchain-based P2P network is a uniform distribution, then we have $P\{v_i\} = 1/n$, which means the probability depends on the scale of the network n. However, when some nodes increase the investment to gain more powerful computing power, the P2P network will tilt to the powerful node. An extreme case is that node i controls over 50%, yielding $P\{v_i\} > 50\%$, which theoretically implies that this node i takes over the whole network.

To prevent computing-power-based fraud, another factor worth noticing is the ratio of the cost-efficiency between mining and fraud. Take double payment as an example of fraud behavior, the mining cost-efficiency M_{CE} and fraud cost-efficiency F_{CE} can be modeled as:

$$\begin{cases} M_{CE} = \frac{M_I}{M_R} \\ F_{CE} = \frac{F_I}{F_R} \end{cases} \tag{3}$$

where M_I represents mining investment, M_R mining revenue, F_I double payment investment, and F_R double payment profit revenue, which can be calculated by

$$\begin{cases} M_I = M_C + E_f t \\ M_R = N_B \times V_B + I_B \\ F_I = F_C + T_f + \omega_F \\ F_R = F_p + I_F - R_F \end{cases} \tag{4}$$

in which the factors are shown in Table 1.

Table 1. Factors in calculation (4).

Variables	Factor
t	time
M_C	mining machine cost
E_f	electricity
N_B	the number of blocks obtained
V_B	the value of the block
I_B	the interest brought by the income of the block
F_C	mining machine cost for fraud
T_f	transaction fee
ω_F	acquisition fluctuation
F_p	double payment profit
I_F	interest by fraud
R_F	risk cost

It can be seen that when $M_{CE} \geq F_{CE}$, the fraud or attach behavior can be prevented by the balance of the cost-efficiency ratio. Theoretically, even if node i controls over 50% of computing power, the malicious damage behaviors are less likely to happen in terms of return.

3.1.3. Discussion

The PoW consensus algorithm is considered to be one of the best solutions to the Byzantine general problem, which can make the blockchain a Byzantine fault-tolerant system. Not only because the network is decentralized but also because of the workload proof algorithm with asymmetric encryption, identification authentication, and data fingerprint. The high costs involved in the mining process make it difficult and less likely for miners to invest resources to disrupt the network. Even if the worst case is that the P2P network suffered from over 50% attacks, the behavior of tampering or double payment will be recorded in the block as well.

The advantages of PoW include that the algorithm is simple and easy to implement, and the consensus process can be reached without exchanging additional information between nodes. Because its security is supported by the entire network, the cost of destroying or tampering with the system is enormous, so PoW has the property of preventing damaging attacks at a cost-effective price. However, it needs tremendous energy to generate the block, and the process of confirming the block is time-consuming, and it is hard to shorten the period. Many other PBC methods, such as PoC, PoET, and PoCT, serve as extensions of PoW-based consensus using different replacements of the workload with capacity for storage and computing time to optimize the cost-effective performance.

3.2. Proof of Stake

3.2.1. Principles and Mechanism

The Proof of Stake (PoS) consensus algorithm is proposed as an alternative to workload proof, thereby increasing network system efficiency and scalability. Although Proof of Work and Proof of Equity have the same goal, they have fundamental differences and specificities.

The PoS consensus algorithm replaces the mining of the workload proof with a mechanism that verifies the currency based on the participant's pledge. The verifier (can be seen as the founder or the mint from the perspective of cryptocurrency) of each block is determined by its own investment in the block or cryptocurrency other than the amount of power allocated. This design makes PoS distinguished from PoW, specifically in the process of verifying new blocks. Each equity certification system can implement the algorithm in different ways, but, in general, the blockchain is protected by a random electoral process that takes into account the wealth of the node, the age of the coin or token (the time the coin is locked or the time of pledge), and random factors.

Unlike PoW, PoW consumes a lot of energy during the search process, and PoS chooses a leader based on the number of bets to perform the mining process and add new blocks to the chain. The PoS process is a hash function that takes a seed, consisting of a string of any length, as an input. Such a seed can be flexibly composed, for example, using the title or other information of the previous block and then a random string created by some other selected node. Because the tokens in a PoS-based network are indexed, the PoS process will output a token index so that the index can be used to search the transaction history to find and select the current owner of the token as the leader.

3.2.2. Modeling and Analysis

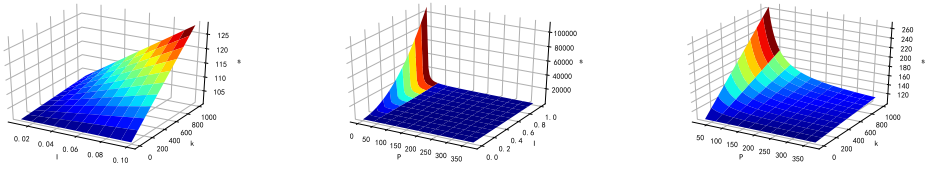
The PoS determines the decision weight allocation for generating a new block based on the amount and time that the members hold the block. From an economic point of view, it can be understood as the relationship between stake and interest. The time of holding the currency is called the age of the coin or stake, and each coin produces 1 unit of currency per day. The longer the member holds the stake, the more stake the member holds, the more

likely the member can generate the block. The stake-generating mechanism in PoS can be formulated as:

$$s(k+1) = s(k) \times \left(1 + \frac{I_s(k) \times T(k)}{P_T}\right) \tag{5}$$

in which $s(k)$ refers to the amount of stake at moment k, k time moment, $T(k)$ period of time as stage age or coin age, $I_s(k)$ the interest of a unit stake, and P_T the period of interest.

The correlations between interest I_s, interest period P_T, time moment k, and stake s are shown in Figure 1.



(**a**) the correlation in I_s, k, s (**b**) the correlation in I_s, P_T, s (**c**) the correlation in P_T, k, s

Figure 1. The Comprehensive Correlation Analysis of PoS.

Similarly, the PoS blockchain network can be modeled as a system measured by stake $s_j, j = 1, \ldots, n$, and the probability $P\{v_i\}$ that member node i becomes a leader in the blockchain network with n members and changes the consensus agreement value as v_i can be expressed as:

$$P\{v_i\} = \frac{s_i}{\sum_k^n s_j} \tag{6}$$

where s_i refers to the stake of member i.

PoS can have faster transaction confirmation speed than that of the PoW mechanisms with low energy consumption. The confirmation of a transaction depends on transaction throughput and block confirmation time. We can measure the performance of the network with the number of transactions during a certain time period. Thus, the transaction rate r_{Tx} can be formulated as:

$$r_{Tx} = \frac{N_{Tx}}{T} = \frac{S_B}{S_{Tx} \times T_B} \tag{7}$$

in which N_{Tx} refers to number of transactions, T time period for Tx, S_B the size of block, S_{Tx} the size of a unit transaction, and T_B the size of the block.

This measurement is very important in analyzing multitasking scenarios. The transaction throughput is the number of transactions per second Tx/s a network can process, which is vital to the performance of the network especially when there are many concurrent transactions.

We introduce cost-efficiency between stake keeping and fraud to discuss the fraud behavior. The stake keeping cost-efficiency S_{CE} and fraud cost-efficiency F_{CE} in PoS can be modeled as:

$$\begin{cases} M_{CE} = \frac{M_I}{M_R} \\ F_{CE} = \frac{F_I}{F_R} \end{cases} \tag{8}$$

where S_I represents stake investment, S_R stake revenue, F_I fraud investment, and F_R fraud revenue, which can be calculated by

$$\begin{cases} S_I = S_C + \int_T C_f(t)dt \\ S_R = S_v \times (1 + I_r + \alpha T) \\ F_I = F_C + T_f + \omega_F \\ F_R = F_p + I_F - R_F \end{cases} \tag{9}$$

in which the factors are shown in Table 2.

Table 2. Factors in PoS Dynamics (2).

Variables	Factor
t	time
S_c	stake-keeping cost
C_f	fluctuating cost
S_v	stake value
α_T	value conversion coefficient of stake age
I_r	interest of the stake
T	stake age
F_C	fraud cost
T_f	transaction fee
ω_F	acquisition fluctuation
F_p	fraud profit
I_F	interest by fraud
R_F	risk cost

Therefore, for preventing BGP traitors, when $S_{CE} \geq F_{CE}$, the fraud or attach behavior can be prevented by the motivation in terms of cost-efficiency. Theoretically, node i, who has the largest stake s_i in the network, will be more likely to protect the network than damage it due to its share and right in the whole group.

3.2.3. Discussion

PoS can be regarded as an upgrade consensus mechanism of PoW. The main idea is that the under-receipt of counter-counting rights is inversely proportional to the equity held by the owner. According to the proportion and time of each participant's tokens, the difficulty of mining is reduced proportionally, thus speeding up the search for random numbers.

Compared with PoW, the resource consumption caused by mathematical operations is reduced to a certain extent and the performance of the corresponding consensus mechanism is the same as that of PoW. The advantage of PoS is that it extends the time for consensus to a certain extent, and does not require a lot of energy to mine. However, PoS also inevitably needs mining, which essentially does not address the pain points of commercial applications because all confirmations are only a probabilistic expression, not a certainty, and, in theory, there may be other attacks. For example, a DAO attack event can cause a hard fork of the blockchain [106–108].

The PoS mode reduces the resource consumption caused by mathematical operations to a certain extent, and the performance is also improved accordingly, but it is still based on the hash operation to obtain the accounting rights. Some PBC methods (e.g., PoB) can be seen as an extension of PoS with a stake in the consumption of the coin.

3.3. Delegated Proof of Stake (DPoS)

3.3.1. Principle and Mechanism

DPoS has the same principle as PoS, but in consensus, it is approved by some selected representatives [78]. The main difference from PoS is that the node elects several agents, which are verified and billed by the agent. Its compliance monitoring, performance, resource consumption, and fault tolerance are similar to PoS. Determining the legitimacy of a transaction is performed by a similar vote to the board of directors. The holders cast a certain number of nodes and represent them for verification and accounting.

The working principle of DPoS is to decentralize that each shareholder has influence according to their shareholding ratio, and the result of 51% shareholder voting will be irreversible and binding. The challenge is to achieve 51% approval in a timely and efficient manner. To achieve this goal, each shareholder can delegate his or her voting rights to a representative. A certain number of delegates (e.g., the top 100) with the highest number of votes are rotated to generate blocks according to the established schedule. Each delegate is assigned a time period to produce the block. All representatives will receive 10% of the transaction fee equivalent to an average block. If an average block contains 100 stakes as a

transaction fee, one representative will receive 1 share as compensation. Network delays may cause some delegates to fail to broadcast their blocks in time, which will cause the blockchain to fork. This is known as rotated DPoS consensus.

Authorized ballots are an important part. The process is to select one or more representatives and rank them by each blockchain user or owner. Once set, each transaction made by the user will transfer the ballot from the input representative to the output representative. In general, users will not create transactions that are specifically for voting purposes, as that would cost them a transaction fee. However, in an emergency, some users may find it worthwhile to change their vote by paying more.

In order to reflect the credibility of the representative, each block will pass a status indicator to let the user know how their representative is performing. If they miss too many blocks, then the system will recommend the user change to a new representative. If any representative is found to have issued an invalid block, then all standard members will ask for a new representative before the member wastes more transactions and time.

3.3.2. Modeling and Analysis

To model the DPoS, the factors shown in Table 3 are considered.

Table 3. Factors Used in Modeling DPoS.

Variables	Factor
N_d	the number of delegated representatives
n	the total number of members
N_r	the number of representatives
Ω_n	the aggregation of delegated representatives
Ω_{N_d}	the aggregation of representatives
Ω_i	the aggregation of members i's advocators
s_i	stake of member i
S_{CE}	cost-efficiency of a stake holder
F_{CE}	cost-efficiency of fraud
S_I	stake investment
S_R	stake revenue
F_I	fraud investment
F_R	fraud revenue
S_c	stake-keeping cost
C_f	fluctuating cost
S_v	stake value
S_r	stake revenue as a reward of representative
S_d	stake revenue as a reward of delegate
α_T	value conversion coefficient of stake age
T	stake age
F_C	fraud cost
T_f	transaction fee
ω_F	acquisition fluctuation
F_p	fraud profit
I_F	interest by fraud
R_F	risk cost

According to the rotated DPoS protocol, the probability of the value v_i can be formulated as

$$P\{v_i\} = \frac{P\{i \in \Omega_{N_d}\}}{N_d} \tag{10}$$

DPoS uses the stake as a means to limit the motivation of the fraud, and the probability of fraud is further reduced by means of election. Since $P\{i \in \Omega_{N_d}\}$ not only reduces $P\{v_i\}$, but further through the rotation mechanism, its probability continues to decrease by $1/N_d$. Of course, if the worst happens, even if the fraud behavior occurs in member i, other members can no longer choose the fraudulent member i as the representative in the next round, thus further reducing the chance of fraud.

When every member in the group has the same probability to be selected as the delegate, then:

$$P\{i \in \Omega_{N_d}\} = \frac{n!}{(n-N_r)!N_d!(N_r-N_d)!} \tag{11}$$

where n refers to the number of whole members, N_r refers to the number of representatives, and N_d refers to the number of delegates.

If the mechanism of the generating block is not rotated but determined according to the stake of the delegate i, then the $P\{v_i\}$ can be formulated as:

$$P\{v_i\} = P\{i \in \Omega_n\}\frac{s_i}{\sum_{N_d} s_k} = \frac{n!s_i}{(n-N_r)!N_r!\sum_{N_d} s_k} \tag{12}$$

Further, if the stake of members who vote to member i is also considered, then Equation (13) can be upgraded as:

$$P\{v_i\} = \frac{n!(s_i + \sum_{s_l \in \Omega_i} s_l)}{(n-N_r)!N_r![\sum_{N_d}(s_k + \sum_{s_l \in \Omega_k} s_l)]} \tag{13}$$

Considering the representatives can receive the transaction fee under a certain rate I_r as equivalent to an average block. If an average block contains $\bar{s}$ stakes as a transaction fee, one representative will receive one share as compensation. Network delays may cause some delegates to fail to broadcast their blocks in time, which will cause the blockchain to fork.

Similarly, the stake keeping cost-efficiency S_{CE} and fraud cost-efficiency F_{CE} in DPoS can be modeled by S_I, S_R, F_I, F_R as shown in (14).

$$\begin{cases} S_I = S_C + \int_T C_f(t)dt \\ S_R = S_v \times (1 + I_r + \alpha T) + S_r + S_d \\ F_I(i) = F_C + T_f + \omega_F \\ F_R(i) = F_p + I_F - R_F \end{cases} \tag{14}$$

It can be seen that the stake revenue of DPoS is different from PoS in Equation (4) as in DPoS, there are S_r and S_d in S_R, which adds more profit to the positive contributors, motivating them to prevent BGP traitors. Thus, when $S_{CE} \geq F_{CE}$, the fraud or attach behavior can be limited.

3.3.3. Discussion

The advantage is that the number of participating verification and accounting nodes can be greatly reduced, and second-level consensus verification can be achieved. However, the disadvantage is that the entire consensus mechanism still relies on tokens, so this method is used in many commercial applications that do not require tokens. What is noticeable, the decision-making group is delegated from all members aggregation Ω_n to representative group Ω_{N_r} then delegate group Ω_{N_d}, in which $\Omega_{N_d} \subseteq \Omega_{N_r} \subseteq \Omega_n$. This potentially means the DPoS-based systems tend to be more centralized. Because, for exam-

ple, there are only 100 delegates, it is conceivable that an attacker will perform a denial of service attack on behalf of each turn of the production block.

Fortunately, the threat of this particular attack is easily mitigated by the fact that each delegate's identity is its public key rather than its IP address. This will make it more difficult to determine the DDOS attack target. The potential direct connection between the representatives will make it more difficult to hinder them from producing blocks. Meanwhile, if each delegate has equal voting power and the powers of the top 100 delegates are the same, it is not possible to concentrate power on a single representative by obtaining more than 1% of the votes. Therefore, if the elections to generate representatives, delegates, and blocks are fair, then the system can still be seen as semi-decentralized with a hierarchical decentralization. Theoretically, in this case, it is still reasonable for the delegate node i to protect the network other than damage it in terms of its long-term profit.

3.4. Proof of Authority (PoA)

3.4.1. Principle and Mechanism

The Proof of Authority (PoA) consensus is a modified form of PoS. In PoS, the identity of the verifier replaces the equity with monetary value and plays the role of equity, while PoA is a reputation-based consensus algorithm that provides a practical and efficient solution for blockchains, especially private blockchains. The basic principle is to ensure the reliability of communication and authentication in the blockchain P2P network by using the value of the real identity. In other words, unlike PoS, the owner of the right to speak is no longer mortgaged by the share of the workload or stake but by the credential of the mortgaged individual and selected as the verifier of the block. Therefore, the PoA blockchain is protected by a trusted authentication node.

This consensus mechanism can also be understood as a consensus method for the allocation of voice rights through authentication, which also enables the PoA system to be extended by authorization based on trusted identity authentication. If the identity is associated with some actual credible evidence, such as qualifications, reputation, and influence, then PoS is the block and transaction for the entity with more credible evidence to control so that all affairs are participants who have been validated by approved participants are the system administrators.

The basic mechanism for running PoA is that the verifier runs the local terminal and automatically packages the transaction over the network to the blockchain. These data are reported and updated can be seen as authorization authentication. The verifier needs to ensure that its hardware facilities are always in good condition. At the same time, in terms of identity, PoA verifiers must meet certain conditions, including: (1) the verifier must officially verify the identity on the network and be able to find identity information in the public domain; (2) the verifier first obtains the right to confirm the proceeds, and the assessment block has potential; (3) establishing authoritative inspections and procedures should be consistent. During uploading identity verification, different systems have different assessment criteria. For example, a potential notary must obtain a national notary license to formally become a notary. This kind of certification and authorization work is similar to the government authorization assessment process in society. Therefore, designing different standards for authorization and certification is key to influencing the efficiency of consensus. Although the conditions may vary from system to system, the PoA consensus mechanism design usually requires the certifier to recognize that the standard is fair and open and the method of selecting the certifier is consistent.

The essence of the reputation mechanism is the trust in the identity of the verifier and the hard-won identity of verifiers. Those who succeed in becoming verifiers under PoA can reap the benefits by maintaining a normal trading process, so they want to hold this hard-won seat and avoid undermining the credential's reputation.

3.4.2. Modeling and Analysis

In terms of mechanism, PoA can be equivalent to a consensus mechanism confirmed by a third-party authentication model. To a certain extent, the structure of the PoA network can still be distributed, and communication applied in operation can be peer-to-peer, but from the perspective of decision logic, it tends to be more centralized and hierarchical.

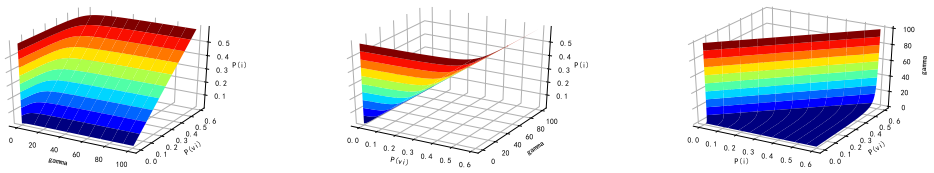
From the perspective of BGP problem solving, the probability of PoA confirming the network consensus agreement does not depend on the network structure and the number of members but on the probability of the authorized verification personnel themselves, which is a non-a priori. To be able to calculate this probability, we introduce risk as the probability of fault $P\{F\}$ to model and analyze.

$$P\{F(i)\} = \frac{1}{1+\gamma(i)} \tag{15}$$

where $\gamma(i)$ refers to the reputation of the authorized member.

$$P\{v_i\} = P\{i\}(1 - P\{F(i)\}) = \frac{P\{i\}\gamma}{\gamma+1} \tag{16}$$

Figure 2 shows the correlations between $P(vi)$ that refers to $P\{v_i\}$, gamma that refers to γ, and $P(i)$ that refers to $P(i)$ from a different view angle.



(**a**) correlation in γ, $P\{v_i\}$, $P(i)$ (**b**) correlation in $P\{v_i\}$, γ, $P(i)$ (**c**) correlation in $P\{v_i\}$, γ, $P(i)$

Figure 2. The Comprehensive Correlation Analysis of PoA.

To prevent fraud, the cost-efficiency of fraud behavior, the authority cost-efficiency A_{CE}, and fraud cost-efficiency F_{CE} can be modeled by the factors shown in Table 4, and the authority and fraud cost efficiency can be formulated as:

$$\begin{cases} A_{CE} = \frac{A_R}{A_I} \\ F_{CE} = \frac{F_R}{F_I} \end{cases} \tag{17}$$

where M_I represents mining investment, M_R mining revenue, F_I double payment investment, and F_R double payment profit revenue, which can be calculated by

$$\begin{cases} A_I = A_C + \gamma_a t \\ A_R = I_B + R_T \\ F_I = F_C + T_f + \omega_F \\ F_R = F_p + I_F - R_F \end{cases} \tag{18}$$

in which, $R_F = \alpha\gamma$. It can be seen that when $\gamma > F_p + I_F/\alpha$, the F_R will be negative, thus guaranteeing the authorized members PoA always keep contributing to the network other than fraud.

Table 4. Factors for PoA Cost-Efficiency Modeling.

Variables	Factor
A_C	cost to perform the tasks
γ	reputation of the member
γ_a	reputation built for authority
R_T	income via the transaction fee
I_B	the interest brought by the income of the block
F_C	cost for fraud
T_f	transaction fee
ω_F	acquisition fluctuation
F_p	fraud profit
I_F	interest by fraud
R_F	risk cost

3.4.3. Discussion

Compared with PoW and PoS, the advantages of PoA are mainly reflected in scalability and controllability. Based on the PoW consensus, the blockchain processing transaction volume (TPS) per second is limited. The reason for the limited transaction volume per second (TPS) is that PoW is based on a distributed node network that needs to reach a consensus; that is, all nodes participating in the verification must agree on the current state of the blockchain to confirm the transaction. In contrast, a PoA system may be the best solution for a private blockchain because PoA performs better than PoS and can do business faster. With the support of authenticated identification, it is easier to conduct accountability and tracking.

The advantages of PoA include the possibility of verifying the node's evil or fraud by the certifier's review and authorization and authentication procedures. PoA can improve computing resources and communication efficiency while saving computing resources and achieving energy saving with high scalability. However, PoA is relatively centralized, especially in the verification and review of participation. Although PoA can be used on the public chain, it is more applicable to be implemented on the private chain. Even if it is deployed on the public chain, its identity review, authentication, and authorization are relatively concentrated on the private system.

3.5. Proof of Luck (PoL)

3.5.1. Principle and Mechanism

PoL is based on blockchain-based betting for the unfairness of the PoL allocation, using a specific bet address, transaction, and block hash to redeem. In practice, the lottery rule is set based on the probability of the number of digits, and the probability of winning is distributed by $1/16, 1/256, (1/16)^n$, and the magnification of the corresponding winning amount is set to a certain rate, such as 8 times, 32 times, or 256 times, thereby performing value distribution. Based on the decentralized blockchain data, the transaction hashes are transparent, contributing to the fairness of PoL allocation [83].

In a typical PoL consensus process, members are required to send certain tokens to specific blockchain addresses, which represent a value similar to that of a stake. After the user transaction receives a certain number $m > 3$ of confirmations, the system will calculate the hash collision draw. The result of the draw will result in a transaction and modify the records in the blockchain data. Winning members have developed new blocks, and the corresponding reward will be recorded in the form of a bonus to the block address of the member response. In the process of obtaining the reward, the member also needs to obtain the private key of the corresponding token address in the blockchain to import the token value corresponding to the record into the corresponding address, thereby obtaining the final token update bound to the member address.

The essence of PoL is actually trying to find the luckiest one in relatively fixed energy consumption, forming a new consensus mechanism to avoid concentration caused by factors such as energy consumption and computing power.

3.5.2. Modeling and Analysis

Corresponding to BGP, PoL's ability to resist fraud is mainly based on the fact that its luck distribution mechanism always follows the blocks generated by a few people and is not welcomed by the wider city members, and the system always assigns more luck to groups with more people. From a technical point of view, if a divergence value is generated, it is possible to treat the record on the blockchain as a fork.

When a fork happens, there is a potential game between the two chains, for example, one of them with a majority population size M and the other with a relatively minority population size m. PoL solves this game with the overall luck allocation. For block t, based on the maximum value of the uniform random variable, PoL protocol modeled the luck of M and m as:

$$\begin{cases} l_M(t) \sim \max\{\{U(0,1)\}_M\} \\ l_m(t) \sim \max\{\{U(0,1)\}_m\} \end{cases} \tag{19}$$

where $U(a,b)$ refers to the random value satisfied with uniform distribution within domain (a,b), and $\{\cdot\}_M, \{\cdot\}_m$ represent the aggregation with M and m elements, respectively. Thus the overall luck of the N_B blocks after the fork is modeled as:

$$L(N_B) = \sum_{t}^{N_B} l_M(t) - l_m(t) \tag{20}$$

It can be seen that

$$\lim_{N_B \to \infty} P\{L_{(N_B)} \leq 0\} = 0 \tag{21}$$

because of the exponentially infinitesimal convergence proved by using a Chernoff bound and independence assumptions [83], which implies that the luck of the minority group m is increasingly less than the majority group M with the generation of blocks in forks.

Figure 3 shows the results of the total luck $L(N_B)$ test. According to the test results, it can be seen that the trends are in exponential dynamics with the generation of the new blocks, which quantitatively indicates that the majority group has an exponential advantage over the minority group to win the lottery for luck allocation.

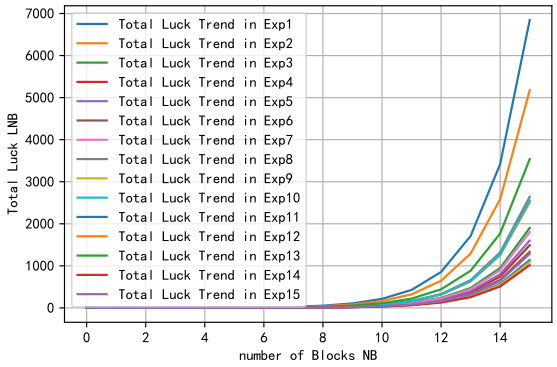


Figure 3. Experiments for Total Luck $L(N_B)$ Testing with $M = 50, m = 5, N_B = 1, 2, \ldots, 15$.

When it comes to attackers, it is nearly impossible to produce a chain that most honest participants prefer, so the records on the blockchain are always updated according to the chain data generated by the majority of the population unless the attackers have very strong luck to win the luck allocation lottery, which is least likely to happen under the constraints of PoL and algorithms [83].

To prevent fraud, the cost-efficiency of fraud behavior, the honest with luck cost-efficiency L_{CE}, and fraud cost-efficiency F_{CE} can be modeled by the factors shown in Table 5, and the authority and fraud cost efficiency can be formulated as:

$$\begin{cases} L_{CE} = \frac{E(L_R)}{E(L_I)} \\ F_{CE} = \frac{E(F_R)}{E(F_I)} \end{cases} \tag{22}$$

where L_I represents honest luck investment for PoL, L_R PoL revenue, F_I the investment for fraud, and F_R the profit revenue of fraud. The lottery mechanism for PoL can be modeled as a Bernoulli distribution, where $p_L(w) \sim B(0,1)$, $p_F(w)$, and L_I, L_R, F_I, F_R can be estimated by

$$\begin{cases} E(L_I) = L_C + \sum_i L_B(i) p_L(i) \\ E(L_R) = (E(L_I) \times R_{PoL}) p_L(w) \\ E(F_I) = F_C + \sum_i F_B(i) p_F(i) \\ E(F_R) = (E(F_I) \times R_{PoL}) p_F(w) \end{cases} \tag{23}$$

It can be seen that when $L_{CE} > F_{CE}$, the fraud or attack behavior is less worthwhile, and the cost-efficiency ratio depends on the probability of luck to win $p_L(w)$ and fraud to win $p_F(w)$, as well as the betting strategy $p_L(i)$, $p_F(i)$, especially when $F_I = L_I, F_C = L_C$. Attackers need to improve the probability of fraud winning $p_F(w) > p_L(w)$ under $E(L_I) = E(F_I)$. If the PoL luck allocation is fair enough to cover all members in the community, then attackers need to devise a better betting strategy than honest players in the community, particularly when $L_C = F_C$, which yields $\sum_i F_B(i) p_F(i) < \sum_i L_B(i) p_L(i)$.

Table 5. Factors for PoL Cost-Efficiency Modeling.

Variables	Factor
L_C	cost to perform the PoL tasks
$L_B(i)$	cost of betting strategy i for PoL
$p_L(i)$	the probability of betting strategy i
$p_L(w)$	the probability of lucky win
F_C	cost to perform the PoL fraud
$F_B(i)$	cost of fraud betting strategy i for PoL
$p_F(i)$	the probability of fraud betting strategy i
$p_F(w)$	the probability of fraud win
R_{PoL}	the bet rate of the PoL lottery winner

3.5.3. Discussion

It is worth noting that, according to the PoL algorithm, it needs to support hardware similar to SGX to provide a credible space on the computing platform, thus ensuring the confidentiality and integrity of user-critical code and data. Therefore, it can be seen that this scheme has certain requirements for the hardware at the bottom.

In addition, during the operation of the PoL, all participants will get different blocks with the same block as the ancestor. The PoL algorithm then chooses a random number between 0 and 1 to quantify the luck. The block held by the member node of the largest number (that is, the block that is the best luck) will be the winner to generate the next block. This, to some extent, indicates the link between luck and the mechanism of random number generation. Random number generation is essentially the core of luck certification.

In order to ensure that the generated random number selection cannot be forged, a certain hardware environment, such as SGX, is required, which is equivalent to relying on the underlying hardware and its manufacturer.

4. Comparison

The characteristics can be summarized in Table 6. Each consensus protocol has its own characteristics applicable to different scenarios and is implemented differently, requiring different trade-offs to be considered.

As the earliest consensus mechanism used to secure and validate transactions on the blockchain, the Proof of Work (PoW) protocol is widely used in Bitcoin. Ether (before the switch to Ether 2.0), Litecoin, and Monroe are examples of cryptocurrencies that use PoW consensus. In a PoW system, individual nodes compete to solve complex mathematical problems to validate transactions and create new blocks. This process is resource intensive as it requires a lot of computing power and electricity. In contrast, in a PoS system, the validation of transactions and the creation of new blocks are performed by stakers, who act as collateral by investing a certain amount of cryptocurrency. The probability of being selected to validate a transaction or create a new block is proportional to the amount of cryptocurrency they have staked. Cardano, Tezos, and Cosmos are examples of cryptocurrencies that use PoS consensus. The practical processes are less resource-intensive than PoW because it does not require miners to solve complex mathematical problems. Thus, PoW is more resource intensive than PoS, which is less resource intensive.

Meanwhile, PoW systems are likely to be more centralized because the mining process requires a lot of expensive hardware and power, which may be beyond the reach of some individuals or organizations. This can result in a few large mining pools controlling a large portion of the network's computing power. PoS systems are typically less centralized because the betting process is more readily available and does not require specialized hardware.

Both PoW and PoS systems can be secure with proper systematic integration of strategies and protocol implementation. However, PoS systems would result in nothing at stake attacks due to incorrect validation by a staker, for example, validating multiple conflicting transactions simultaneously without incurring any penalty.

Delegated Proof of Stake (DPoS) is a variation of the Proof of Stake (PoS) consensus algorithm, whose main feature is that the network selects a set of "representatives" to validate transactions and create new blocks. These delegates are usually chosen based on their stake in the network, but other factors, such as reputation, may also be taken into account. The delegates are then responsible for validating transactions and creating new blocks on behalf of the network. EOS, Steem, and TRON are examples of cryptocurrencies that use DPoS consensus. One of the advantages of DPoS is that it may achieve faster transaction times and higher scalability than PoS because there are fewer nodes responsible for validation. However, it may also suffer from centralization if a small group of representatives holds a large number of shares and has a disproportionate influence on the network.

A typical cryptocurrency that uses PoA consensus is Ethereum 2.0, which relies on nodes in the network to validate transactions and create new blocks. Its system selects specific nodes or "validators" to validate transactions and create new blocks based on their reputation and trustworthiness. These nodes are typically pre-selected by the network, so PoA allows for faster transaction times and higher scalability because the process of validating transactions and creating new blocks does not require as much computational power and is responsible for maintaining the integrity of the network. However, it may also suffer from centralization if a small group of validators has a disproportionate amount of influence over the network.

One difference between PoL and PoA is that PoL's SGX-based consensus protocol will rely on hardware-based security features to provide a secure and trusted environment for consensus, while PoA relies on the reputation and trustworthiness of a particular node. SGX systems allow the software to run in an isolated, secure region of memory that is not accessible by other software, including the operating system. In such an environment, all participants in the network choose a random number (luck), and the person with the highest number wins (luckiest). The luckiest block is then used as the next block in the

blockchain. Since the random number selection occurs in the SGX environment, it cannot be faked. In this case, PoL needs to use SGX-enabled hardware to implement and rely on SGX hardware-based security features to provide a secure and trusted environment for reaching consensus among nodes in the blockchain network. Further, SGX-based consensus protocols require the use of SGX-enabled hardware. SGX-based consensus protocols are newly proposed protocols that are not yet widely used, but their security protection is very far-reaching for the construction of blockchain systems for trusted computing.

Table 6. Summary of the Comparison of Mechanisms, Pros and Cons.

Method	Mechanism	Pros	Cons
PoW	Reconcile competing permissions to add blocks by solving a computational problem.	Provides strong security for the blockchain.	Requires significant energy and resources.
PoS	Validators are chosen to create the next block-based on their stake or how much of the cryptocurrency they hold.	Requires less energy and resources than PoW.	Can be vulnerable to centralization if a few stakeholders control a large portion of the cryptocurrency.
DPoS	Stakeholders vote for a set of delegates to validate transactions and create blocks.	Can be more efficient and faster than PoS.	Can be vulnerable to centralization if a few delegates control a large portion of the votes.
PoA	Validators are chosen based on their reputation or identity, and they have the authority to validate transactions and create blocks.	Provides strong security and can be used in permissioned blockchain systems.	Only suitable for permissioned blockchain systems.
PoL	Validators are chosen randomly to create the next block, with the probability of being chosen based on their stake.	Simplifies the process of selecting validators and can lead to faster block times.	Can be vulnerable to centralization if a few stakeholders control a large portion of the cryptocurrency.

5. Conclusions

This paper mainly studies the modeling and analysis of distributed consensus strategies in blockchain-based consensus systems. Based on the Byzantine general problem (BGP), the existing proof-based methods are briefly combed, and the connection between different methods is discussed. We have attributed all proof-based consensus methods to the typical models, including PoW, PoS, DPoS, PoA, and PoL, focusing on the modeling of these typical strategies and discussing their solving characteristics according to the algorithm mechanism and principle. Other methods, such as PoB, PoAc, PoET, etc., can be obtained through the extension and extension of these models. These results can be used for future distributed consensus and quantitative analysis and result evaluation of corresponding systems, providing a reference and model basis for quantitative analysis of consensus dynamics and service management in the system.

Author Contributions: Writing—original draft, M.A.A. and C.Z.L.; Writing—review & editing, M.A., I.A. and M.M.S.A.; Project administration, A.B.H. All authors have read and agreed to the published version of the manuscript.

Funding: The authors extend their appreciation to the Deanship of Scientific Research at University of Tabuk for funding this work through Research no. 0278-1443-S.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to express our gratitude to the Alan Laicheng Zhong, School of Electrical and Information Engineering, University of Sydney and Naif Al Balwi, Department of Computer Science, Applied College, University of Tabuk for their organizational and academic inputs. We are most grateful to all concerned.

Conflicts of Interest: The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Lin, Q.; Yan, H.; Huang, Z.; Chen, W.; Shen, J.; Tang, Y. An ID-based linearly homomorphic signature scheme and its application in blockchain. *IEEE Access* **2018**, *6*, 20632–20640. [CrossRef]
2. Sidhu, J. Syscoin: A peer-to-peer electronic cash system with blockchain-based services for E-business. In Proceedings of the 2017 26th International Conference on Computer Communication and Networks (ICCCN), Vancouver, BC, Canada, 31 July–3 August 2017; pp. 1–6.
3. Zhang, Y.; Wen, J. The IoT electric business model: Using blockchain technology for the internet of things. *Peer-to-Peer Netw. Appl.* **2017**, *10*, 983–994. [CrossRef]
4. Tapscott, A.; Tapscott, D. How blockchain is changing finance. *Harv. Bus. Rev.* **2017**, *1*, 2–5.
5. Treleaven, P.; Brown, R.G.; Yang, D. Blockchain technology in finance. *Computer* **2017**, *50*, 14–17. [CrossRef]
6. Eyal, I. Blockchain technology: Transforming libertarian cryptocurrency dreams to finance and banking realities. *Computer* **2017**, *50*, 38–49. [CrossRef]
7. Goranović, A.; Meisel, M.; Fotiadis, L.; Wilker, S.; Treytl, A.; Sauter, T. Blockchain applications in microgrids an overview of current projects and concepts. In Proceedings of the IECON 2017—43rd Annual Conference of the IEEE Industrial Electronics Society, Beijing, China, 29 October–1 November 2017; pp. 6153–6158.
8. Zhang, P.; White, J.; Schmidt, D.C.; Lenz, G.; Rosenbloom, S.T. FHIRChain: Applying blockchain to securely and scalably share clinical data. *Comput. Struct. Biotechnol. J.* **2018**, *16*, 267–278. [CrossRef]
9. Griggs, K.N.; Ossipova, O.; Kohlios, C.P.; Baccarini, A.N.; Howson, E.A.; Hayajneh, T. Healthcare blockchain system using smart contracts for secure automated remote patient monitoring. *J. Med. Syst.* **2018**, *42*, 130. [CrossRef]
10. Kaur, H.; Alam, M.A.; Jameel, R.; Mourya, A.K.; Chang, V. A proposed solution and future direction for blockchain-based heterogeneous medicare data in cloud environment. *J. Med. Syst.* **2018**, *42*, 156. [CrossRef]
11. Vos, J.; Lemmen, C.; Beentjes, B. Blockchain based land administration feasible, illusory or a panacea. In Proceedings of the Netherlands Cadastre, Land Registry and Mapping Agency, Paper Prepared for Presentation at the 2017 World Bank Conference on Land and Povertry, Washington, DC, USA, 20–24 March 2017.
12. Klechikov, A.; Pryanikov, M.; Chugunov, A. Blockchain Technologies and their Application in Government Administration. *Int. J. Open Inf. Technol.* **2017**, *5*, 123–129.
13. Ferrer, E.C. The blockchain: A new framework for robotic swarm systems. In Proceedings of the Future Technologies Conference, Vancouver, BC, Canada, 13–14 November 2018; pp. 1037–1058.
14. Aste, T.; Tasca, P.; Di Matteo, T. Blockchain technologies: The foreseeable impact on society and industry. *Computer* **2017**, *50*, 18–28. [CrossRef]
15. Morabito, V. *Business Innovation through Blockchain*; Springer International Publishing: Cham, Switzerland, 2017.
16. Mougayar, W. *The Business Blockchain: Promise, Practice, and Application of the Next Internet Technology*; John Wiley & Sons: Hoboken, NJ, USA, 2016.
17. Swan, M. *Blockchain: Blueprint for a New Economy*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2015.
18. Amit, R.; Zott, C. Value creation in e-business. *Strateg. Manag. J.* **2001**, *22*, 493–520. [CrossRef]
19. Gramoli, V. From blockchain consensus back to byzantine consensus. *Future Gener. Comput. Syst.* **2017**, *107*, 760–769. [CrossRef]
20. Abraham, I.; Malkhi, D.; Vmware Research Group. *The Blockchain Consensus Layer and BFT*; Bulletin of EATCS; EATCS: Camerino, Italy, 2017.
21. Baliga, A. *Understanding Blockchain Consensus Models*; Allen Institute: Seattle, WA, USA, 2017.
22. Lu, Q.; Xu, X. Adaptable blockchain-based systems: A case study for product traceability. *IEEE Softw.* **2017**, *34*, 21–27. [CrossRef]
23. Vo, H.T.; Mehedy, L.; Mohania, M.; Abebe, E. Blockchain-based Data Management and Analytics for Micro-insurance Applications. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, Singapore, 6–10 November 2017; pp. 2539–2542.
24. Gao, Y.; Nobuhara, H. A Decentralized Trusted Timestamping Based on Blockchains. *IEEJ J. Ind. Appl.* **2017**, *6*, 252–257. [CrossRef]
25. Alter, S. *Information Systems: Foundation of e-Business*; Prentice Hall PTR: Hoboken, NJ, USA, 2001.
26. Liautaud, B.; Hammond, M. *e-Business Intelligence: Turning Information into Knowledge into Profit*; McGraw-Hill, Inc.: New York, NY, USA, 2000.

27. Hofmann, F.; Wurster, S.; Ron, E.; Böhmecke-Schwafert, M. The immutability concept of blockchains and benefits of early standardization. In Proceedings of the 2017 ITU Kaleidoscope: Challenges for a Data-Driven Society (ITU K), Nanjing, China, 27–29 November 2017; pp. 1–8.
28. Ahram, T.; Sargolzaei, A.; Sargolzaei, S.; Daniels, J.; Amaba, B. Blockchain technology innovations. In Proceedings of the 2017 IEEE technology & engineering management conference (TEMSCON), Santa Clara, CA, USA, 8–10 June 2017; pp. 137–141.
29. Badzar, A. Blockchain for Securing Sustainable Transport Contracts and Supply Chain Transparency—An Explorative Study of Blockchain Technology in Logistics. Master's Thesis, Lund University, Lund, Sweden, 2016
30. Heilman, E.; Baldimtsi, F.; Goldberg, S. Blindly signed contracts: Anonymous on-blockchain and off-blockchain bitcoin transactions. In Proceedings of the International Conference on Financial Cryptography and Data Security, Christ Church, Barbados, 22–26 February 2016; pp. 43–60.
31. Wang, J.; Chen, W.; Wang, L.; Sherratt, R. S.; and Alfarraj, O. and Tolba, A. Data secure storage mechanism of sensor networks based on blockchain. *CMC-Comput. Mater. Contin.* **2020**, *36*, 2365–2384. [CrossRef]
32. Zhang, J.; Zhong, S.; Wang, J.; Yu, X.; Alfarraj, O. A storage optimization scheme for blockchain transaction databases. *Comput. Syst. Sci. Eng.* **2021**, *65*, 521–535. [CrossRef]
33. Xu, Z.; Liang, W.; Li, K.C.; Xu, J.; Jin, H. A blockchain-based roadside unit-assisted authentication and key agreement protocol for internet of vehicles. *J. Parallel Distrib. Comput.* **2021**, *149*, 29–39. [CrossRef]
34. Wang, J.; Chen, W.; Ren, Y.; Alfarraj, O.; Wang, L. Blockchain based data storage mechanism in cyber physical system. *J. Internet Technol.* **2020**, *21*, 1681–1689.
35. Zhang, J.; Zhong, S.; Wang, T.; Chao, H.C.; Wang, J. Blockchain-based systems and applications: A survey. *J. Internet Technol.* **2020**, *21*, 1–14.
36. Fritz, M.; Schiefer, G. Tracking, tracing, and business process interests in food commodities: A multi-level decision complexity. *Int. J. Prod. Econ.* **2009**, *117*, 317–329. [CrossRef]
37. Valle e Azevedo, J.; Koopman, S.J.; Rua, A. Tracking the business cycle of the euro area: A multivariate model-based bandpass filter. *J. Bus. Econ. Stat.* **2006**, *24*, 278–290. [CrossRef]
38. Winter, M.; Danes, S.M.; Koh, S.K.; Fredericks, K.; Paul, J.J. Tracking family businesses and their owners over time: Panel attrition, manager departure and business demise. *J. Bus. Ventur.* **2004**, *19*, 535–559. [CrossRef]
39. Liu, C.Z.; Kavakli, M. A data-aware confidential tunnel for wireless sensor media networks. *Multimed. Tools Appl.* **2017**, *78*, 26941–26963. [CrossRef]
40. Al-Bassam, M. SCPKI: A smart contract-based PKI and identity system. In Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts, Abu Dhabi, United Arab Emirates, 2 April 2017; pp. 35–40.
41. Hari, A.; Lakshman, T. The internet blockchain: A distributed, tamper-resistant transaction framework for the internet. In Proceedings of the 15th ACM Workshop on Hot Topics in Networks, Atlanta, GA, USA, 9–10 November 2016; pp. 204–210.
42. Nash, A.; Duane, W.; Joseph, C. *PKI: Implementing and Managing E-Security*; McGraw-Hill, Inc.: New York, NY, USA, 2001.
43. Zyskind, G.; Nathan, O.; Pentland, A.S. Decentralizing privacy: Using blockchain to protect personal data. In Proceedings of the Security and Privacy Workshops (SPW), 2015 IEEE, San Jose, CA, USA, 21–22 May 2015; pp. 180–184.
44. Lemieux, V.L. Trusting records: Is Blockchain technology the answer? *Rec. Manag. J.* **2016**, *26*, 110–139. [CrossRef]
45. Nayak, K.; Kumar, S.; Miller, A.; Shi, E. Stubborn mining: Generalizing selfish mining and combining with an eclipse attack. In Proceedings of the Security and Privacy (EuroS&P), 2016 IEEE European Symposium on IEEE, Saarbrücken, Germany, 21–24 March 2016; pp. 305–320.
46. Zheng, Z.; Xie, S.; Dai, H.; Chen, X.; Wang, H. An overview of blockchain technology: Architecture, consensus, and future trends. In Proceedings of the 2017 IEEE International Congress on Big Data (BigData Congress), Honolulu, HI, USA, 25–30 June 2017; pp. 557–564.
47. Crosby, M.; Nachiappan; Pattanayak, P.; Verma, S.; Kalyanaraman, V. Blockchain technology: Beyond bitcoin. *Appl. Innov.* **2016**, *2*, 71.
48. Decker, C.; Seidel, J.; Wattenhofer, R. Bitcoin meets strong consistency. In Proceedings of the 17th International Conference on Distributed Computing and Networking, Hyderabad, India, 5–7 January 2016; p. 13.
49. Sousa, J.; Bessani, A.; Vukolic, M. A byzantine fault-tolerant ordering service for the hyperledger fabric blockchain platform. In Proceedings of the 2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Luxembourg, 25–28 June 2018; pp. 51–58.
50. Perlman, R.J. Network Layer Protocols with Byzantine Robustness. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 1988.
51. Castro, M.; Liskov, B. Practical Byzantine fault tolerance. In Proceedings of the OSDI, New Orleans, LA, USA, 22–25 February 1999; Volume 99, pp. 173–186.
52. Castro, M.; Liskov, B. Practical Byzantine fault tolerance and proactive recovery. *ACM Trans. Comput. Syst.* **2002**, *20*, 398–461. [CrossRef]
53. Dwork, C.; Naor, M. Pricing via processing or combatting junk mail. In Proceedings of the Annual International Cryptology Conference. Springer, Santa Barbara, CA, USA, 16–20 August 1992; pp. 139–147.
54. Nakamoto, S.; Bitcoin: A Peer-To-Peer Electronic Cash System. 2008. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 20 November 2022).

55. Gervais, A.; Karame, G.O.; Wüst, K.; Glykantzis, V.; Ritzdorf, H.; Capkun, S. On the security and performance of proof of work blockchains. In Proceedings of the 2016 ACM SIGSAC conference on computer and communications security, Vienna, Austria, 24–28 October 2016; pp. 3–16.
56. King, S.; Nadal, S. Ppcoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake. 2012. Available online: https://decred.org/research/king2012.pdf (accessed on 20 November 2022).
57. Li, W.; Andreina, S.; Bohli, J.M.; Karame, G. Securing proof-of-stake blockchain protocols. In *Data Privacy Management, Cryptocurrencies and Blockchain Technology*; Springer: Cham, Switzerland, 2017; pp. 297–315.
58. De Angelis, S.; Aniello, L.; Baldoni, R.; Lombardi, F.; Margheri, A.; Sassone, V. Pbft vs proof-of-authority: Applying the cap theorem to permissioned blockchain. In Proceedings of the Italian Conference on Cyber Security, Milan, Italy, 6 February 2018.
59. Vujičić, D.; Jagodić, D.; Randic, S. Blockchain technology, bitcoin, and Ethereum: A brief overview. In Proceedings of the 2018 17th International Symposium INFOTEH-JAHORINA (INFOTEH), East Sarajevo, Bosnia and Herzegovina, 21–23 March 2018; pp. 1–6.
60. King, S. Primecoin: Cryptocurrency with Prime Number Proof-of-Work. 2013. Available online: https://primecoin.io/primecoin-paper.pdf (accessed on 20 November 2022).
61. Vukolić, M. The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication. In Proceedings of the International Workshop on Open Problems in Network Security, Zurich, Switzerland, 29 October 2015; pp. 112–125.
62. Upadhyay, N. Rule of Code. In *UnBlock the Blockchain*; Springer: Singapore, 2019; pp. 11–24.
63. Saleh, F. Blockchain without waste: Proof-of-stake. *Rev. Financ. Stud.* **2021**, *34*, 1156–1190. [CrossRef]
64. Ren, L. Proof of Stake Velocity: Building the Social Currency of the Digital Age. 2014. Available online: https://cryptochainuni.com/wp-content/uploads/Reddcoin-Proof-of-Stake-Velocity.pdf (accessed on 20 November 2022).
65. David, B.; Gaži, P.; Kiayias, A.; Russell, A. Ouroboros praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, 29 April–3 May 2018; pp. 66–98.
66. Gaži, P.; Kiayias, A.; Zindros, D. Proof-of-stake sidechains. In Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP), Francisco, CA, USA, 20–22 May 2019; pp. 139–156.
67. Gao, Y.; Nobuhara, H. A proof of stake sharding protocol for scalable blockchains. *Proc. Asia-Pac. Adv. Netw.* **2017**, *44*, 13–16. [CrossRef]
68. Poelstra, A. Distributed Consensus from Proof of Stake is Impossible. 2014. Available online: https://epicfundsllc.com/wp-content/uploads/2020/12/old-pos.pdf (accessed on 20 November 2022).
69. Lin, I.C.; Liao, T.C. A Survey of Blockchain Security Issues and Challenges. *IJ Netw. Secur.* **2017**, *19*, 653–659.
70. Kanjalkar, S.; Kuo, J.; Li, Y.; Miller, A. Short Paper: I Can't Believe It's Not Stake! Resource Exhaustion Attacks on PoS. In Proceedings of the International Conference on Financial Cryptography and Data Security, St. Kitts, Saint Kitts and Nevis, 18–22 February 2019.
71. Brown-Cohen, J.; Narayanan, A.; Psomas, A.; Weinberg, S.M. Formal barriers to longest-chain proof-of-stake protocols. In Proceedings of the 2019 ACM Conference on Economics and Computation, Phoenix, AZ, USA, 24–28 June 2019; pp. 459–473.
72. Jain, A.; Arora, S.; Shukla, Y.; Patil, T.; Sawant-Patil, S. Proof of stake with casper the friendly finality gadget protocol for fair validation consensus in ethereum. *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.* **2018**, *3*, 291–298.
73. Deirmentzoglou, E.; Papakyriakopoulos, G.; Patsakis, C. A Survey on Long-Range Attacks for Proof of Stake Protocols. *IEEE Access* **2019**, *7*, 28712–28725. [CrossRef]
74. Zhang, S.; Lee, J.H. Eclipse-based Stake-Bleeding Attacks in PoS Blockchain Systems. In Proceedings of the 2019 ACM International Symposium on Blockchain and Secure Critical Infrastructure, Auckland, New Zealand, 8 July 2019. 2019, pp. 67–72.
75. Gaži, P.; Kiayias, A.; Russell, A. Stake-bleeding attacks on proof-of-stake blockchains. In Proceedings of the 2018 Crypto Valley Conference on Blockchain Technology (CVCBT), Zug, Switzerland, 20–22 June 2018; pp. 85–92.
76. Kiayias, A.; Russell, A.; David, B.; Oliynykov, R. Ouroboros: A provably secure proof-of-stake blockchain protocol. In Proceedings of the Annual International Cryptology Conference, Santa Barbara, CA, USA, 20–24 August 2017; pp. 357–388.
77. Kiayias, A.; Konstantinou, I.; Russell, A.; David, B.; Oliynykov, R. A Provably Secure Proof-of-Stake Blockchain Protocol. *IACR Cryptol. EPrint Arch.* **2016**, *2016*, 889.
78. Bachani, V. and Bhattacharjya, A., Preferential Delegated Proof of Stake (PDPoS)—Modified DPoS with Two Layers towards Scalability and Higher TPS. *Symmetry* **2023**, *15*, 118541–118555.
79. Yang, F.; Zhou, W.; Wu, Q.; Long, R.; Xiong, N.N.; Zhou, M. Delegated Proof of Stake With Downgrade: A Secure and Efficient Blockchain Consensus Algorithm With Downgrade Mechanism. *IEEE Access* **2019**, *7*, 118541–118555. [CrossRef]
80. Bentov, I.; Lee, C.; Mizrahi, A.; Rosenfeld, M. Proof of Activity: Extending Bitcoin's Proof of Work via Proof of Stake. *IACR Cryptol. EPrint Arch.* **2014**, *2014*, 452.
81. Network, P. Proof of Authority: Consensus Model with Identity at Stake. *Medium*. Available online: https://medium.com/poa-network/proof-of-authority-consensus-model-with-identity-at-stake-d5bd15463256 (accessed on 12 November 2017).
82. A Avasthi, A.; Saxena, A. Two Hop Blockchain Model: Resonating between Proof of Work (PoW) and Proof of Authority (PoA). *Int. J. Inf. Syst. Manag. Sci.* **2018**, *1*, 1–4.
83. Milutinovic, M.; He, W.; Wu, H.; Kanwal, M. Proof of luck: An efficient blockchain consensus protocol. In Proceedings of the proceedings of the 1st Workshop on System Software for Trusted Execution, Trento, Italy, 12–16 December 2016; p. 2.

84. Ogawa, T.; Kima, H.; Miyaho, N. Proposal of Proof-of-Lucky-Id (PoL) to Solve the Problems of PoW and PoS. In Proceedings of the 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Halifax, NS, Canada, 30 July–3 August 2018; pp. 1212–1218.
85. Gupta, D.; Mood, B.; Feigenbaum, J.; Butler, K.; Traynor, P. Using intel software guard extensions for efficient two-party secure function evaluation. In Proceedings of the International Conference on Financial Cryptography and Data Security, West Indies, Barbados, 22–26 February 2016; pp. 302–318.
86. McKeen, F.; Alexandrovich, I.; Anati, I.; Caspi, D.; Johnson, S.; Leslie-Hurd, R.; Rozas, C. Intel® software guard extensions (intel® sgx) support for dynamic memory management inside an enclave. In Proceedings of the Hardware and Architectural Support for Security and Privacy 2016, Seoul, Republic of Korea, 18 June 2016; p. 10.
87. Pires, R.; Pasin, M.; Felber, P.; Fetzer, C. Secure content-based routing using intel software guard extensions. In Proceedings of the 17th International Middleware Conference, Trento, Italy, 12–16 December 2016; p. 10.
88. Chen, L.; Xu, L.; Shah, N.; Gao, Z.; Lu, Y.; Shi, W. On security analysis of proof-of-elapsed-time (poet). In Proceedings of the International Symposium on Stabilization, Safety, and Security of Distributed Systems, Boston, MA, USA, 5–8 November 2017; pp. 282–297.
89. Corso, A. Performance Analysis of Proof-of-Elapsed-Time (PoET) Consensus in the Sawtooth Blockchain Framework. Master's Thesis, University of Oregon, Eugene, OR, USA, 2019.
90. Ateniese, G.; Bonacina, I.; Faonio, A.; Galesi, N. Proofs of space: When space is of the essence. In Proceedings of the International Conference on Security and Cryptography for Networks, Amalfi, Italy, 3–5 September 2014; pp. 538–557.
91. Dziembowski, S.; Faust, S.; Kolmogorov, V.; Pietrzak, K. Proofs of space. In Proceedings of the Annual Cryptology Conference, Santa Barbara, CA, USA, 16–20 August 2015; pp. 585–605.
92. Ren, L.; Devadas, S. Proof of space from stacked expanders. In Proceedings of the Theory of Cryptography Conference, Beijing, China, 31 October–3 November 2016; pp. 262–285.
93. Moran, T.; Orlov, I. Proofs of Space-Time and Rational Proofs of Storage. *IACR Cryptol. EPrint Arch.* **2016**, *2016*, 35.
94. Moran, T.; Orlov, I. *Rational Proofs of Space-Time*; Technol Report; Bar-Ilan University Cyber Center: Ramat Gan, Israel, 2017.
95. Schwartz, D.; Youngs, N.; Britto, A. *The Ripple Protocol Consensus Algorithm*; White Paper; Ripple Labs Inc.: San Francisco, CA, USA, 2014; Volume 5, p. 8. Available online: https://ripple.com/files/ripple_consensus_whitepaper.pdf (accessed on 12 November 2017).
96. D'Agostino, S.F.; Timpanaro, J.P. Ripple Protocol performance improvement: Small world theory applied to cross border payments. In Proceedings of the XIX Simposio Argentino de Ingeniería de Software (ASSE)-JAIIO 47 (CABA, 2018), La Plata, Buenos Aires, 3–7 September 2018.
97. Dillenberger, D.E.; Su, G. Parallel Execution of Blockchain Transactions. U.S. Patent 10,255,108, 9 April 2019.
98. Innerbichler, J.; Damjanovic-Behrendt, V. Federated byzantine agreement to ensure trustworthiness of digital manufacturing platforms. In Proceedings of the 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems, Munich, Germany, 15 June 2018; pp. 111–116.
99. Mazieres, D. *The Stellar Consensus Protocol: A Federated Model for Internet-Level Consensus*; Stellar Development Foundation: San Francisco, CA, USA, 2015; p. 32.
100. Lamport, L.; Shostak, R.; Pease, M. The Byzantine generals problem. *ACM Trans. Program. Lang. Syst.* **1982**, *4*, 382–401. [CrossRef]
101. Lamport, L. The weak Byzantine generals problem. *J. ACM* **1983**, *30*, 668–676. [CrossRef]
102. Dolev, D. The Byzantine generals strike again. *J. Algorithms* **1982**, *3*, 14–30. [CrossRef]
103. Donet, J.A.D.; Pérez-Sola, C.; Herrera-Joancomartí, J. The bitcoin P2P network. In Proceedings of the International Conference on Financial Cryptography and Data Security, Christ Church, Barbados, 3–7 March 2014; pp. 87–102.
104. Biryukov, A.; Khovratovich, D.; Pustogarov, I. Deanonymisation of clients in Bitcoin P2P network. In Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, Scottsdale, Arizona, USA, 3–7 November 2014; pp. 15–29.
105. Palomar, E.; Tapiador, J.M.; Hernandez-Castro, J.C.; Ribagorda, A. Secure content access and replication in pure p2p networks. *Comput. Commun.* **2008**, *31*, 266–279. [CrossRef]
106. DuPont, Q. Experiments in algorithmic governance: A history and ethnography of "The DAO", a failed decentralized autonomous organization. In *Bitcoin and Beyond*; Routledge: London, UK, 2017; pp. 157–177.
107. Reijers, W.; Wuisman, I.; Mannan, M.; De Filippi, P.; Wray, C.; Rae-Looi, V.; Vélez, A.C.; Orgad, L. Now the code runs itself: On-chain and off-chain governance of blockchain technologies. *Topoi* **2018**, *40*, 821–831. [CrossRef]
108. Mehar, M.I.; Shier, C.L.; Giambattista, A.; Gong, E.; Fletcher, G.; Sanayhie, R.; Kim, H.M.; Laskowski, M. Understanding a revolutionary and flawed grand experiment in blockchain: The DAO attack. *J. Cases Inf. Technol.* **2019**, *21*, 19–32. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.