# Blockchain Consensus Algorithms: A Survey

Pooja Khobragade[1] and Ashok Kumar Turuk[2]

[1] NIT Rourkela, Rourkela Odisha, India
[2] NIT Rourkela, Rourkela Odisha, India
{520cs1004,akturuk}@nitrkl.ac.in

**Abstract.** Blockchain is one of the emerging technologies based on Distributed peer-to-peer networking. Blockchain gained popularity in 2009 with the launch of Bitcoin. Since, then blockchain has found applications in various domains such as finance, supply chain, health care, agriculture, pharmacy, IOT, automobile, energy, and many more. Blockchain is decentralized in nature which providing an immutable and tamper-proof ledger of transactions that includes data integrity and security. The consensus algorithm is a common agreement between block nodes to become a part or publish a new block in the blockchain. In this study, we survey different consensus algorithms reported in the literature.

**Keywords:** Blockchain · Consensus Algorithm · PoW· PoS · DPoS · PoET · PBFT · PoC · PoB · PoA · VR · Raft · Paxos.

## 1 Introduction

Blockchain is an emerging technology that gained popularity with its first application in Bitcoin. It is a decentralized, distributed, tamper-proof technology. The problem of the centralize system such as single point of failure and data integrity, can be overcome by Blockchain. It provides a trusted environment for participants. Peers in blockchain share information over the network [1]. Blockchain is distributed digital ledger of cryptographically signed transaction groups together to make a chain like structure called blockchain [2]. Transactions are stored in a chronological order with a time stamp assigned to each block. Transactions are continuously growing, and new blocks are added to the blockchain, with the consensus of other block nodes. According to the application scenario Blockchain is broadly classified into two categories as follows: permission-less blockchain is when participants can join the network without permission or prior authority. However in a permissioned blockchain only authorized users can join the network [3]. Blocks are created and added to the existing chain of blocks through a consensus protocol, which is agreed upon among its peers. In this work, we have surveyed different blockchain consensus algorithms reported in the literature.

The consensus protocol can be broadly classified into two types: i) Byzantine fault-tolerant (BFT) and ii) Crash fault-tolerant (CFT) as shown in Fig. 1. CFT consensus builds a degree of resiliency, and they are mainly used in an environment where nodes with a certain degree of closure and credibility. It

solves consistency problems such as process crashes, network failure, machine downtime, etc. The consensus is reached even if some network components does not work properly. CFT does not guarantee to provide security under malicious activity. However, the BFT consensus algorithm provides a solution to the consistency problem as well as deals with the malicious node and assures security in the network. The fault tolerance capability of nodes is higher than the CFT algorithm [4].

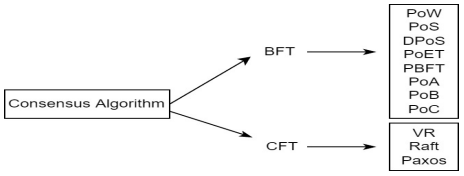


Fig. 1: Classification of consensus algorithm

The rest of the paper is organized as follows: Section 2 explains the need for consensus among the peers in blockchain. Section 3 describes the different consensus algorithms and analyses the working of the consensus algorithm. Section 4 concludes the paper.

## 2 Consensus Protocol

Blockchain is a distributed decentralized network. The consensus protocol is a general agreement among all nodes to synchronize the network transaction [5]. In blockchain, all nodes perform a series of operations, and it is difficult to get consistency between nodes.

Six properties of the consensus algorithm (i) **Cooperation:** Nodes should work as a whole group for the common benefit of the group, rather than individual benefit. (ii) **Collaboration:** All nodes in the system should have a shared goal and work together to achieve the same objective. (iii) **Inclusive:** Maximum number of participants from the group should be done. (iv) **Agreement seeking:** Obtain as much as agreement from the individual nodes in the system. (v) **Participatory:** Active group members participation is required to get success and (vi) **Democratic:** Each node in the system casts an equal-weighted vote. Fig. 2 shows the property of the consensus algorithm.

No central authority is present in the network, but all the transactions are secure and valid because of the consensus algorithm. The primary purpose of the consensus algorithm is to provide security to the blockchain network [6]. The goal of the consensus algorithm is to reach a joint agreement in terms of a network transaction [7]. A Consensus algorithm also solves: (i) The consistency problem is that multiple nodes try to perform a series of operations. It isn't easy to obtain the results saved at each node to reach consistency. (ii) Solves

Byzantine's general problem of reaching the common agreement in a distributed network with possible malicious nodes. (iii)Activeness refers to the nodes in the blockchain network that are active and participate in the consensus algorithm and provide adequate computing power to the blockchain network. (4)Prevent the double-spending problem.


Fig. 2: Properties of consensus algorithm[8]

### 2.1 Goals of consensus algorithm

The goals of the consensus algorithms are: (i) **Termination**: For every transaction, there are only two states, accept or reject. Honest nodes either accept or reject the transaction. (ii) **Agreement**: At every new transaction, honest nodes accept or reject the transaction. If all honest node accepts transactions, it stored in the same sequence in all nodes in the blockchain network. (iii) **Validity**: Only valid transactions are accepted by the nodes that become part of the network. (iv) **Integrity**: All accepted transactions generate valid hash chains, so the nodes are consistence with each other to provide integrity to the network.

### 2.2 Components of consensus algorithm

There are five key components of the consensus algorithm [4]. They are: (i) **Block Proposal**: Where the validators or miners select the next proposal for a block, (ii) **Information Propagation**: The process of verifying the proposed block, which is verified by all the selected nodes, (iii) **Block validation**: Full nodes distribute all the transaction information across the whole network, (iv) **Block Finalization**: Validators reaches to common agreement either accept or reject the block. (v) **Incentive Mechanism**: Miner who follows all the rules get rewards, and those who break the rule get penalties. The above prevents malicious activity in the network.

## 3 Types of Consensus Protocol

Different consensus algorithms provide different capabilities like storage, computing power, and other configurations [9].

### 3.1 Proof-of-Work (PoW)

Proof-of-Work (PoW) was proposed by Dwork and Moni in 1993 [10]. However, it is used by "Satoshi Nakamoto" in his application for Bitcoin uses PoW consensus algorithm [11]. PoW is required to solve a complex problem. The node that can solve the problem obtains the right to add a new block into the blockchain.

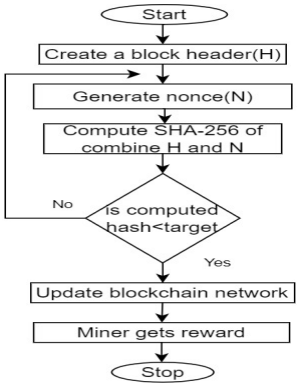


Fig. 3: PoW consensus process

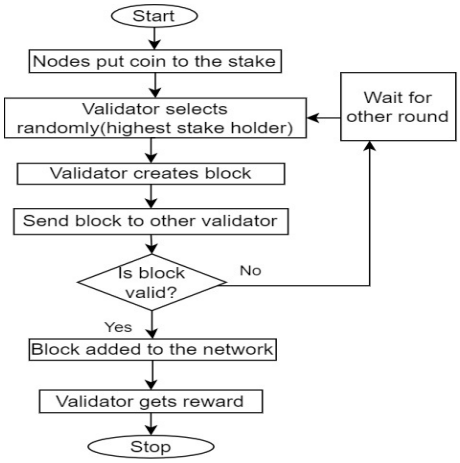


Fig. 4: PoS consensus process

Fig. 3 shows the flowchart of the PoW consensus process. A miner compute the SHA256 of a block header which contains a fixed value and a variable value (nonce). The fixed value is computed apriori from the transaction information in all blocks. The miner obtains all rights to add a block to the blockchain network, if the computed value is less than the target value. For computed value greater than the target value, the value of nonce is changed, and the hash of the header is computed. The above process continues until the header's computed hash value is less than the target value. Solving the problem is an intensive task. Nodes adjust the nonce value and compute the hash of the header until it is less than the target value. To modify a block, an attacker must redo the block's PoW and all the blocks after it [12].

### 3.2 Proof-of-Stake (PoS)

In PoW, nodes invest their resources and computation power in solving a complex problem. PoW algorithm requires a high computation of power for mining, which leads to increased energy usage. Moreover, the transaction rate of PoW is low. To overcome the limitations of PoW, King and Nadal proposed Proof-of-Stake (PoS). In PoS, nodes put a certain coin at stake to become a part of the validation process. The more a node has a stake, the higher the chance of becoming a validator [12]. The validator is chosen pseudo-randomly and becomes

a part of the consensus algorithm [13]. A node having the highest stake can monopolize the validation process. Fig. 4 shows the flowchart of the PoS consensus process.

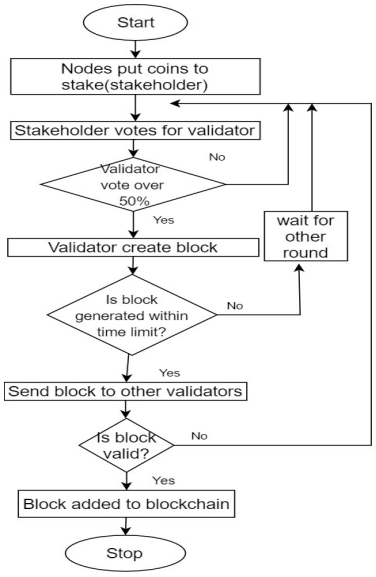


Fig. 5: DPoS consensus Process

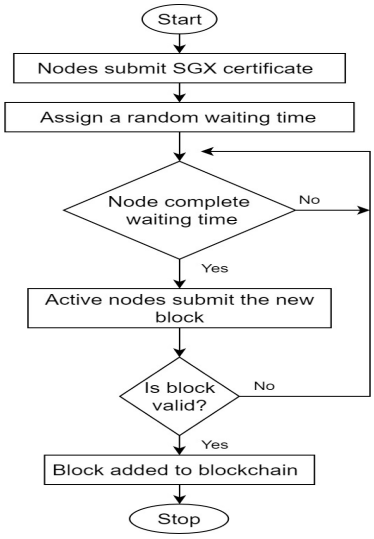


Fig. 6: PoET consensus process

### 3.3 Delegated Proof-of-Stake (DPoS)

Delegated Proof-of-Stake (DPoS) is more energy-efficient compared to PoW and PoS. In PoS, the highest stakeholder can control all validation processes. To overcome the chance of centralization in PoS and enhance security, DPoS is proposed. In DPoS, the validator is voted by stakeholders for producing a new block. The number of votes allocated to a participatory node depends upon the number of currencies held by the node. Nodes participating in the voting process are the decision-makers in the consensus algorithm [14]. The time limit is decided for each delegated node. If the delegated node cannot generate a block within the allocated time limit, then the delegated node is dismissed. The stakeholders choose a new delegate node, and the next round of block creation begins. Fig.5 shows the flowchart of the DPoS consensus process.

### 3.4 Practical Byzantine Fault Tolerance (PBFT)

Practical Byzantine Fault Tolerance (PBFT) was proposed by Liskov and Castro [15] to reduce the algorithm's complexity to a polynomial-time of Byzantine general problems. PBFT has different phases: pre-preparation, preparation,

submission, and reply. In the pre-preparation phase, the master node sends information to all the nodes. In the preparation phase, nodes receive information and send it to other nodes except themselves. In the submission phase, all nodes receive 2f+1 information, where 2f is the number of honest nodes, and one is the master node. In the reply phase master gets a reply only from an honest node. Fig. 7 shows the PBFT consensus process.

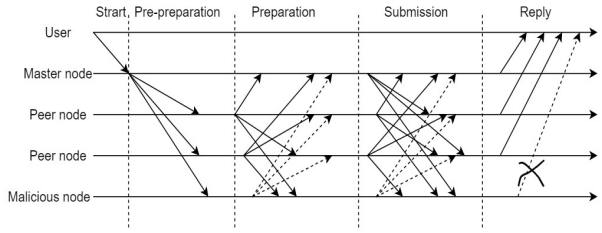


Fig. 7: PBFT consensus process

### 3.5 Raft

In the Raft algorithm, a node uses the log replication of other nodes to maintain a unified transaction. Nodes are divided into three categories: leader, follower, and candidate. The leader is responsible for interactive communication. Followers become voters in the voting activity, and candidates are transformed from followers and can be part of the leader selection process [16]. Fig. 8 shows the Raft consensus process.

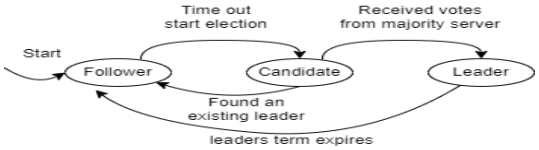


Fig. 8: Raft Consensus Algorithm

### 3.6 Proof of Elapsed Time (PoET)

The Proof-of-Elapsed-Time (PoET) is similar to PoW. It requires fewer computational resources. In PoET a separate timer is attached to each node, which allocates a random waiting time for the node. Every miner gets a random and fair waiting time and decides which miner to publish a block. The miner whose waiting time finished first gets a chance to publish the next block and broadcast it to the network. Fig. 6 shows the flowchart of the PoET consensus process.

### 3.7 Proof of Activity(PoA)

Proof-of-activity (PoA) combines the PoW and PoS consensus algorithm. PoA works similar to PoW but with less complexity. In PoA, the miner solves a cryptographic puzzle-like PoW and claims for reward, then shifts to PoS [17]. The block does not contain transactions, instead contains templates that include header information and reward address. The header information is used to select the validator randomly. The more the stake value a validator has, more is its chance to selected and sign the block. Once block is signed by all validator, it become a part of the blockchain. If a chosen validator does not sign the block, then the block is discarded, and the next block with a high stake will be use. Fig.9 shows the flowchart of the PoA consensus process.

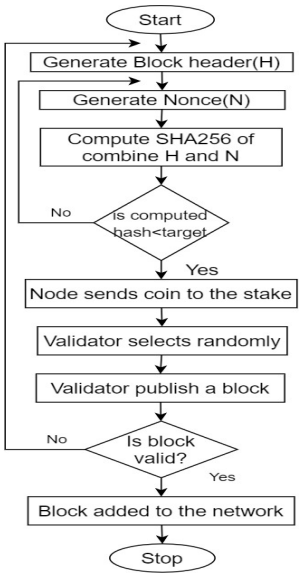


Fig. 9: PoA consensus process

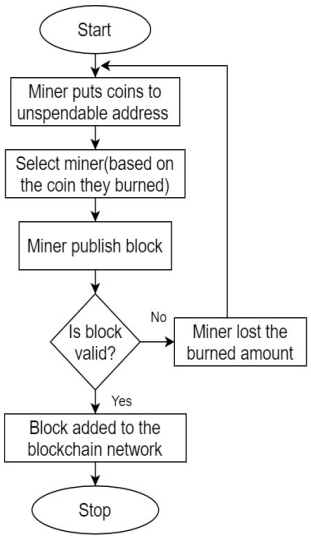


Fig. 10: PoB consensus process

### 3.8 Proof of Burn (PoB)

The Proof-of-burn (PoB) consensus algorithm is a mechanism to burn the coin in a verifiable manner to generate a new coin [18]. The burned coin is destroyed forever. It requires less energy than PoW because it burns its coin instead of burning energy. Miners put their coin in burn address or unspendable address. The coin in this address is not immediately destroyed, but the miner can not spend this coin anywhere else. The higher the amount spent by a miner, greater the chance of being chosen to mine a block. If the miner can publish a valid block, it gets a reward. If not, the miner only wastes the coin. The unspendable

address is the blockchain address with no private key, and burn coins are locked in this address and lost forever. Fig.10 shows the flowchart of PoB consensus process.

### 3.9 Proof of capacity (PoC)

Proof-of-capacity (PoC) overcomes the limitation of PoW, PoS, and PoB. It requires less storage space and consumes less energy as compared to PoW, PoS, and PoB. PoC works in two phases: plotting and mining [19]. In PoC, before mining a block, the miner stores the list of possible solutions for the correct block. Solutions are stored at free disk storage. Nodes use these solutions to plot a graph. The more solutions a node stores, the higher its chance of becoming a block's next publisher .The higher capacity holder wins the consensus. After plotting miner start the mining process. Fig 11 shows the process flowchart of the PoC consensus process.

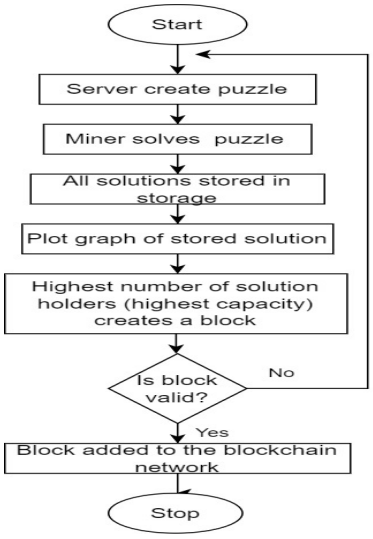


Fig. 11: PoC consensus process

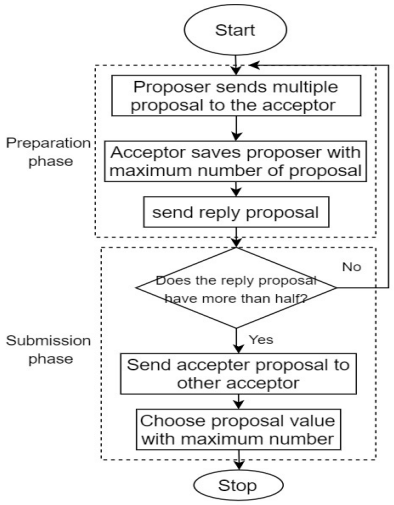


Fig. 12: Paxos consensus process

### 3.10 Viewstamped Replication (VR)

Viewstamped replication (VR) is a replication protocol rather than a consensus protocol. It performs log replication of nodes to get consistency in the network. In VR, one node is selected as a primary node, and the others are backup nodes [20]. The Primary node decides the sequence of commands. The Backup node

follows the sequence and executes the command in order. If the primary node fails, the VR algorithm performs a view change operation, and selects the next primary node from the voting process. The next node becomes the primary node if it gets f+1 votes from a total of 2f nodes. The new primary node starts a new consensus process and the old primary node goes to a recovery state and cannot perform any operation.

### 3.11 Paxos

Paxos solves network-related problem such as message delay, machine down-time, network failure, data loss etc. Nodes are divided into three categories: proposer, acceptor, and learner. A node can belong to more than one category. Proposer sent a proposal to the network. The acceptor votes on specified proposal and learner collects all the accepted proposals from the acceptor. Paxos operates in two phase: The preparation phase and the submission phase. In the preparation phase, the proposer sends multiple proposals with proposal number to acceptor. The acceptor compares all proposals and stores the proposal, and returns the maximum proposals value sent by the proposer, and discards the proposer who has proposal number less than a maximum number of proposal. In submission phase, if a proposer receives more than 1/2 of acceptance out of the total number of acceptor, proposer sends the received proposal to all other acceptors and continues the preparation phase. A consensus is reached when more than half of the recipients have the same proposal value [21]. Fig. 12 shows the process flow of Paxos consensus Process.

## 4 Methodology

The following metrics are constructed for the comparison of the blockchain consensus algorithm: Throughput, Energy consumption, scalability and cost. Throughput is the maximum agreement rate to verify the transaction expressed as the number of transactions per second (TPS). In a decentralized system, if any new block is arrived there is some time gap, for all nodes to agree on it. If there is no time gap, many blocks arrive at some fraction of delay and do not get any optimization benefits, and it just becomes a chain of transactions. Maximum throughput means the maximum rate at which the blockchain confirms a transaction. Blockchain scalability is another essential factor in consensus algorithms, and higher scalability means the blockchain can achieve more transaction per second. Blockchain can achieve high throughput by modifying existing consensus algorithms or changing other parameters [3]. Energy consumption is the next important factor in the consensus algorithm.The amount of energy consume by the blockchain network is depends on it's consensus mechanism. Two factors contributing to the cost of consensus algorithm are: transaction fees and transaction verification time. Nodes pay transaction fees to the miner to verify their transaction. Not all cryptocurrencies require transaction fees. Blockchain

Table 1: Comparative analysis of consensus algorithm

| **Blockchain type** | **Consensus algorithm** | **Energy consumption** | **Throughput (transaction per second)** | **Scalability** | **Cost** | **Example of representative blockchain** |
|---|---|---|---|---|---|---|
| Permission less | PoW | High | Low | Low | High | Bitcoin, Ethereum |
| | PoS | Medium | Low | Medium | Medium | Peercoin, Ethereum |
| | DPoS | Medium | High | Medium | Low | Bitshares, Steem and steemit |
| | PoA | High | High | Low | High | Decred, Espers |
| | PoB | Low | Medium | Low | Medium | Bitcoin, Slimcoin |
| | PoC | Low | High | Medium | High | Burstcoin |
| Permissioned | PBFT | Low | High | Low | Low | Hyperledger Fabric, Zilliqa |
| | PoET | Low | Medium | High | High | Hyperledger, Sawtooth |
| | VR | Low | High | Low | Low | - |
| | Paxos | Low | Medium | Low | High | etcd, Kubernetes |
| | Raft | Low | High | High | Medium | Quorum etcd, Kubernetes |

uses some methods to eliminate transaction fees. Table 1 shows a comparison of different consensus algorithms.

Consensus is a set of rules to reach a decision in a decentralized environment. Without consensus, the blockchain is reduced to a platform for storing encrypted data. From the Table 1, we can conclude that energy consumption is always low for permissioned blockchain. Permissionless blockchain chains mostly faced scalability issues. The blockchain cost depends upon the transaction fee and transaction storage. If energy consumption and storage demand are high, the cost is also high.

## 5 Conclusion

Consensus is one of the vital technology in blockchain. Over the period, many consensus algorithms have been developed to solve security, scalability, energy consumption, and fault tolerance. The consensus algorithms have their limitation in terms of throughput. Improvement in the consensus algorithm can significantly improve the overall performance of the blockchain network. The consensus algorithm applies on blockchain depending on the network type and application scenario. This paper summarizes the popular blockchain consensus algorithm. Choosing a suitable consensus algorithm is essential for making an efficient blockchain network.

## References

1. Lakshmi Siva Sankar, M Sindhu, and M Sethumadhavan. Survey of consensus protocols on blockchain applications. In *2017 4th international conference on advanced computing and communication systems (ICACCS)*, pages 1–5. IEEE, 2017.
2. Dylan Yaga, Peter Mell, Nik Roby, and Karen Scarfone. Blockchain technology overview. *arXiv preprint arXiv:1906.11078*, 2019.
3. Abdurrashid Ibrahim Sanka, Muhammad Irfan, Ian Huang, and Ray CC Cheung. A survey of breakthrough in blockchain technology: Adoptions, applications, challenges and future research. *Computer Communications*, 2021.

4. Yang Xiao, Ning Zhang, Wenjing Lou, and Y Thomas Hou. A survey of distributed consensus protocols for blockchain networks. *IEEE Communications Surveys & Tutorials*, 22(2):1432–1465, 2020.
5. Weiwei Gu, Jianan Li, and Zekai Tang. A survey on consensus mechanisms for blockchain technology. In *2021 International Conference on Artificial Intelligence, Big Data and Algorithms (CAIBDA)*, pages 46–49. IEEE, 2021.
6. Giang-Truong Nguyen and Kyungbaek Kim. A survey about consensus algorithms used in blockchain. *Journal of Information processing systems*, 14(1):101–128, 2018.
7. Yang Xiao, Ning Zhang, Jin Li, Wenjing Lou, and Y Thomas Hou. Distributed consensus protocols and algorithms. *Blockchain for Distributed Systems Security*, 25:40, 2019.
8. Jayapriya Jayabalan and N Jeyanthi. A study on distributed consensus protocols and algorithms: the backbone of blockchain networks. In *2021 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1–10. IEEE, 2021.
9. Md Sadek Ferdous, Mohammad Jabed Morshed Chowdhury, Mohammad A Hoque, and Alan Colman. Blockchain consensus algorithms: A survey. *arXiv preprint arXiv:2001.07091*, 2020.
10. Ashok Kumar Yadav and Karan Singh. Comparative analysis of consensus algorithms of blockchain technology. In *Ambient communications and computer systems*, pages 205–218. Springer, 2020.
11. Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. *Decentralized Business Review*, page 21260, 2008.
12. Ranjith Kumar Rama. Overview of blockchain technology: Consensus algorithms, applications.
13. Shijie Zhang and Jong-Hyouk Lee. Analysis of the main consensus protocols of blockchain. *ICT express*, 6(2):93–97, 2020.
14. Fan Yang, Wei Zhou, QingQing Wu, Rui Long, Neal N Xiong, and Meiqi Zhou. Delegated proof of stake with downgrade: A secure and efficient blockchain consensus algorithm with downgrade mechanism. *IEEE Access*, 7:118541–118555, 2019.
15. Miguel Castro, Barbara Liskov, et al. Practical byzantine fault tolerance. In *OsDI*, volume 99, pages 173–186, 1999.
16. Yaqin Wu, Pengxin Song, and Fuxin Wang. Hybrid consensus algorithm optimization: A mathematical method based on pos and pbft and its application in blockchain. *Mathematical Problems in Engineering*, 2020, 2020.
17. Iddo Bentov, Charles Lee, Alex Mizrahi, and Meni Rosenfeld. Proof of activity: Extending bitcoin's proof of work via proof of stake [extended abstract] y. *ACM SIGMETRICS Performance Evaluation Review*, 42(3):34–37, 2014.
18. Kostis Karantias, Aggelos Kiayias, and Dionysis Zindros. Proof-of-burn. In *International conference on financial cryptography and data security*, pages 523–540. Springer, 2020.
19. Shubhani Aggarwal and Neeraj Kumar. Cryptographic consensus mechanisms. In *Advances in Computers*, volume 121, pages 211–226. Elsevier, 2021.
20. Brian M Oki and Barbara H Liskov. Viewstamped replication: A new primary copy method to support highly-available distributed systems. In *Proceedings of the seventh annual ACM Symposium on Principles of distributed computing*, pages 8–17, 1988.
21. Huanliang Xiong, Muxi Chen, Canghai Wu, Yingding Zhao, and Wenlong Yi. Research on progress of blockchain consensus algorithm: A review on recent progress of blockchain consensus algorithms. *Future Internet*, 14(2):47, 2022.