A Comparative Study of the Recent Blockchain Consensus Algorithms

Nassime El Rharbi[1*], Hicham Atteriuas[1], Ali Younes[1], Abdelali Harchaoui[1], Oumaima Izem[1]

[1,2,3,4,5*]CSSE LABS, Faculty of scineces, Tetouan, 93002, Tetouan-Tangier-AlHoceima, Morocco.

*Corresponding author(s). E-mail(s): Nassime.elrharbi.etu.uae.as.ma;
Contributing authors: attariuas.hicham@gmail.com; ayounes@uae.ac.ma; harchaoui.abdelali@gmail.com; oumaima.izem@etu.uae.ac.ma;

Abstract

As Blockchain gains traction, it can be said that it is the primary focus of many computer science studies due to its enormous impact on thousands of projects and organizations, which prompted them to completely alter their approach to storing and processing large quantities of data. Despite attempts by detractors to undermine its durability, security, and stability. Consensus is required for the Blockchain network to be completely secure and decentralized. In brief, consensus is the process of approving or rejecting network transactions. Our work will consist of a statistical analysis and a comprehensive overview of the most recent consensus algorithms, comparing their throughput speed, scalability of the network, security resilience against a variety of attacks, decentralization of the database, and the amount of resources required each time a block is added. All of this is intended to provide a comprehensive understanding of these algorithms and their respective use cases.

Keywords: Blockchain, Consensus, Scalability, Mining, Decentralization, Consensus algorithms, Private and Public Blockchain.

1 Introduction

In recent years, the news of cryptocurrencies such as Bitcoin and Etherium, as well as Smart Contracts, has had a significant global impact on Blockchain technology. Most

M. Khaldi et al. (eds.), *Proceedings of the E-Learning and Smart Engineering Systems (ELSES 2023)*, Atlantis Highlights in Social Sciences, Education and Humanities 14,
https://doi.org/10.2991/978-94-6463-360-3_32

people know Blockchain systems from the development of bitcoin, as Satoshi Nakamoto published a white paper in 2008 introducing his ideas to the world as the first peer-to-peer transferable currency similar to actual cash that could be used without the need for a central bank or other controlling body to manage the ledger. However, until Blockchain 2.0 was introduced, there had been no actual applications of the technology other than currency. (Cross-Industry Applications of Blockchain Technology and Future Opportunities) A Blockchain is a distributed, decentralized digital ledger that logs transactions across several computers. It operates by storing information in several blocks that are cryptographically connected and protected. A network of nodes (computers) verifies and appends the new transaction to the block. [4] When a block is filled, it is connected to the block before it to form a chain. This procedure ensures that the data is accurate, legible, and unaltered. To modify a block, you would have to modify all the blocks that follow it and receive network approval, which is very impossible. A Blockchain is a collection of data-storing blocks created in 1991 by a group of researchers to protect digital data from being altered or tampered with. The operation of a Blockchain depends on three components: data, hash, and the hash of the preceding block. Data includes transaction details such as sender, recipient, and transaction amount, while hashing helps detect changes in a Blockchain. Previously hashed data includes the hash of the preceding block, which contributes to the construction of a chain and is incredibly trustworthy and secure. Modern computers are capable of storing hundreds of thousands of hashes, and to restore the Blockchain's validity, the hashes of tampered blocks and other blocks are modified by default. [2] Blockchain technology offers a vast array of application options, with multiple Blockchain varieties catering to unique requirements. Public Blockchains provide platforms that are accessible, transparent, and safe, but private and consortium Blockchains give restricted access and more privacy. For complicated use cases, hybrid and federated Blockchains provide a flexible and interoperable solution. Knowing the distinctive characteristics and possible applications of each Blockchain type is vital for enterprises seeking to leverage the disruptive power of this technology.

2 Consensus algorithms

Consensus algorithms are vital for preserving the consistency and dependability of distributed systems.In this conference paper we will conclude an introduction to consensus algorithms, a discussion of their significance, and a comparison of the most prominent algorithms in the area.[11] We cover the Paxos, Raft, and Byzantine Fault Tolerance (**BFT**) ,**PoW**,**PoS**,Delegated PoS algorithms to highlighting their virtues and drawbacks, and investigate prospective developments in consensus algorithm research in the future. Consensus algorithms as represented in the figure below 1 are crucial for preserving the consistency and dependability of distributed systems. In the presence of failures or malevolent actors, they guarantee that all system nodes agree on a single version of the truth. This is especially crucial in decentralized trust systems, such as Blockchain networks, where consensus methods are employed to validate transactions and maintain the integrity of the ledger.

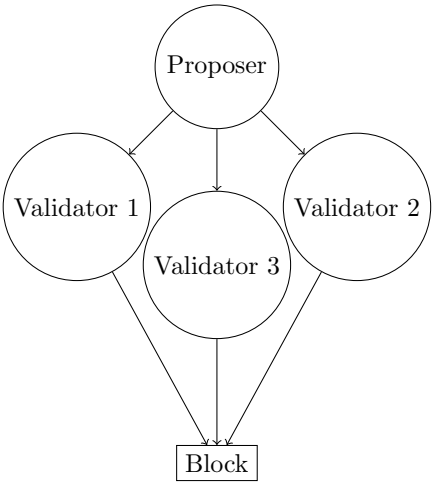

Figure 1: Basics of Consensus Algorithms

2.1 Paxos Algorithm

Leslie Lamport proposed Paxos in 1998, making it one of the earliest and best-known consensus algorithms. It is intended to ensure that a single value is selected by a group of nodes regardless of failures. Paxos is a series of consensus-achieving algorithms for distributed systems. Since its introduction by Leslie Lamport in 1989, it has been a popular approach for distributed consensus. The Paxos method is intended for scenarios in which several participants in a distributed system must agree on a single value. It guarantees that all participants will finally agree on the same value, regardless of network faults or delays. Three phases comprise the algorithm: proposal, acceptance, and commitment.[10] A participant of the network , also known as a proposer, provides a value to be agreed upon during the proposal phase. The proposal can be accepted or rejected by the other participants, known as acceptors. If the suggestion is accepted, the participant notifies all other participants of the decision. If the majority of validators approve the proposal during the acceptance phase, the value is accepted as the agreed-upon value. If not, the proposer resubmits the proposal with a different value, and the procedure is repeated until a value is accepted. At the final phase, the agreed-upon value is locked into the system and cannot be modified.

2.2 Raft Algorithm

Raft[6] is a consensus technique created for logging management in a distributed system. Diego Ongaro and John Ousterhout introduced it in 2014 as an alternative to the Paxos algorithm. The Raft algorithm divides a distributed system into nodes, each of which has a specific function: leader, follower, or candidate. The leader node is responsible for managing the log and replicating it to other nodes, whereas the follower nodes are responsible for duplicating the leader's log and responding to client requests. In the electoral process, the candidate node is a transitional condition. In Raft, the

follower nodes are still operational and duplicating the log, the leader sends heartbeat signals to the follower nodes. The leader begins a new election to choose a new leader if a follower node fails to reply. When there is no current leader or the existing leader is not responding to heartbeat messages, the election process is initiated. A follower node may then become a candidate and solicit votes from other nodes. A candidate becomes the new leader if it obtains votes from a majority of nodes. Raft is supposed to be straightforward and simple to comprehend in comparison to other consensus algorithms such as Paxos. It offers a fault-tolerant technique for managing a replicated log in a distributed system, ensuring that all nodes converge on the same log entries.

2.3 Byzantine Fault Tolerance (BFT) Algorithm

Byzantine Fault Tolerance (**BFT**) shown in the figure 2 is a class of consensus methods in distributed systems that can withstand failures and malicious conduct. Byzantine is a hypothetical situation in which nodes in a distributed system may fail or behave maliciously in random ways, such as delivering contradictory messages or providing false information. **BFT** algorithms are intended to ensure that a distributed system can continue to operate correctly even if some nodes are compromised or fail. Some methods do this by requiring a minimum number of nodes to concur with a decision before it can be considered final. This number is commonly known as the "quorum." Typically, **BFT** algorithms involve a set of nodes, some of which are labelled as "faulty" or "Byzantine" and presumed to behave arbitrarily. In the presence of faulty nodes, the technique ensures that the non-faulty nodes attain consensus on a decision.[8] Miguel Castro and Barbara Liskov published the Practical Byzantine Fault Tolerance (PBFT) algorithm in 1999, which is a prominent **BFT** technique. PBFT divides nodes into replicas and a single primary node, which is responsible for proposing a new block to the network. If a quorum of replicas approves the request, it is committed to the network.

Financial institutions, Blockchain networks, and other decentralized systems frequently employ **BFT** algorithms. Even in the event of errors or malicious conduct, they provide a strong and fault-tolerant approach for attaining consensus.

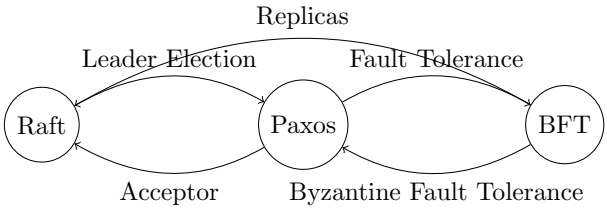

Figure 2: Comparison of Raft, Paxos, and BFT consensus algorithms.

Proof of Work (**PoW**) is a consensus mechanism used by several Blockchain networks to validate transactions and reach consensus. In Proof of Work, participants, also known as miners, compete to use processing power to solve complex mathematical problems. The solution is broadcast to the network by the first participant to solve the problem, and other participants then verify the result. The participant gets awarded with newly minted cryptocurrency and the transaction is added to the Blockchain if the solution is right. **PoW** is intended to be a secure and decentralized consensus process, as it requires substantial computational resources to participate in the network, making it difficult for any individual or organization to gain network control. The network also runs under the idea that the majority of participants are trustworthy and will adhere to the rules, making it difficult for any one or organization to alter the Blockchain's history.[9] **PoW** is extensively employed and has proven to be effective, however it has limitations and difficulties. High computing power is necessary to participate in Proof-of-Work, resulting in substantial energy consumption and carbon emissions. It is also susceptible to 51% assaults, in which a single participant or group gets control of the bulk of the network's computational power and rewrites the Blockchain's history. **PoW** is a popular consensus mechanism, and many Blockchain networks continue to utilize it to achieve consensus and validate transactions despite these obstacles.

2.5 Proof of Stake (PoS)

The Proof-of-stake (**PoS**) algorithm operates via an algorithm that picks participants with the largest stakes to serve as validators, presuming that the participants with the biggest stakes are incentivized to ensure a transaction is executed. The theory is that those with the greatest number of coins in circulation have the most to lose, and are therefore in the best position to act in the network's best interest. As the difficulty of **PoW** varies, so does the amount of coins a network may require. With Proof-of-Stake, blocks are generated not by miners performing work, but by minters "betting" on which blocks are genuine by staking their tokens. Minters use their tokens to vote on which fork to support in the event of a fork. In the event that the majority of voters choose the incorrect fork, validators who choose the incorrect fork would "lose their interest" in the correct one. The most prevalent argument against Proof-of-stake is the Nothing at Stake issue. The risk is that validators could vote for both sides of any fork that occurs, as supporting a fork requires essentially minimal processing effort compared to **PoW**.[4]

2.6 Delegated Proof of Stake (DPoS)

Many Blockchain platforms, including EOS, BitShares, and Steem, utilize the Delegated Proof of Stake (**DPoS**) consensus process. It is a Proof of Stake (**PoS**) consensus process version aimed to address **PoS**'s weaknesses, such as the possibility of centralization and low participation rates. With the **DPoS** consensus mechanism, token holders are responsible for selecting a group of delegates who will validate transactions and create new blocks. The number of delegates may vary depending on the platform,

and they are incentivized to execute transactions expeditiously and effectively. A delegate can be voted out and replaced with a new delegate if they fail to perform their responsibilities.[5]

2.7 Delayed Proof of Work (dPoW)

Delayed Proof of Work (**dPoW**) is a consensus algorithm proposed in 2017 by the Komodo platform. It combines the security of Proof of Work (**PoW**) with the efficiency of Proof of Stake (**PoS**). The **dPoW** consensus algorithm uses a two-tier network to function. The first layer is a Proof-of-Work (**PoW**) network, which validates transactions and creates new blocks. A Proof-of-Stake (**PoS**) network is used to secure the Proof-of-Work (**PoW**) network. In the **dPoW** consensus algorithm, the **PoW** network is utilized to validate transactions and create new blocks. Not immediately after a block is formed is it added to the Blockchain. Instead, it is transferred to the **PoS** network, where a set of notary nodes verify and sign it. Notary nodes are responsible for protecting the **PoW** network by periodically capturing and preserving a snapshot of the Blockchain on the **PoS** network. This snapshot is then used to validate the **PoW** network's integrity. If a block is determined to be invalid, notary nodes can reject it and prevent its addition to the network.

2.8 Proof of Burn (PoB)

Proof of Burn (**PoB**) is a consensus method that enables certain Blockchain networks to reach consensus without requiring computing work or staking. Participants in a **PoB** system must instead "burn" or destroy a set quantity of bitcoin by sending it to an unspendable address, often known as a "burn address."[**empty citation**] Burning cryptocurrency acts as evidence of a participant's commitment to the network, making it an attractive choice for people who do not wish to stake their cryptocurrency or undertake computational work. **PoB** also provides a fair and decentralized method for distributing freshly generated tokens, as tokens are allocated to individuals who have burned cryptocurrency rather than those with processing power or a stake in the network. While **PoB** can serve as an alternative to **PoW** and **PoS**, it is not without its own restrictions and difficulties. For instance, it can be difficult to correctly estimate the quantity of burned cryptocurrency, and there is no assurance that participants will not seek to trick the system. **PoB** offers a novel technique to obtaining consensus in Blockchain networks and has been successfully used in multiple projects.

2.9 Proof of History(PoH)

The Proof of History (**PoH**) consensus mechanism was implemented by the Solana Blockchain in 2018.The **PoH** consensus algorithm runs utilizing a cryptographic clock that generates a verifiable sequence of hashes. This hash sequence is utilized to build an event chronology that may be used to validate transactions and generate new blocks.Each tick of the cryptographic clock generates a new hash; hence, the clock is intended to be highly efficient.This indicates that the clock is capable of generating millions of hashes per second, making it ideal for high-throughput transactions.In the

valid signature for a given hash.As they are compensated for their efforts, validators are encouraged to act in the network's best interest.A validator risks losing their network stake if they engage in destructive behavior.One of the key benefits of the **PoH** consensus algorithm is its high throughput. The platform can process a large number of transactions per second since the cryptographic clock can generate millions of hashes per second.This makes it ideal for applications that demand quick transactions, such as social media and online gaming.Decentralization potential is an additional benefit of the **PoH** consensus algorithm.Because validators are selected based on their ability to provide a valid signature for a given hash, power is dispersed throughout the community rather than concentrated in the hands of a few individuals or institutions.Nonetheless, the **PoH** consensus technique has several limitations.One of its key disadvantages is its dependence on the cryptographic clock.The network's security may be compromised if the cryptographic clock is compromised.In addition, the **PoH** consensus process requires a high level of collaboration between validators, which can be difficult.[3]

2.10 Hedera Hashgraph consensus

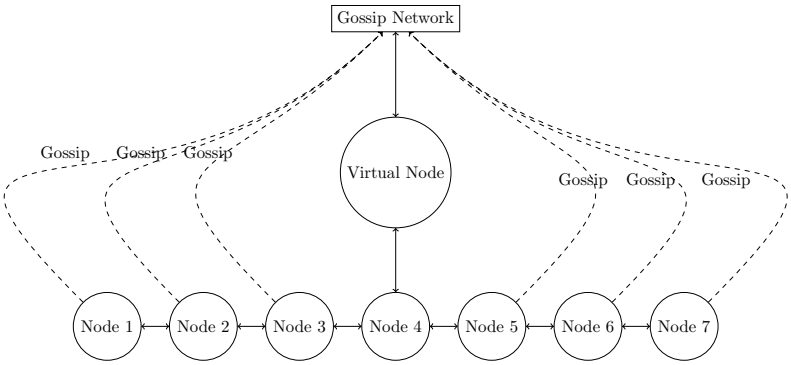

Figure 3: Details of the Hedera Hashgraph Consensus Algorithm

In 2018, **Hedera Hashgraph**, a distributed ledger system, was introduced.Represented in he figure 3 ,It is a novel approach to distributed ledger technology that aims to address the scalability and security limitations of existing Blockchain technology. Hashgraph consensus algorithm is the consensus algorithm utilized by the **Hedera Hashgraph** technology[1]. This process is supposed to be extremely efficient, with each network node talking with every other node to obtain consensus on the ledger's state.

The Hashgraph consensus algorithm shares information between nodes via a gossip mechanism. This protocol assures that all network nodes share identical information, which is needed to obtain consensus on the ledger's state.

In recent years, Blockchain technology has garnered considerable attention because to its potential to transform numerous industries. As with any system, though, Blockchain is not immune to attack[7]. Listed here are some of the most prevalent Blockchain attacks:

- **51% Attack**: A 51% assault occurs when a single entity or group of entities controls more than fifty percent of the Blockchain network's computational power. This enables the attacker to modify the data on the Blockchain, including double-spending and transaction reversal.
- **A Sybil attack**:It happens when an attacker generates numerous bogus identities or nodes on a Blockchain network. This allows the attacker to modify Blockchain data and control a major chunk of the network.
- **DoS Attack**: A DoS attack happens when an attacker floods the Blockchain network with a huge number of transactions or requests, preventing the network from processing valid transactions.
- **Eclipse Attack** :An eclipse attack happens when an attacker isolates a Blockchain node by manipulating the information it gets. This enables the adversary to modify the Blockchain's data and potentially double-spend or reverse transactions.
- **Vulnerabilities in Smart Contracts** : Smart contracts are contracts that execute themselves and are kept on the Blockchain. Yet, they are susceptible to vulnerabilities such as coding mistakes and design faults. These vulnerabilities can be exploited by adversaries to modify Blockchain data.
- **Insider Attacks**: Insider attacks occur when someone with authorized access to the Blockchain network, such as a developer or administrator, intentionally or unintentionally manipulates the data on the Blockchain.

4 Discussion

In this section, we will examine the performance of the different consensus algorithms under three conditions: latency, decentralization, and security. Proof of Work (PoW): Bitcoin and several other cryptocurrencies use **PoW** as a consensus technique. Nodes compete to solve a mathematical challenge to validate transactions and add a new block to the Blockchain using Proof-of-Work (PoW). Due to its rigorous computing requirements, Proof-of-Work provides a high level of security, but it also has a significant level of delay. In addition, **PoW** is extremely centralized because mining pools control a substantial percentage of the network's computational capacity. Proof of Stake (**PoS**): PoS is an alternative to **PoW** that is utilized by networks like Ethereum. In Proof-of-Stake, nodes are selected to validate transactions based on the quantity of cryptocurrency they hold, as opposed to their processing capacity. **PoS** has shorter latency than **PoW** since nodes are not required to solve hard puzzles, but it has weaker security because nodes with large quantities of cryptocurrency have a greater influence on the network. **PoS** is also somewhat centralized, as nodes with greater wealth have greater influence on the network. Delayed Proof of Work (dPoW): Komodo and

struct blocks using **PoW**, but these blocks are not deemed genuine until they have been certified by another network using **PoW**. **dPoW** has greater latency than **PoW** or **PoS** because it requires notarization, but it also has greater security because the notarization network provides extra security. In addition, **dPoW** is extremely decentralized because nodes from both networks can participate in the consensus method. Delegated Proof of Stake (DPoS): Networks such as EOS and BitShares utilize **DPoS** as a consensus method. Nodes are selected to validate transactions in **DPoS** based on the number of votes they receive from other nodes. As nodes are chosen to validate transactions without having to solve riddles or keep huge quantities of bitcoin, **DPoS** has low latency. **DPoS** is also more secure than **PoS** because malicious nodes are subject to vote and can be deleted. Nonetheless, **DPoS** is still rather centralized because nodes with more votes have greater network influence. Proof of Burn (PoB): Slimcoin and Counterparty utilize **PoB** as a consensus mechanism. To demonstrate their commitment to the network, nodes in Proof-of-Burn burn a fixed quantity of cryptocurrency, rendering it unrecoverable. **PoB** has low latency since nodes do not need to solve puzzles or store huge quantities of money, but it has worse security because nodes can execute a 51% assault if they gather enough burned coins. Moreover, **PoB** is relatively concentrated, as nodes with more burned coins have a greater impact on the network. **Hedera Hashgraph**: **PoB** is a consensus mechanism that achieves consensus using a DAG and the gossip protocol. Due to its rapid and fair consensus method, **PoB** has low latency, and its security is excellent, as it is almost impossible to alter the transaction history without the network being aware. **PoB** is likewise extremely decentralized, as nodes are elected by council and no single node controls the network. We can in the table 1 a full summary of the comparaison of consensus algorithms that we had talked about.

Table 1: Comparison of Consensus Algorithms

Algorithm	Energy Efficiency	Security	Scalability	Decentralization	Finality
Proof of Work	Low	High	Low	High	Probabilistic
Proof of Stake	High	High	High	Medium	Probabilistic
Proof of History	High	High	High	High	Probabilistic
Proof of Burn	High	High	Low	High	Probabilistic
Delegated Proof of Stake	High	High	High	Medium	Deterministic
Delayed Proof of Work	High	High	High	High	Deterministic
Hedera Hashgraph	High	High	High	High	Probabilistic

In conclusion, each consensus mechanism possesses a distinct set of advantages and disadvantages pertaining to latency, security, and decentralization. **PoW** offers excellent security, but at the expense of high latency and centralization. **PoS** offers low latency, but is less secure and centrally located. **dPoW** and **DPoS** offer greater security than **PoS**, but have greater latency. **PoB** has a short latency, however it is less secure and less centralized than alternative approaches.

PoB, on the other hand, is a potential alternative that achieves low latency, high security, and great decentralization, making it an attractive choice for numerous applications. Its usage of a DAG and the gossip protocol provides rapid and fair consensus as well as the immutability of transaction history. In addition, its council election mechanism assures that no single node controls the network, hence fostering a high degree of decentralization.

Many considerations, including as the use case, the network's requirements, and the desired level of security, latency, and decentralization, influence the choice of consensus mechanism for a Blockchain network. By analyzing the benefits and drawbacks of each consensus method, developers can select the optimal solution for their network. **PoB** achieves a balance of high security, low latency, and great decentralization, making it an attractive solution for many Blockchain applications.

References

[1] Mohammad Alahmad et al. "INFLUENCE OF HEDERA HASHGRAPH OVER BLOCKCHAIN". en. In: 17 (2022).

[2] Shikah J. Alsunaidi and Fahd A. Alhaidari. "A Survey of Consensus Algorithms for Blockchain Technology". en. In: *2019 International Conference on Computer and Information Sciences (ICCIS)*. Sakaka, Saudi Arabia: IEEE, Apr. 2019, pp. 1–6. ISBN: 978-1-5386-8125-1. DOI: 10.1109/ICCISci.2019.8716424. URL: https://ieeexplore.ieee.org/document/8716424/ (visited on 03/01/2023).

[3] Lin Chen et al. "On Security Analysis of Proof-of-Elapsed-Time (PoET)". en. In: *Stabilization, Safety, and Security of Distributed Systems*. Ed. by Paul Spirakis and Philippas Tsigas. Vol. 10616. Series Title: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2017, pp. 282–297. ISBN: 978-3-319-69083-4 978-3-319-69084-1. DOI: 10.1007/978-3-319-69084-1_19. URL: http://link.springer.com/10.1007/978-3-319-69084-1_19 (visited on 03/30/2023).

[4] Xiang Fu, Huaimin Wang, and Peichang Shi. "A survey of Blockchain consensus algorithms: mechanism, design and applications". en. In: *Science China Information Sciences* 64.2 (Feb. 2021), p. 121101. ISSN: 1674-733X, 1869-1919. DOI: 10.1007/s11432-019-2790-1. URL: https://link.springer.com/10.1007/s11432-019-2790-1 (visited on 04/08/2022).

[5] Qian Hu et al. "An Improved Delegated Proof of Stake Consensus Algorithm". en. In: *Procedia Computer Science* 187 (2021), pp. 341–346. ISSN: 18770509. DOI: 10.1016/j.procs.2021.04.109. URL: https://linkinghub.elsevier.com/retrieve/pii/S1877050921009133 (visited on 03/30/2023).

Raft Consensus Algorithm for Private Blockchains". en. In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 50.1 (Jan. 2020), pp. 172–181. ISSN: 2168-2216, 2168-2232. DOI: 10.1109/TSMC.2019.2895471. URL: https://ieeexplore.ieee.org/document/8666147/ (visited on 03/30/2023).

[7] Xiaoqi Li et al. "A survey on the security of blockchain systems". en. In: *Future Generation Computer Systems* 107 (June 2020), pp. 841–853. ISSN: 0167739X. DOI: 10.1016/j.future.2017.08.020. URL: https://linkinghub.elsevier.com/retrieve/pii/S0167739X17318332 (visited on 03/30/2023).

[8] Henrique Moniz. *The Istanbul BFT Consensus Algorithm*. en. arXiv:2002.03613 [cs]. May 2020. URL: http://arxiv.org/abs/2002.03613 (visited on 03/01/2023).

[9] Giang-Truong Nguyen and Kyungbaek Kim. "A Survey about Consensus Algorithms Used in Blockchain". en. In: *Journal of Information Processing Systems* 14.1 (Feb. 2018), pp. 101–128. DOI: 10.3745/JIPS.01.0024. URL: https://doi.org/10.3745/JIPS.01.0024 (visited on 04/07/2022).

[10] Roberto De Prisco, Butler Lampson, and Nancy Lynch. "Revisiting the Paxos algorithm". en. In: ().

[11] Mehrdad Salimitari and Mainak Chatterjee. *A Survey on Consensus Protocols in Blockchain for IoT Networks*. en. arXiv:1809.05613 [cs]. June 2019. URL: http://arxiv.org/abs/1809.05613 (visited on 03/01/2023).