electronics

MDPI

Review

Blockchain Consensus Mechanisms: A Comprehensive Review and Performance Analysis Framework

Zhihua Shen, Qiang Qu and Xue-Bo Chen *

School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan 114051, China; szh949950573@163.com (Z.S.); quqiang@ustl.edu.cn (Q.Q.)
* Correspondence: xuebochen@126.com

Abstract

In recent years, blockchain consensus mechanisms have evolved significantly from the original proof-of-work design, transitioning towards more efficient and scalable alternatives. This paper presents a comprehensive review and analysis framework for blockchain consensus mechanisms based on a systematic examination of 200+ publications. We categorize consensus mechanisms into four performance-oriented groups: high throughput, strong security, low energy, and flexible scaling, each addressing specific trade-offs in the blockchain trilemma of decentralization, security, and scalability. Through quantitative metrics including transactions per second, energy consumption, fault tolerance, and communication complexity, we evaluate mainstream mechanisms. Our findings reveal that no single consensus mechanism optimally satisfies all performance requirements, with each design involving explicit trade-offs. This paper provides researchers and practitioners with a structured framework for understanding these trade-offs and selecting appropriate consensus mechanisms for specific application contexts. Finally, we discussed future development trends, as well as regulatory and ethical considerations.

Keywords: blockchain; consensus mechanism; performance analysis

Academic Editor: Fabio Grandi

Received: 5 August 2025
Revised: 29 August 2025
Accepted: 3 September 2025
Published: 8 September 2025

Citation: Shen, Z.; Qu, Q.; Chen, X.-B. Blockchain Consensus Mechanisms: A Comprehensive Review and Performance Analysis Framework. *Electronics* **2025**, *14*, 3567. https://doi.org/10.3390/electronics14173567

1. Introduction

The advent of blockchain technology, as exemplified by Bitcoin in 2008, has profoundly transformed the landscape of data storage and transaction verification. The seminal work by Bailey W. Diffie and Martin E. Hellman in 1976, entitled New Direction in Cryptography [1], laid the foundational principles of cryptography that have since been instrumental in the development of blockchain technology. In 1982, Lamport proposed the General Byzantine Problem [2], which provided the basis for the theory and practice of reliability in distributed computing. In the 1990s, W. Scott Stornetta and Stuart Haber proposed the ideas of timestamps and Merkle trees [3]. The genesis of blockchain technology can be traced back to the conception of Bitcoin [4], as it was initially proposed by Satoshi Nakamoto in 2008 as the underlying technology for Bitcoin. This was followed by the establishment of the Bitcoin Network in 2009, which led to the creation of the inaugural block, known as the "Genesis Block" [5]. Since the inception of the Bitcoin network, blockchain has evolved into a global technology that has attracted global attention and investment. It has been argued that it has solved the world's most difficult problem, the Byzantine Generals' Problem. The blockchain is defined as a peer-to-peer network system that uses cryptography [6,7] and consensus mechanisms to build and store large chains of transaction data. The blockchain shared value system was first emulated by numerous cryptocurrencies,

with improvements in consensus mechanisms. Perhaps the best known and most widely used blockchain consensus mechanism is proof of work (PoW) [4]. PoW is used by Bitcoin and Ether [8] (approximately 50% of energy and blockchain projects are developed on Ether [9]). However, PoW is also known for expending significant time and effort on the resolution of a complex mathematical puzzle that ultimately holds little significance. To illustrate this point, consider the Cambridge Bitcoin Electricity Consumption Index (CBECI). This index demonstrates that the Bitcoin network consumed approximately 107.65 TWh of electricity in 2022. However, this figure has increased to over 150 TWh in 2024, which is equivalent to the annual electricity consumption of some medium-sized countries. This elevated energy consumption has been demonstrated to engender increased operating costs while concomitantly giving rise to environmental concerns. Consequently, researchers have been compelled to explore more energy-efficient alternatives.

The CCAF website shows that the power consumption of Bitcoin mining is increasing, which is a waste of power resources. As the number of Bitcoins decreases, the reward for each outgoing block decreases, meaning that the cost of power consumed for each new Bitcoin mined will increase. It is evident that the emergence of criticism directed towards PoW has given rise to the development of alternative consensus mechanisms, including proof of stake, proof of capacity, proof of authority, and proof of burn. In addition to this, by 2025, blockchain-based carbon credit trading systems will utilise proof of authority (PoA) for efficient private chain deployment. Cross-chain technologies will improve interoperability between networks, and research on quantum-resistant algorithms will address the potential threat of quantum computing. Blockchain technology has made significant progress in the direction of cross-chain interoperability, quantum resistance, and carbon neutrality. Proof of stake (PoS) and its variants (e.g., delegated proof of stake (DPoS)) are gradually replacing PoW as the mainstream. Consensus mechanisms such as the Delayed Acceptance Gap (DAG), Hot Stuff, and so on are beginning to be explored and applied to financial and Internet of Things (IoT) combined applications.

As demonstrated in Figure 1, there has been a clear trend of an increase in the number of blockchain-related papers published over the past decade, with a notable rise in the circulation of articles related to the consensus mechanism, reaching an average of 815 articles per year. It is evident that blockchain technology possesses promising application prospects.

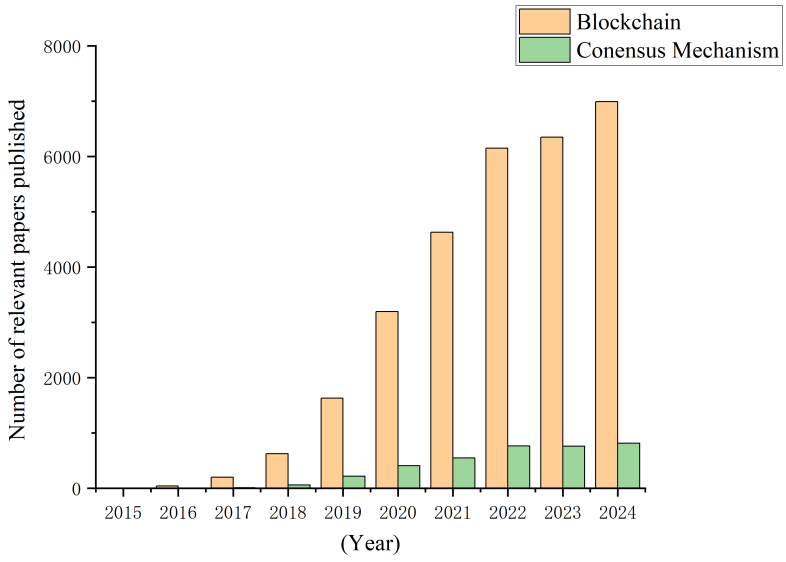

Figure 1. Number of relevant papers published.

However, there is a dearth of literature from a macro perspective on the research outcomes in this domain. The primary objective of this paper is to address this lacuna by undertaking a comprehensive and systematic review. The paper's objective is threefold: firstly, to systematically compare the mainstream consensus mechanisms' performance; secondly, to identify and weigh the most applicable and effective criteria as part of an analytical framework to evaluate the strengths and weaknesses of consensus mechanisms; and thirdly, to analyse its performance and combine it with the latest research to propose optimisation directions. Specific data will be available from the White Paper on the consensus mechanism, the official website, and subsequent research by academics. This will enable us to discern the possible future development trends.

In order to guarantee the comprehensiveness and reliability of the literature sources, a systematic review of articles related to consensus mechanisms was conducted. The PRISMA methodology was used, as illustrated in Figure 2. This approach serves to enhance the transparency and reproducibility of the review process.

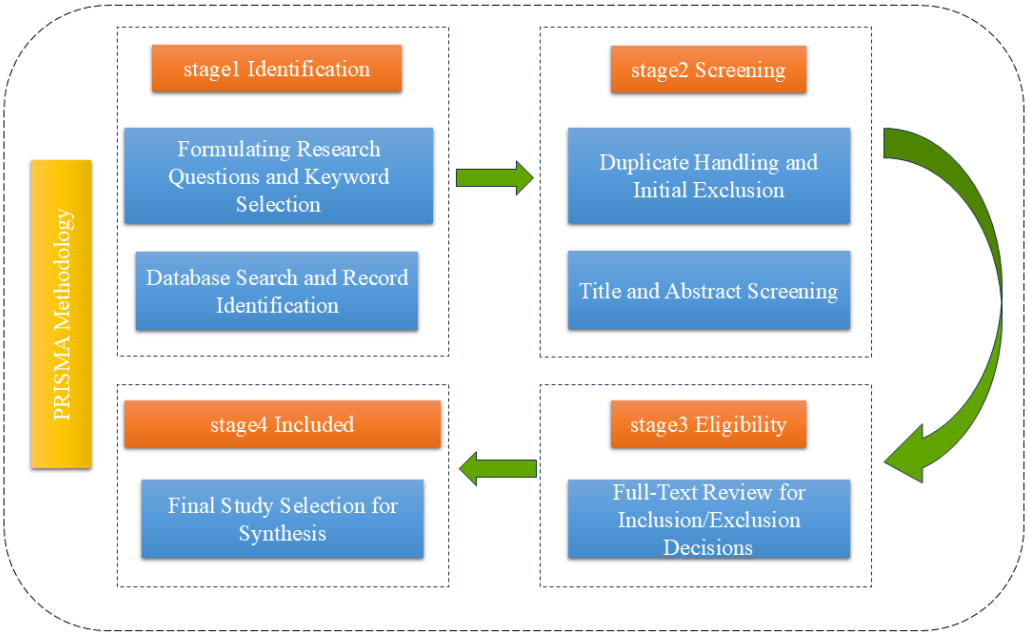

Figure 2. Applying the PRISMA methodology to consensus mechanism research.

Stage 1: The initial research question was clearly defined in order to focus on the performance characteristics of key consensus mechanisms in blockchain systems, specifically PoW, PoS, DPoS, practical Byzantine fault tolerance (PBFT), and so on. This overarching question served as the primary impetus for the entire review process. A comprehensive list of keywords was developed for retrieval, including "blockchain", "consensus mechanism", "throughput", "latency", "scalability", "security", and "decentralization" along with their synonyms and related terms. Subsequently, systematic searches were performed across multiple academic databases, including IEEE xplore, ACM digital library, ArXiv, Elsevier, Springer, and Web of Science. Search syntax was applied with the objective of optimising results.

Stage 2: Screening, at the initial step, entailed the management of duplicates, which entailed the removal of all redundant entries identified across various databases. This approach was adopted to guarantee that each unique article was considered only once. Subsequently, the titles and abstracts of the remaining records were systematically screened against predefined inclusion and exclusion criteria. The screening process was meticulously designed to prioritise articles that were deemed to be pertinent to blockchain consensus mechanisms, empirical studies, and academic rigour. Articles that were considered to

be clearly irrelevant, such as non-research articles, or those falling outside the scope of consensus mechanisms, were explicitly excluded from the analysis.

Stage 3: Eligibility involved a full-text review of the remaining articles to determine their final inclusion or exclusion based on the established criteria.

In the final stage, designated as "Included Studies", the research that had passed the full-text eligibility assessment was incorporated into the systematic review.

During the literature screening process, in which multiple researchers participated, consistency was ensured through the initial testing of screening criteria and the subsequent discussion of any discrepancies. Despite the authors' best efforts to conduct a comprehensive review, the study is subject to several limitations. These include the potential for language bias in the primary consideration literature, a specific time frame, and the exclusion of the grey literature not included in mainstream databases.

To provide a comprehensive analysis of consensus mechanisms, we investigated a wide range of resources from academic journals, industrial websites/blogs, conferences, and workshops, to white papers. A total of 270 publications have been selected for consensus analysis. According to statistics, there are three leading publication sources: academic journals (51.5%) that represent the data gathered from the academic domain, various prioritized improvements to the consensus mechanism published at the Conference/Symposium (17.4%), and industrial websites (10.7%) that denote websites and blogs from the blockchain industry.

The primary contributions of this text are as follows:

- This paper sets out to examine the current status of the development of a consensus mechanism for the distributed ledger technology, and to identify the subjects that are currently being studied with the greatest intensity.
- The present study elucidates the underlying design principles and internal trade-offs of disparate consensus mechanisms from the perspective of their functionality.
- The objective is to identify the key performance indicators (KPIs) that are of paramount importance, including the throughput, security, energy consumption, and expandability.
- The establishment of a multi-dimensional analytical framework was undertaken for the purpose of evaluating the merits and demerits of the prevailing consensus mechanism.
- This paper provides a prospective analysis of the future development and regulator and ethical considerations of the blockchain consensus mechanism.

The structure of the paper is as follows: Section 2 reviews the foundation of blockchain technology and analyses the important position of the consensus mechanism in blockchain technology; Sections 3–6 compare the performance of the consensus mechanism in detail according to four categories: high throughput, high security, low energy consumption and, flexible expansion; and finally, the challenges that the consensus mechanism in blockchain technology is still facing are analysed; Section 7 discusses regulatory issues, ethical considerations, and future directions; and Section 8 provides the conclusion for this paper.

2. Overview of Blockchain Technology

2.1. Basic Components of Blockchain

The blockchain is a distributed ledger technology that functions on a peer-to-peer network. It guarantees the immutability of data through cryptography and consensus mechanisms. The following components constitute the core of the system:

- **P2P network:** enables direct communication between nodes.
- **Distributed ledger:** records the history of all transactions.

- **Cryptography:** such as elliptic curve signatures and hash functions to ensure security.
- **Consensus mechanisms:** coordinate node consistency.
- **Smart contracts:** automate the execution of transaction logic, as shown in Figure 3.

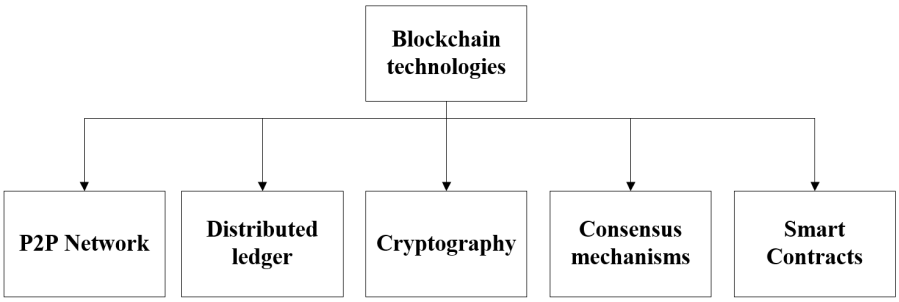

Figure 3. Blockchain technology component diagram.

According to Figure 4, the composition of blockchain can also be described in layers. This layered model is analogous to the layered architecture of computer networks. The division of blockchain into distinct layers facilitates a more profound comprehension of its technical architecture and functional capabilities [10]. The following are the common compositional layers of blockchain, from high to low: the application layer, contract layer, service layer, protocol layer, and network layer, as well as the underlying physical layer.

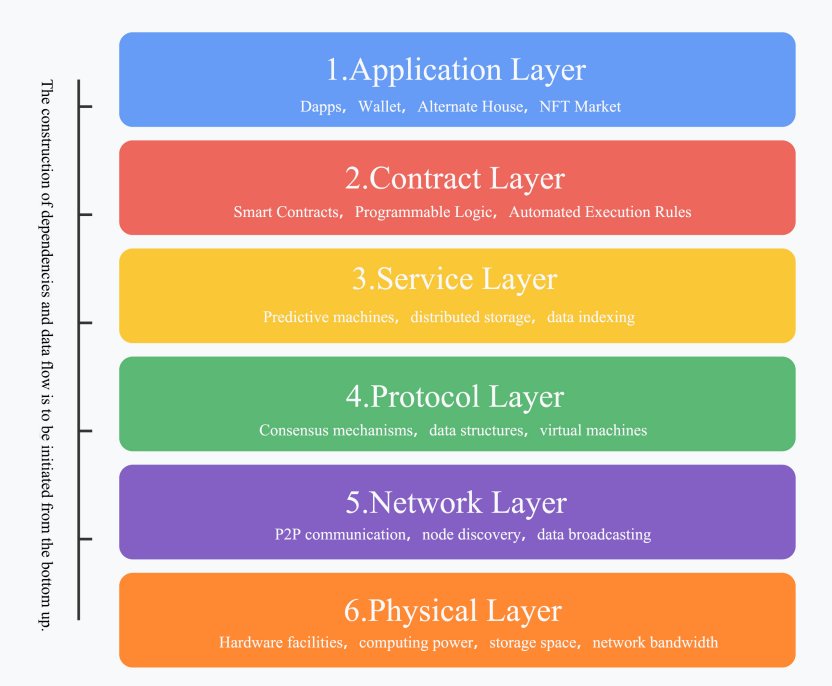

Figure 4. Blockchain layered architecture diagram.

Regardless of the perspective adopted, consensus mechanisms are widely recognised as playing a pivotal role in ensuring the consistency of protocols within the blockchain paradigm.

2.2. Application Areas of Blockchain

The blockchain relies on consensus mechanisms to ensure that network nodes agree without a central authority. The advent of blockchain technology has engendered a plethora of novel financial instruments, including digital currencies, e-commerce, global payments, remittances, peer-to-peer lending, and smart contracts. These innovations have profound

implications for various domains, such as digital entitlements, gaming, and escrow payments. Moreover, the immutability of blockchain technology has led to its applications in domains such as healthcare data storage, title registration, property ownership verification, voting systems, and intellectual property protection.

As of 2025, the expansion of blockchain applications has not been limited to cryptocurrencies; significant growth has been observed in areas such as the Internet of Things (IoT), fintech, and energy trading. The reduction of intermediary costs and the acceleration of cross-border payments (for example, Ripple's XRP) are two such factors. Decentralised finance, as exemplified by Uniswap and Aave, is another. Finally, asset tokenisation, whereby physical assets (for example, real estate and artwork) are converted into digital tokens for the purpose of facilitating trading and division, is also worthy of note. It is evident that the application of blockchain technology has permeated various sectors, including but not limited to supply chains, logistics, healthcare, digital identity, the Internet of Things (IoT), copyright and content management, and energy management. In essence, blockchain has become an inextricable part of our daily lives, seamlessly integrating into all aspects of our existence. The consensus mechanism constitutes a pivotal component of the blockchain, thereby rendering it a subject of considerable research significance. In the context of diverse application scenarios, the selection of a particular consensus mechanism becomes imperative. It is noteworthy that the quest to identify a consensus mechanism that can fulfil all performance requirements remains an enduring pursuit for all relevant researchers.

2.3. Analysis of Consensus Mechanisms

In the context of blockchain technology, data is stored in a chain-like structure, thereby achieving immutability. In contrast to centralized architectures, each participating node in a blockchain has an equal right to record data. In order to ensure the veracity of the data, it is necessary to establish a consensus mechanism that will guarantee all nodes agree on the data and prevent the submission of fraudulent data by malicious nodes. In simple terms, consensus is the agreement reached by all parties through negotiation, and within the context of a centralized architecture, there exists an authority to whom all other parties must adhere. However, given that the blockchain is a decentralised mechanism, a fundamental challenge lies in ensuring the consistency of data across nodes. To address this issue, the development of consensus algorithms has been proposed as a solution. These algorithms aim to facilitate the consistency and correctness of book data across various book nodes.

The consensus process is centred on two fundamental mechanisms: "leader election" and "bookkeeping". Each round of the process is subdivided into four stages: leader election, block generation, data validation, and chain update. In the operation process under consideration, each round is comprised of four distinct stages: leader election, block generation, data validation, and chain update. The inputs to the consensus process are the transactions or data generated and validated by the data nodes, and the outputs are the encapsulated data blocks and the updated blockchain. As illustrated in Figure 5, the four phases are executed in a cyclical manner, with the generation of a new block occurring for each round of execution.

The decentralised nature of blockchain precludes the existence of a central bookkeeping node; rather, a network-wide consensus on the ledger is required. Consequently, the consensus mechanism, as one of the key technologies of blockchain, plays an important role in business throughput, transaction speed, tamperability, access threshold, and so on.

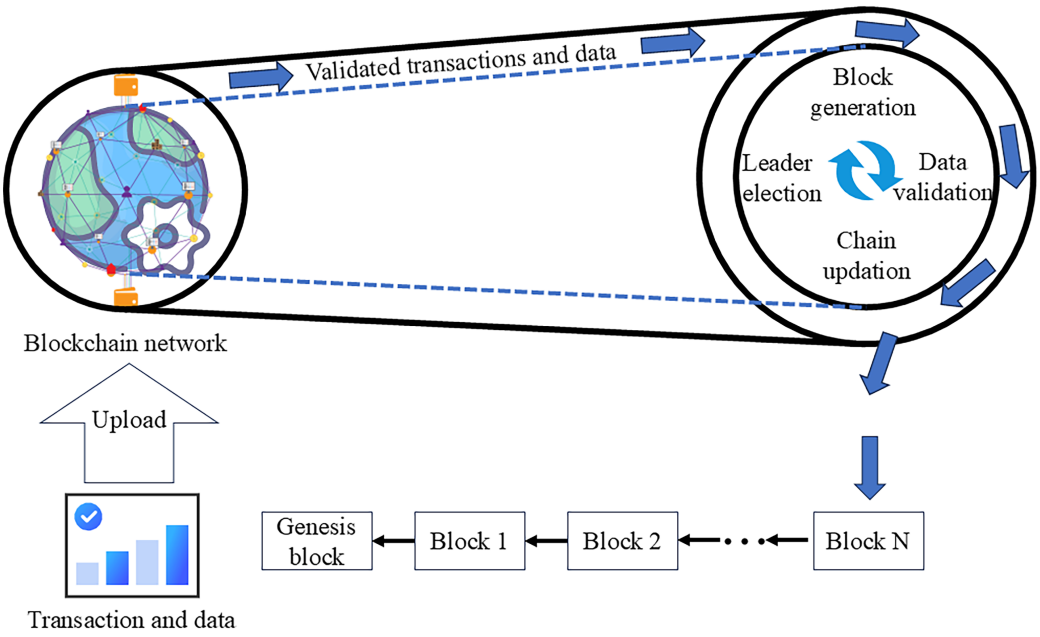

Figure 5. Consensus mechanism in the blockchain.

Hazari's article provides a visual representation of the 2018 statistics for the top 50 cryptocurrencies according to their current market capitalisation [11], illustrated in terms of the consensus mechanisms employed. CoinMarket statistics reveal the statistics pertaining to the consensus mechanisms employed by the top 50 cryptocurrencies in terms of current market capitalisation.

As demonstrated in Table 1, while the predominant portion of market value is still held by PoW, significant alterations have been observed in the other components of the consensus mechanism. These changes signify that as society and technology evolve, the prevailing consensus mechanism may no longer satisfy the demands of emerging technologies. Consequently, a novel mainstream consensus mechanism will emerge, superseding the existing mainstream consensus mechanism that selects the best. Consequently, contemporary mainstream consensus mechanisms are currently being screened and reclassified.

Table 1. Evolution of consensus mechanisms in top 50 cryptocurrencies.

Consensus Type	2018 Market Share	2025 Market Share	Key Changes
PoW	65.6%	48.3%	Bitcoin dominance decreased
PoS	11.8%	31.5%	Ethereum transition, new PoS chains
DPoS	14%	9.3%	Consolidation around major platforms
BFT variants	2%	6.8%	Enterprise adoption growth
Others	6.6%	4.1%	Experimental mechanisms declined

The present study draws upon the comprehensive evaluation framework proposed by Bamakan [12] and a systematic search of journals, conference papers, books, and articles. It was noted that the criteria and metrics used by researchers to compare consensus algorithms are limited to throughput, number of transactions per second, block time/latency, block validation time, block size, energy consumption, double-payment attacks, 51% attacks, degree of decentralisation, and so on [6,10,11,13–18]. In the context of reclassifying the consensus mechanisms, a classification method is proposed that is based on performance objectives. The method is founded upon a thorough examination of the proposed meaning and realised performance of the consensus mechanism, which is itself categorised into four distinct classifications. Following a comprehensive review of the extant literature and a

thorough online search, the following specific categories are proposed: high throughput oriented, high security oriented, low energy consumption oriented, and flexible scaling oriented. In order to facilitate the clear categorization and comparison of advantages and disadvantages, it is necessary to develop clear definitions and quantitative criteria for each category. For this purpose, reference is made to Table 2. As illustrated in Table 2, a comprehensive list of quantitative criteria is provided for the four categories of categorization. The quantitative criteria were identified and weighted following a systematic analysis of the existing literature and expert consensus. The intention is that they provide a comprehensive and actionable framework for assessment.

Table 2. Quantitative criteria for the classification of consensus mechanisms.

Categories	Central Goal	Major Indicators	Quantitative Standards	Typical Application Scenarios
High throughput [10,11,13]	Maximize trading speed and capacity	TPS, block finality time	TPS $\geq$ 1000, 5 s $\geq$ confirmed time	Payment systems, DEX, NFT markets
Strong security [10,14,15]	Defend against attacks and ensure consistency	Fault tolerance, attack costs	Tolerates 33% of malicious nodes, costly to attack	Financial, healthcare, government data
Low energy [6,16,17]	Reduced energy and hardware consumption	Energy per transaction, hardware requirement	Energy consumption $\geq$ 0.001 kWh, No specialized hardware required	IoT, green blockchain, low-cost networks
Flexible extension [6,18]	Adapting to changes in network size	Node scalability, communication complexity	Supports 500+ nodes with low communication complexity	Cross-chain, enterprise chain, dynamic network

TPS, Transactions Per Second: measures the number of transactions processed per unit of time. **Fault tolerance:** the percentage of malicious nodes that can be tolerated. **Attack cost:** the resources (e.g., arithmetic, tokens) required to launch an attack. **Energy per transaction:** electricity consumption per unit of trade, usually measured in kilowatt hours (kWh). **Hardware requirement:** the computing, storage, or bandwidth resources required to run the node. **Node scalability:** the maximum number of nodes supported. **Communication complexity:** the amount of message exchanges required between nodes to reach consensus.

In addition to the quantitative indicators referenced above, the performance of consensus mechanisms is also influenced by qualitative factors, including governance models, community support and ecosystem activity, and the capacity to upgrade protocols. It is challenging to quantify these factors directly; however, they play a decisive role in the long-term development, practical utility, and market adoption of consensus mechanisms.

3. High Throughput

High throughput-oriented consensus mechanisms are designed to optimise transaction processing speed and network capacity, thereby supporting substantial, high-frequency transactions or data operations. Such mechanisms typically achieve efficient performance by reducing the number of consensus participants, optimising communication protocols, or introducing parallel processing.

3.1. Multi-Paxos

Multi-Paxos introduces a stable leader on top of the basic Paxos system with the purpose of improving performance by reducing the number of communication rounds required for each consensus. This phenomenon is illustrated in Figure 6. The node in question is responsible for the coordination of all proposal submissions. In the presence of a stable leader, the Prepare phase of the basic Paxos is omitted, with the system progressing directly to the Propose phase and the Accept phase [19].

The leader can propose a series of values (e.g., log entries) consecutively without having to run the full Paxos process for each value [20]. Multi-Paxos extends Paxos from a single consensus to a consecutive consensus, reduces the number of communication rounds

(usually only one round of message exchange is required), significantly reduces latency and overhead, and is well suited to the common log or state machine replication requirements in distributed systems. The system has been shown to exhibit strong consistency and fault tolerance, as evidenced by its ability to tolerate a few node failures. This property renders it well suited to scenarios in distributed systems that require efficient handling of multiple ordered operations [21,22].

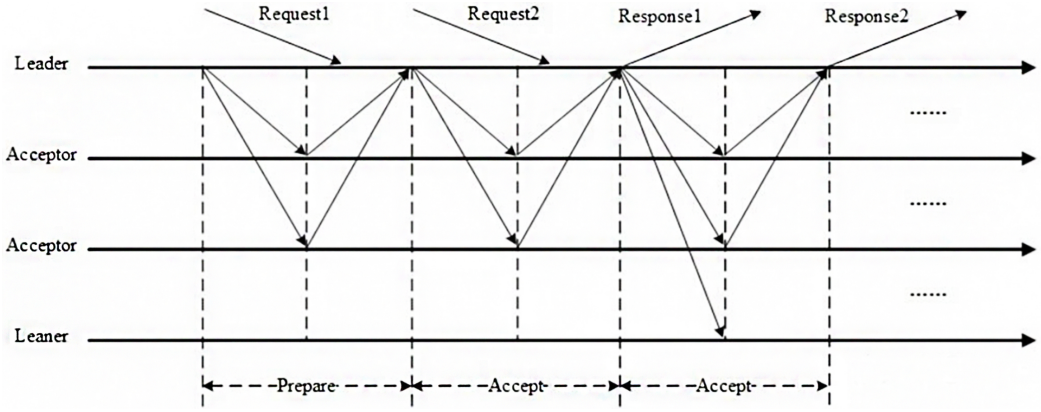

Figure 6. The operational workflow of the Multi-Paxos consensus mechanism.

However, Multi-Paxos performance is found to be contingent on leader stability and network conditions. In the event of a failure of leadership, the system must re-elect the leader, which may result in brief service interruptions or delays. Despite its relative simplicity in comparison to basic Paxos, the system still involves role partitioning and fault recovery, both of which can be difficult to understand and implement. Furthermore, the leader must address all suggestions, which have the potential to hinder system performance under high load conditions, thereby creating a potential single-point bottleneck. Multi-Paxos enhances the efficiency of Paxos by introducing a leader and simplifying the process, rendering it well suited for distributed systems that require high throughput and strong consistency. Nevertheless, the issue of its reliance on leaders, in conjunction with the inherent intricacy of the system, remains a salient concern that must be given due consideration.

3.2. Delegated Proof of Stake (DPoS)

The DPoS mechanism is predicated on PoS and incorporates the notion of eligibility elections. Elections have been shown to enhance the efficacy and security of the mechanism by electing representative nodes [23]. DPoS systems generally possess a reduced number of decentralised nodes and are thus subject to fewer security challenges; however, they may also result in a concentration of power [24]. After reviewing the relevant literature, Table 3 is summarized, which summarizes the performance metrics of the DPoS variants.

Table 3. DPoS variant performance indicators.

	Throughput	Security	Centralization	Scalability	Energy Consumption
DPoS [25]	High	Moderate	Partially Centralized	High	Very Low
HL-DPoS [26]	Very High	Moderate	Centralized	Very High	Very Low
DL-DPoS [27]	Very High	Moderate	Partially Centralized	Very High	Very Low
PDPoS [28]	Very High	Moderate	Centralized	Very High	Very Low
Roll-DPoS [29]	Very High	High	Decentralized	High	Very Low
DT-DPoS [30]	High	Moderate	Partially Centralized	High	Very Low
SP-DEWOA [31]	Very High	Very High	Partially Centralized	Very High	Low
RP-DPoS [32]	High	High	Partially Centralized	High	Very Low

The Hierarchical Layered DPoS (HL-DPoS) optimises the verification process by introducing a hierarchical structure. The layering of consensus nodes has been demonstrated to reduce the direct communication burden and improve TPS (transactions per second). While the layered design may increase complexity, the inter-layer supervision mechanism is perfect, thus improving the resistance to attacks. Furthermore, layers have been shown to lead to the concentration of power at the higher level, and layering helps to support more nodes and transactions, which has excellent scalability. Finally, the low-energy consumption characteristics are retained [26].

DL-DPoS (Dynamic Layered DPoS) is a dynamic system that adjusts tiers or node roles in order to adapt to the network load. It has been demonstrated that dynamically allocating resources can significantly improve TPS, especially at high loads. However, dynamism can also introduce uncertainty, and it is therefore vital to maintain a high level of security through real-time monitoring. Furthermore, although dynamic adjustment makes the distribution of power more flexible, it is still limited to a small number of nodes. Despite these limitations, DL-DPoS has been shown to have excellent scalability and is suitable for large-scale networks. Finally, it has been demonstrated to have low energy consumption [27].

PDPoS (Preferential Delegated Proof of Stake) has been demonstrated to enhance efficiency and TPS through the implementation of a two-tier structure, comprising super-representatives and regular representatives. The efficacy of TPS is notably augmented by this approach, which mitigates the necessity for active verifiers. The security of the system is contingent upon the calibre of super-representatives, with the election process playing a pivotal role in determining its overall strength. The two-tier design has been hypothesised to centralise power in a small number of super-nodes, thereby weakening the degree of decentralisation. However, the two-tier structure has been shown to optimise the consensus process and exhibits excellent scalability. It has been demonstrated that energy efficiency is lower than in DPoS because there are fewer authentication nodes [28].

Roll-DPoS (Randomized Delegated Proof of Stake) is a system that introduces randomness with a view to enhancing decentralisation and adapting to complex scenarios, such as the Internet of Things (IoT). The random selection of the verifier may result in a slight reduction in the TPS, but it remains higher than that of PoW/PoS. The introduction of randomness has the effect of reducing the possibility of collusion and improving resistance to attacks. Furthermore, random election enhances power decentralisation, which is preferable to standard DPoS. Roll-DPoS is suitable for large-scale distributed systems (e.g., IoT) with higher scalability and low energy consumption as DPoS [29].

DT-DPoS (Dynamic Threshold DPoS) optimises performance by dynamically adjusting consensus thresholds. It has been demonstrated that dynamic thresholds improve TPS at high loads. Furthermore, the security of the system depends on threshold adjustments, with the degree of decentralisation depending on the number of nodes. Dynamism makes the system more adaptable to network changes and it scales better. Finally, low energy consumption remains largely unchanged [30].

It is hypothesised that SP-DEWOA (Spark-based Parallel Differential Evolution and Whale Optimization Algorithm) may result in a slight reduction in real-time throughput, given that its election process is optimised by iterative optimisation rather than simple statistical voting. The enhancement of the role of voting polarity is achieved by the introduction of Kendall coefficients. This, in turn, enhances the voting polarity and increases the difficulty of manipulation. The randomness of the heuristic algorithm makes the manipulation results unpredictable, thereby significantly improving security. The degree of decentralisation is similar to that of the standard DPoS, and the number of nodes has not been significantly changed (it remains the top-k). Lint has demonstrated, through

experimental investigation, that it exhibits exceptional scalability [31]. While maintaining low-energy characteristics, it introduces additional computational overheads, such as iterative optimisation of DE and WOA, and Spark parallel computation.

RP-DPoS (Reputation-based DPoS) introduces a reputation model to screen high-quality nodes, thus improving efficiency and security. Reputation screening reduces ineffective nodes, and TPS may be higher than the standard DPoS. The reputation mechanism reduces the probability of the election of malicious nodes, and significantly improves security. However, it still relies on a small number of nodes, and the degree of decentralisation has not been significantly improved. The reputation system supports more node participation, and better scalability. It also has low energy consumption, consistent with DPoS [32].

Furthermore, an enhanced DPoS mechanism founded on Vague set has been developed, which renders the selection of proxy nodes more consistent with human voting behaviour, thereby enhancing the security and fairness of blockchain [33]. Chen and Liu have proposed a functional framework for a collaborative autonomy mechanism of network opinion based on blockchain, which improves the nodes' participation and system operation efficiency by enhancing the DPoS consensus algorithm. The introduction of a reputation mechanism is intended to achieve two principal objectives. Firstly, it is expected to enhance the operational efficiency of the system. Secondly, it is anticipated that it will introduce a mechanism by which to achieve consensus in scenarios involving the governance of opinion. The mechanism is designed to stimulate voter participation and to ensure the integrity of information behaviour on witness nodes.

3.3. Tendermint

Tendermint is a blockchain technology that utilises a committee-based consensus algorithm to achieve consensus among a group of block creators, known as validators, even in the event of some of them being malicious. Figure 7 presents a flowchart elucidating the operational dynamics of the Tendermint consensus mechanism. The election of validators to the committee is a periodic process, based on their investments. In instances where validators possess inadequate assets to undertake investments, they have the option to augment their assets through the assistance of participants, designated as principals, who allocate assets to the validators [34]. Tendermint achieves a more optimal balance between security, energy efficiency, and throughput through a hybrid design of Byzantine fault tolerance (BFT) + proof of stake (PoS), a configuration that is particularly well suited for federated chains and scenarios that necessitate expeditious termination. However, the degree of decentralisation and scalability is limited by the size of the verifier set and the design of the consensus mechanism. This must be weighed against the concentration of interests and network performance.

Lagaillardie and Djari demonstrated that Tendermint, in its capacity as a closed system, exhibits characteristics that may be regarded as inequitable by certain participants. This conclusion was reached through a mathematical analysis, which led to the proposition of Tendermint as an open system. The objective of this study was to ascertain the fairness of the system through a computational analysis [35]. Guenou established the existence of a (final) fair reward mechanism under the condition that the system is (eventually) synchronised. Furthermore, he demonstrated that the original Tendermint reward mechanism is not equitable but can be rendered ultimately fair by minor adjustments to its latency and commit message handling [36]. Buchman's experimental results demonstrate that Tendermint is capable of achieving a throughput of 20,000+ TPS, with latency controlled within 100 ms under normal network conditions. The system exhibits linear scalability, with throughput decreasing less than 20% when the number of nodes increases from 10 to

50. Additionally, Tendermint demonstrates resilience to one out of three node Byzantine failures, ensuring security. The system also exhibits fast recovery under crash failures with RTO < 5 s and consensus under random network latency [37].

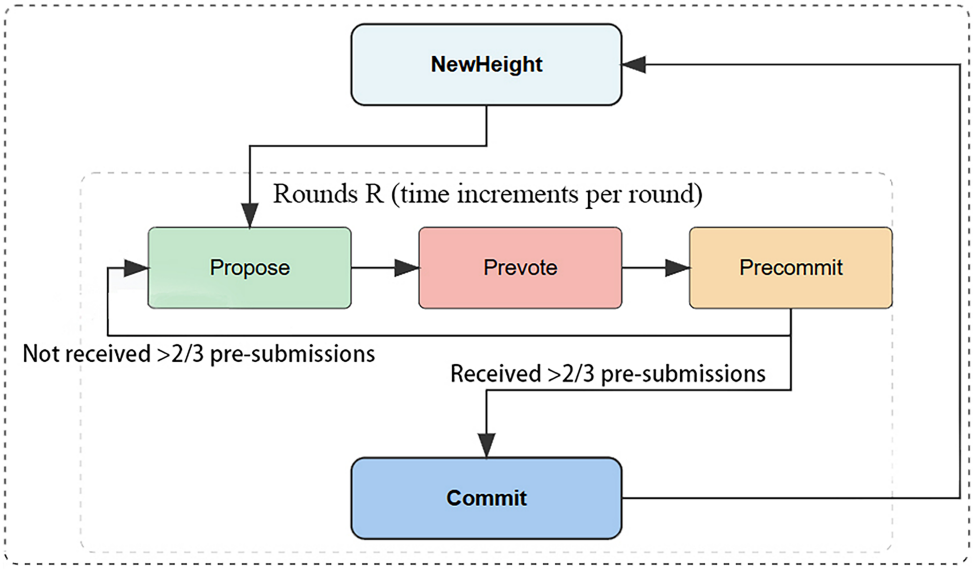

Figure 7. The four-stage cyclical workflow of the Tendermint consensus mechanism.

Tendermint has the capacity to achieve thousands of TPS in certain synchronous networks by means of batch processing transactions and optimising the consensus process (e.g., three-phase fast termination). It also supports a dynamic set of verifiers and adapts to high-throughput demand scenarios. Provided that fewer than one-third of the nodes are malicious, the agreement can be reached and is irreversible, and the punishment mechanism (e.g., the elimination of collateral tokens) has been demonstrated to be an effective measure in deterring malicious behaviour. The verifiers are elected through a proof of stake (PoS) election, with the objective of preventing centralisation. Furthermore, support is given for multi-node participation, with the intention of avoiding centralisation. Additionally, support is given for dynamic adjustment of the set of verifiers through the gossip protocol, with the objective of achieving flexible expansion. Finally, PoW mining is completely abandoned, with only lightweight signature verification being utilised. It has been demonstrated that energy consumption is only a very small percentage of that of PoW, but the fairness based on the Tendermint mechanism is still to be improved [38,39].

3.4. HotStuff

HotStuff represents a pioneering advancement in the field of BFT replication protocols, as it is the first to demonstrate these combined properties and is built around a novel framework. It facilitates the implementation of alternative recognised protocols (e.g., DLS, PBFT, Tendermint, Casper) as well as its own within a unified framework [40].

The HotStuff design incorporates a three-stage core, enabling the new leader to simply select the highest quality control available. A second stage was introduced to allow copies to "change their mind" after a vote without any proof from the leader. This protocol has been developed to address the complexity previously described while concomitantly facilitating a significant simplification of the leader change protocol. In order to provide a more accurate representation of the workflow of the HotStuff mechanism, the diagram shown in Figure 8 has been drawn. In conclusion, the standardisation of virtually all stages of the HotStuff process renders it eminently straightforward to streamline and to change leaders with great facility. The financial implications of a new leader facilitating consensus through

the protocol are not greater than those of the incumbent leader. Consequently, HotStuff facilitates frequent leadership transitions, a practice that is regarded as advantageous within a blockchain context for maintaining chain integrity [41].

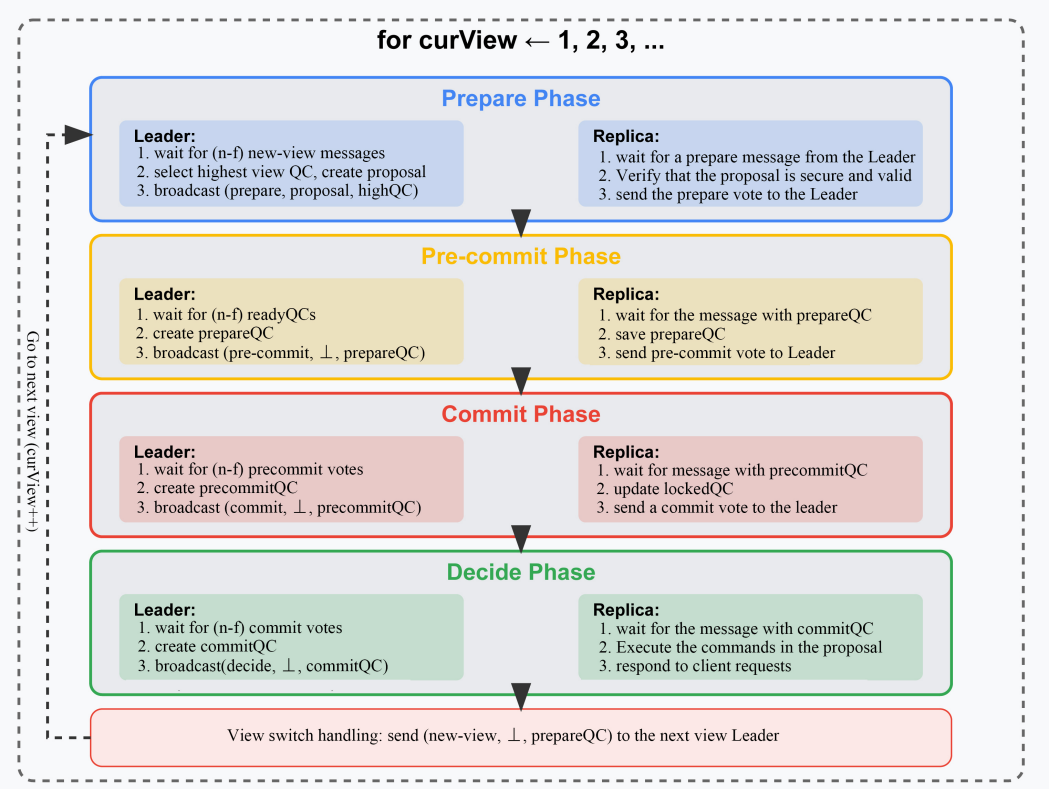

Figure 8. HotStuff.

QC generation and delivery.

- prepareQC: formed by (n-f) prepare votes, proving that the proposal is accepted.
- precommitQC: formed by (n-f) pre-commit votes, nodes lock the block at this stage.
- commitQC: formed by (n-f) commit votes, finally confirms that the block is ready for execution.
- Tolerates up to f Byzantine nodes out of a total number n of nodes. Requirements: $n \geq 3f + 1$

In the context of scalability, Maofan Yin's experimental findings indicate that, despite the symmetric encryption-based MAC employed by BFT-SMaRt exhibiting a significant increase in efficiency relative to the asymmetric encryption utilised in the digital signatures of HotStuff, it is evident that the three-phase HotStuff exhibits a greater number of round-trips in comparison to the two-phase PBFT variant employed by BFT-SMaRt. HotStuff has been shown to attain latency that is analogous to that of BFT-SMaRt while concurrently demonstrating higher throughput. Moreover, the scalability of HotStuff has been proven to exceed that of BFT-SMaRt through the execution of three experiments [40]. In the context of security and throughput, Momose highlighted the forced locking attack vulnerability inherent in the Sync HotStuff protocol and proposed an enhancement scheme to address these concerns. This scheme was designed to enhance the protocol's security and operational integrity across diverse models, encompassing the standard synchronization model, the mobile retardation model, and the response mode [42]. Abraham proposed a simple and practical synchronous Byzantine fault tolerance (BFT) Secure Multi-Party (SMR) protocol, Sync HotStuff. This protocol can tolerate half of the Byzantine replicas, does not require lock-step execution, tolerates slow movement, and improves throughput and scalability

relative to traditional HotStuff, and the slow failure model can capture transient network failures. However, it is not suitable for replicas being offline for an extended period of time [43]. Mingan Gao proposed the MRPBFT consensus algorithm by introducing the multi-master node mechanism and ed25519LRS signature algorithm. These innovations have been shown to significantly improve the performance and security of HotStuff-based consensus algorithms. In consequence, MRPBFT has been demonstrated to outperform existing consensus algorithms in terms of throughput and latency. Moreover, MRPBFT has been shown to exhibit better performance when the number of nodes increases [44]. The Fast-HotStuff algorithm achieves high efficiency through the introduction of small block overheads for view change and responsiveness. Furthermore, it has been shown to maintain robustness when the primary node fails. The experimental findings demonstrate that the Fast-HotStuff algorithm exhibits superior performance in terms of latency and throughput when confronted with fork attacks, thereby demonstrating its enhanced resilience [45].

Dakai Kang's introduction of HotStuff-1 leads to a significant reduction in the client's finality confirmation latency within streamlined Byzantine fault-tolerant (BFT) consensus protocols. The speculative execution and slotting mechanisms employed address the challenges posed by slow leader and tail fork attacks. Additionally, prefix speculation rules are proposed to ensure security, reduce latency, and enhance attack resistance [46,47]. Rong Wang's contributions to the field of distributed systems are significant, with a particular focus on enhancing the transaction ordering speed and overall performance of such systems. The approach adopted by Wang involves the introduction of several key concepts, including the optimistic response assumption, the message aggregation tree, the dynamic adjustment threshold mechanism, the dynamic channelling mechanism, and the asynchronous leader multiround mechanism. The efficacy of these concepts is evident in the substantial enhancement of transaction ordering speed and overall performance, particularly in the context of handling faulty copies and transaction ordering. In comparison with the conventional HotStuff algorithm, the enhanced system demonstrates a notable improvement in responsiveness and efficiency. In response to the chained HotStuff aspect, Jianyu Niu proposed two countermeasures: broadcast QCs and the longest chain rule. These measures have the capacity to significantly reduce latency and prevent attackers from degrading the chain quality through simple attacks [48].

Despite HotStuff demonstrating superior performance in terms of throughput and latency in comparison to previous consensus mechanisms, it is imperative to prioritise the enhancement of its security protocols, particularly in regard to vulnerabilities such as forced locking attacks.

3.5. Proof of History (PoH)

PoH (Proof of History) is an innovative consensus mechanism proposed by the Solana blockchain. It is not a standalone consensus algorithm, but rather a combination of PoSs that work together to improve the efficiency of a distributed system. This is achieved by utilising temporal encoding and segmented clocks to reduce the load on network nodes [15].

The core of PoH is the generation of a time sequence via a verifiable, one-way cryptographic function (usually SHA-256) that proves the order and timing of events. The system essentially provides a decentralised "clock" for distributed systems, thereby reducing the need to synchronise time between nodes. By circumventing the voluminous messages exchanged in traditional consensus, PoH substantially augments transaction processing speed (Solana claims to process tens of thousands of transactions per second) [49]. Nodes have the capacity to verify the sequence of events without the necessity of network negotiation, thereby reducing latency. It is evident that the history is cryptographically secure

and can be verified independently by any party without reliance on trust. However, there is a strong tendency towards centralisation, and PoH relies on the leader to generate the time sequence, which may lead to a single point of risk if the leader selection or switching mechanism is not decentralised enough, and high hardware requirements [50].

It is evident that PoH has a strong tendency to be centralized. The generation of time series is dependent on the leader, which may result in a single point of risk if the selection or switching mechanism of the leader is not sufficiently decentralized. Furthermore, PoH has high hardware requirements. Although throughput of the system is significantly enhanced, elevated standards are demanded for leader selection. Furthermore, the necessity for the enhancement of security and extensibility remains, and the number of related studies is limited.

3.6. Avalanche

Avalanche selects transactions and sends queries through a polling mechanism, and decides whether to accept a transaction based on majority voting. The protocol has been demonstrated to enhance parallelism by forming a DAG [51]. The Avalanche mechanism is a recently proposed distributed consensus protocol by Ava Labs, which belongs to the Snow family of protocols. The protocol combines the advantages of classical consensus and Satoshi Nakamoto Consensus (Nakamoto Consensus) with the objective of achieving high throughput, low latency, and strong scalability. However, there exists a risk of a negligible probability of not reaching a final agreement, and it requires frequent inter-node communication, which is sensitive to network bandwidth and latency. Consequently, this results in high computational and communication overheads under conditions of a large number of competing transactions [52].

Rocket proposed the Snow family of protocols, a leaderless BFT consensus protocol with random sampling and substable mechanisms, which was found to confirm most transactions within approximately 0.3 s when the network size was increased from 125 to 2000 and after disabling signature verification. The latency of the system was only slightly increased by adding 25% of malicious clients and the system throughput was found to be increased by a factor of about 2.6 times to 7002 tps [53,54]. A thorough examination of the consensus algorithm outlined in the Avalanche white paper by Ketchum and Williams exposes deficiencies in its security and operational integrity, thereby illustrating its susceptibility to attacks by malicious nodes [55]. Sesar and Cachin's findings indicate a vulnerability in Avalanche's transaction acceptance process, enabling an adversary to delay the target transaction. This vulnerability was addressed through a modification of the voting mechanism. The restoration of the protocol's activity was achieved through modifications to the voting mechanism [51].

Kniep et al. utilised experimental methods to quantify the survivability and security of the Snowball protocol. This was undertaken to analyse its robustness in the face of malicious attacks. The results of this study revealed a vulnerability to attacks that can be launched by only five actors. In the context of malicious attacks, it has been observed that a mere 2% to 2.8% of nodes are susceptible to such attacks [56]. In a related study, Ullah et al. explored the application of the Avalanche consensus protocol in vehicular communication networks. The consensus mechanism attained a notable 1007 TPS, accompanied by a delay of 1 ms. However, the researchers emphasised the necessity for continued research to enhance scalability in dense scenarios, to devise incentives for verifiers, and to safeguard user privacy through the utilisation of data anonymisation techniques [57]. Doddipatl's proposal of the novel payment system, Avalanche, has been shown to demonstrate superior performance in terms of throughput and low latency when operating within the same

number of nodes and hardware configuration as other systems [58]. This suggests that Avalanche is a viable solution for payment systems.

While the high throughput and low latency of Avalanche have been acknowledged, concerns regarding the security of the consensus mechanism persist. The paucity of research in this area is a matter of concern.

3.7. Hedera Hashgraph

Dr. Leemon Baird, Mance Harmon, and Paul Madsen have identified five fundamental hurdles that must be overcome before distributed ledgers can be widely accepted and adopted by organisations. These hurdles are performance, security, governance, stability, and regulatory compliance. Consequently, the development of Hedera Hashgraph [59,60] was initiated.

In terms of performance, Alahmad analysed the Hedera Hashgraph algorithm from both technical and economic perspectives through experiments. The HBAR cryptocurrency has the capacity to process 10,000 transactions per second; however, in reality, the Hedera Hashgraph network requires the connection of a substantial number of nodes in order to ascertain the true TPS that can be managed by such a network. In terms of economics, the HBAR (Hedera cryptocurrency) has an average cost of USD 0.0001 per transaction in the Hedera network. By comparison, Bitcoin (utilising a functional blockchain network) has an average cost of USD 0.20 per transaction in the blockchain network. It is also observed that the performance of HBAR is resilient and has the potential to continue making a significant impact in the future, with regard to its speed, performance, and efficiency [60]. Subsequently, Amherd proposed a series of metrics and methodologies for the analysis of the decentralisation level of Hedera Hashgraph. The analysis revealed a consistent increase in the number of transactions conducted on the Hedera network, as well as a steady rise in the number of active accounts per day. However, the ratio of active accounts to total accounts remains low. Concurrently, the degree distribution of Hedera Hashgraph evinces a pronounced skewed distribution, bearing a striking resemblance to that of Bitcoin and Ether. It is evident that the core size of the network undergoes a gradual decrease, exhibiting minimal fluctuations even subsequent to the removal of early users. Concurrently, the Kamoto index demonstrates a marked increase, signifying a pronounced degree of centralisation within the network. The Gini coefficient approaches one, indicating that a small number of accounts possess a significant proportion of the wealth, and the Theil-T index for public accounts commences its decline after reaching its peak in mid-2022 [61]. Roh proposed a deep learning network parameter management system based on Hedera Hashgraph to ensure the security and reliability of parameters, thus broadening the ideas for the practice of Hedera Hashgraph consensus [62].

It appears that, despite Hedera Hashgraph's initial alignment with the concept of blockchain and its potential for omnipotence, there has been a gradual divergence from the ideal range in terms of gradual centrality and economic efficiency over time.

3.8. Ripple

Ripple is a decentralised, open-source payment system based on a credit network that enables decentralised currency exchange, payment, and clearing functions. Due to the opacity surrounding the merits of the Ripple mechanism, a flowchart has been included below in Figure 9 to illustrate the process. Schwartz believed that the consensus mechanism at that time should be improved in terms of Correctness, Agreement, and Utility. The Ripple Protocol Consensus Algorithm (RPCA) is therefore applied by all nodes every few seconds to maintain the correctness and consistency of the network. Once a consensus has been reached, the current ledger is considered to be "closed" and becomes the final ledger

to be closed. Assuming the efficacy of the consensus algorithm and the absence of forks in the network, it is theorised that the final closed book maintained by all nodes on the network will be identical and convergent, thus terminating the consensus process in a finite amount of time [63].

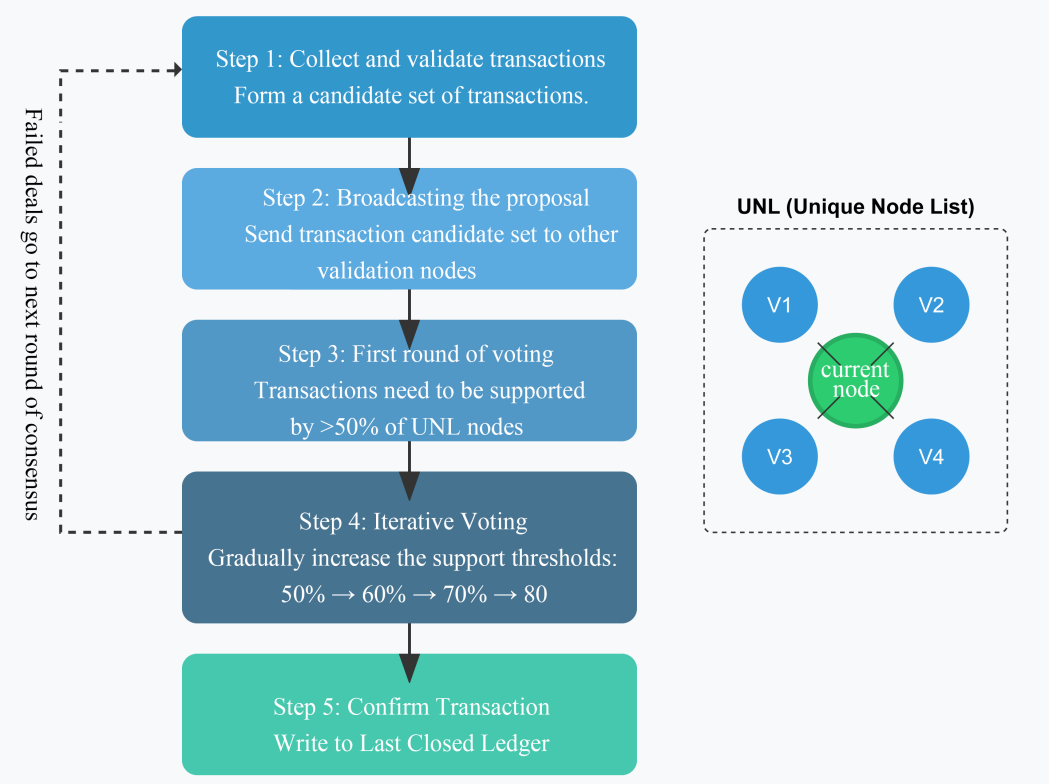

Figure 9. The five-step workflow of the Ripple Protocol Consensus Algorithm.

In Ripple's consensus algorithm, the identities of the participating nodes are known in advance, rendering it more efficient than anonymous consensus algorithms such as PoW. Furthermore, the confirmation of transactions is expeditious, taking only a few seconds. It is evident that this point also determines that the consensus algorithm is only suitable for permissioned chain scenarios. The Byzantine fault tolerance (BFT) capability of the Ripple consensus algorithm is $(n - 1)/5$, which means that a transaction is approved only if 80% of the UNLs of the servers agree to it, and as long as 80% of the UNLs are honest, no fraudulent transaction will be approved [64].

Armknecht draws parallels between Ripple and Bitcoin, noting that both systems rely on ECDSA signatures to ensure the authenticity and non-repudiation of transactions. Ripple, being an open-source system, facilitates the detection of double payment attempts and incorrectly formatted transactions. Ripple's inherent support for fast payments, with the majority of ledgers closing within seconds, suggests that payments can be verified within seconds of execution. In the context of Bitcoin, transactions may utilise inputs from multiple accounts. In the Ripple system, payments are typically initiated from one account. Despite the implementation of measures to safeguard user identity in Ripple and Bitcoin, the user's transaction behaviour (i.e., the time and amount of the transaction) is exposed during the process due to the public announcement of the transaction within the system. It is also noted that when Wavecoin is deployed centrally, the majority of the verification servers are operated by Ripple Labs [65].

In contrast, Todd's work highlighted concerns pertaining to scalability, privacy, and jurisdiction, as evidenced by the simulation of consensus splitting, transaction flooding, validator duress, software backdoor, validator key theft, and simulated ledger attacks [66].

Luzio et al. conducted a thorough investigation of the Ripple system that revealed that its consensus process is dependent on a limited number of validators, leaving it vulnerable to potential attacks. These attacks could result in the de-anonymization of users with a high degree of precision, as evidenced by the analysis of side-channel information from individual transactions. This represents a significant threat to user privacy and underscores the necessity for enhancements to be made to enhance the system's robustness and user privacy protection [67]. In their seminal work, Baseera et al. pioneered a hybrid technique that integrates two distinct types of fish and the Ripple Consensus Algorithm (TF-RC) to ensure the secure collection and storage of data through blockchain technology based in the cloud. This pioneering approach also implements an intelligent intrusion detection method to enhance blockchain security [68]. Building on this foundation, Li's research utilised the Ripple Consensus Algorithm to develop an intelligent, automated cross-border payment system in e-commerce. This innovative system aims to improve transparency, automation, and efficiency in managing international transactions [69]. What is more, the recent Ripple's virtual currency XRP has exhibited high levels of activity and significant demand. With regard to technological innovation, Ripple is also making progress. In addition to fundamental payment functions, Ripple is developing a smart contract platform, with the objective of supporting a more diverse range of application scenarios.

Chapter Summary

The objective of high-throughput consensus mechanisms is to optimise transaction processing speed and network capacity, thereby supporting high-frequency transactions. The mechanisms examined in this chapter realise this objective through diverse design choices, yet they are also subject to their own set of trade-offs.The performance metrics of the consensus mechanism for high throughput are summarised in Table 4.

Table 4. High-throughput consensus mechanism performance indicators.

	Throughput	TPS	CT	Security	FT	CoA	Energy Consumption	CPT	HR	Scalability	NS	CC
Multi-paxos [70–72]	Very High	10 k+	100 ms	High	2f + 1	High	Low	Low	Standard Server	Moderate	Low	O(n)
DPoS [23,24,73–75]	Very High	3 k–10 k	0.5–3 s	Moderate	3f + 1	High	Very Low	Low	Standard Server	High	11–100	$O(n^2)$
Tendermint [76,77]	Very High	4 k–10 k	1–3 s	High	3f + 1	High	Very Low	Low	Low	High	1000+	$O(n^2)$
HotStuff [28,45,78–80]	Very High	16 k+	0.5–2 s	High	3f + 1	High	Low	Low	Standard Server	Very High	Moderate	O(n)
PoH [49,50]	Very High	6.5 k++	0.4–1 s	Moderate	3f + 1	High	Very Low	Low	Standard Server	Moderate	Moderate	O(n)
Avalanche Consensus [81–83]	Very High	4500+	1–2 s	Moderate	3f + 1	High	Low	Very Low	Moderate	High	1 k+	O(n·logn) (k is a safety parameter)
Hedera Hashgraph [59–61,84]	Very High	10 k+	3–5 s	High	3f + 1	High	Low	Low	High	Moderate	Moderate	O(k·n·logn)
Ripple [85–89]	High	1.5 k–3 k	3–5 s	Moderate	5f + 1	High	Low	Low	Standard Server	Moderate	Moderate	O(n)

CT: Confirmation time. **FT:** Fault tolerance. **CPT:** Consumption per transaction. **NS:** Node scalability. **CoA:** Cost of attack. **HR:** Hardware requirements. **CC:** Communication complexity.

These mechanisms commonly employ strategies such as reducing consensus participants, optimizing communication protocols, or introducing parallel processing. For example, Multi-Paxos, DPoS, Tendermint, and HotStuff all accelerate consensus through some form of leader election or committee mechanism, thus avoiding the inefficiency of network-wide competition in PoW. PoH and Avalanche further enhance throughput by implementing parallel processing through time-series coding and DAG structures.

High throughput is often accompanied by a sacrifice in the degree of decentralization or dependence on specific hardware/network conditions.

Decentralization vs. efficiency trade-off: DPoS improves efficiency by delegating to a few representatives, but at the cost of potentially concentrating power in the hands of a few

nodes. Hedera Hashgraph also shows a tendency towards centralization. The Ripple protocol, due to its permissioned chaining nature, is efficient but has limited decentralization.

Security versus performance trade-off: Although HotStuff and Tendermint provide high security through the BFT mechanism, their communication complexity still limits the node size. Avalanche, despite its high throughput, suffers from security vulnerabilities and sensitivity to network bandwidth as its main challenges. PoH achieves high throughput while facing the risk of centrality and strong dependence on the leader.

Hardware requirements and popularity: PoH has high hardware requirements, which may limit its widespread adoption. The core design and application scenario refinement analysis for high throughput are summarised in Table 5.

Table 5. High-throughput consensus mechanism core design and application scenario refinement analysis.

Consensus Mechanisms	Core Design and Trade-Offs	Applicable Scenarios
Multi-Paxos	Stable leader reduces communication rounds for high throughput and strong consistency. Leader failures require reselection and may bottleneck under high load.	Distributed logging, state machine replication, private chaining
DPoS	Delegated representatives accelerate consensus with extremely low energy consumption.There is a risk of power concentration, but scalability is high.	Public blockchains (such as EOS), consortium blockchains
Tendermint	BFT + PoS hybrid, fast final confirmation. The size of the validator set limits decentralization and scalability.	Consortium blockchains, scenarios requiring rapid final confirmation
HotStuff	Three-phase BFT protocol, simplified leader switching, high throughput, and low latency. Sensitive to network latency, security protocols need to be strengthened.	High-performance permissioned blockchains, DApps requiring rapid final confirmation
PoH	Time-series encoding enables parallel processing and extremely high throughput. Relies on a leader, with high hardware requirements. Magnetization.	High-performance public blockchains (such as Solana)
Avalanche	Random sampling + DAG achieves high throughput and low latency. Risks of non-final confirmation, high communication overhead, and insufficient security research.	Payment systems, large-scale networks
Hedera Hashgraph	Hashgraph consensus offers high throughput, fast confirmation, and low energy consumption. Centralization trend, with complex hardware configuration.	Enterprise-level applications, consortium blockchains
Ripple Protocol	Known node identities enable efficient and fast payments. Only applicable to permissioned chains, with centralization risks and privacy issues.	Cross-border payments, financial institutions

The core design of the high-throughput consensus mechanism and refinement analysis of application scenarios are shown in Table 5. A review of the extant literature reveals a consensus among researchers that these consensus mechanisms exhibit trade-offs between throughput, security, energy consumption, and scalability.

Multi-Paxos and Raft (to be discussed in the section on flexible scalability) are classic choices in traditional distributed systems. They excel in strong consistency but their scalability is typically limited by the number of nodes, making them more suitable for private or enterprise-level scenarios.

DPoS and Tendermint enhance throughput by reducing the number of nodes participating in consensus, but this also raises concerns about centralization. DPoS is widely adopted in public blockchains, while Tendermint excels in consortium blockchains.

HotStuff represents the latest advancement in BFT protocols, achieving extremely high throughput and rapid finality through optimized communication rounds, though its security remains an ongoing concern.

PoH and Avalanche achieve parallel processing through innovative data structures (such as time series and DAG), resulting in remarkable throughput, but they each face challenges in terms of decentralization and security.

Hedera Hashgraph and Ripple Protocol demonstrate high throughput in specific scenarios, but their high degree of centralization makes them more suitable for permissioned or consortium blockchain environments.

- Future trends:

The future development of high-throughput consensus mechanisms will focus on further optimising parallel processing capabilities, reducing communication costs, and improving decentralisation and security while maintaining performance. Hybrid consensus mechanisms and sharding technology (to be discussed in the Flexible Scaling chapter) will be key to achieving these goals.

4. Strong Security

High-security-oriented consensus mechanisms are designed to defend against various forms of attack, including 51% attacks, double payments, and malicious nodes. These mechanisms also ensure data consistency and immutability. Such mechanisms typically guarantee security through high computational costs, financial penalties, or Byzantine fault-tolerant (BFT) designs.

4.1. Proof of Work (PoW)

PoW is considered to be one of the most common consensus algorithms, and the concept was first introduced by Cynthia Dwork and Moni Naor in 1993 in an academic paper [90]. The term POW was introduced in 1999 in an article by Markus Jakobsson and Ari Juels [91], but it was not until 2008 with the release of Bitcoin by Satoshi Nakamoto [4] that it attracted significant public attention. POW functions by requiring nodes to solve mathematical puzzles (e.g., SHA-256 hash calculations) to validate transactions and generate new blocks. The security of this system is predicated on the high computational costs of the process, but the energy consumption of the system is a significant problem. In order to facilitate a more comprehensive visualisation of the workflow of the PoW consensus mechanism, a diagram has been drawn, as illustrated in Figure 10.

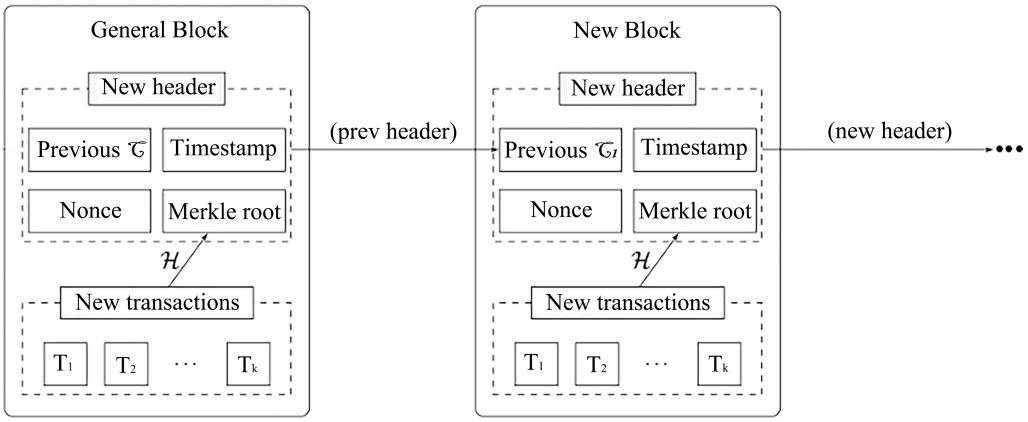

Figure 10. The core structure and linking process of the PoW blockchain.

In recent years, while Bitcoin continues to rely on PoW, numerous projects have sought to reduce their environmental impact through the utilisation of renewable energy sources, such as solar mining. A significant number of PoW mining operations (e.g., Bitcoin mining) are beginning to transition towards the utilisation of renewable energy sources, such as hydro, wind, or solar power. For instance, certain mining operations have opted to conduct their activities in regions abundant in hydropower, such as Sichuan, China, or Scandinavia, with a view to reducing their carbon footprints. The adoption of a bill in New York State in 2022 that also promotes the utilisation of renewable energy sources for mining

is propelling the industry's transition to green energy. For PoW, a multitude of variants have been generated, including PoMW, HPoW, PoWT, dPoW, ePoW, SSPoW, and numerous others [92]. As illustrated in Figure 10, the performance metrics of the PoW mechanism variants are summarised in Table 6.

Table 6. PoW variant performance indicators.

	Platform	**Scalability**	**Maintenance Cost**	**Validator Selection Criteria**	**Mining Profitable**
PoW	Bitcoin	High	Very High	Computation based	Yes
PoMW [93]	Venelium	–	Very High	Computation-based	Yes
HPoW [93]	Lynx	Moderate	Very High	Vote-Based	No
PoWT [94]	Vveujum	High	Low	Vote-Based	Yes
dPoW [93]	Komodo	High	Low	Vote-Based	Yes
ePoW [95]	HDAC	–	Low	Computation-based	Yes
SSPoW [96]	Purple	–	–	Computation-based	–

	Scalability	**51% Attack Chances**	**Energy Consumption**	**Block Generation Time**
PoW	Low	High	Very High	10 min
PoMW	–	High	Very High	–
HPoW	Low	High	Very Low	30 s
PoWT	High	High	Low	15 s–6.2 min
dPoW	Very High	Low	Low	1 min
ePoW	High	–	Low	3 min
SSPoW	Very High	–	–	15 s

Among the variants, the "dPoW" and "HPoW" algorithms perform the best in the comparison of all parameters and can be used as an alternative to PoW algorithms for future use.

The core advantage of the PoW mechanism lies in its extremely high security, which is ensured by the deployment of massive computational power. In order to launch an attack, attackers would need to control over 51% of the network's total computing power. This is an extremely costly endeavour in large proof-of-work networks, thereby providing robust security. However, it should be noted that this level of security is accompanied by a significant energy expenditure and a comparatively low transaction throughput. The competitive nature of the mining process leads to substantial energy consumption, and due to the fixed block generation time, its transaction processing speed is significantly slower than that of other consensus mechanisms. Furthermore, the scalability of PoW is constrained due to the requirement that each full node must validate all transactions, thus impeding the network's capacity to process a substantial volume of transactions.

4.2. Proof of Useful Work (PoUW)

In contrast to conventional PoW which utilises computational resources exclusively, PoUW proposes the application of mining computation for the resolution of real-world optimisation problems (e.g., combinatorial optimisation problems) [97]. In 2021, several studies were conducted proposing the practical application of mining for the resolution of scientific computing or data processing tasks. The role of miners in the validation of transactions is predicated on the resolution of complex mathematical and scientific problems. These problems, which may include the folding of proteins or the simulation of weather patterns, are solved by miners and subsequently submitted to the network for verification. This approach theoretically reduces energy expenditure; however, its practical implementation remains encumbered by technical complexity and substantial validation costs. Notable projects such as Primecoin [98], Coinami [99], and CoinAI [100] employ searching for specific types of prime chains, multiple sequence comparison of protein sequences, or training of deep learning models as useful working consensus algorithms.

In terms of security, PoUW retains the original security of PoW, but PoUW applies the wasted arithmetic resources in PoW to solving existing problems. This results in a more rational use of resources and a reduction in pointless wasteful behaviour.

The objective of PoUW is to address the energy consumption issue inherent to PoW by leveraging computing resources for significant tasks. This approach is intended to ensure the continued high security of PoW while concurrently enhancing its social value. Nevertheless, the primary challenge confronting this approach is the effective verification of the correctness of "useful work" and the coordination of the computing time of these complex tasks with the block generation time of the blockchain. This results in longer confirmation times and higher technical complexity and verification costs in practical applications.

4.3. Paxos

The Paxos algorithm is a distributed consistency algorithm based on message passing. The algorithmic flow is delineated in Figure 11. It was proposed by Lamport in 2001. Paxos is the first proposed consensus algorithm to achieve efficient data consistency through three roles (acceptor, proposer, and learner). Its most important feature is that it is difficult to understand and implement [101]. Paxos demonstrates robust scalability through its modular design, fault-tolerant mechanisms, Multi-Paxos optimisation, and partition parallelisation. These features render it well suited for deployment in small and medium-sized distributed systems.

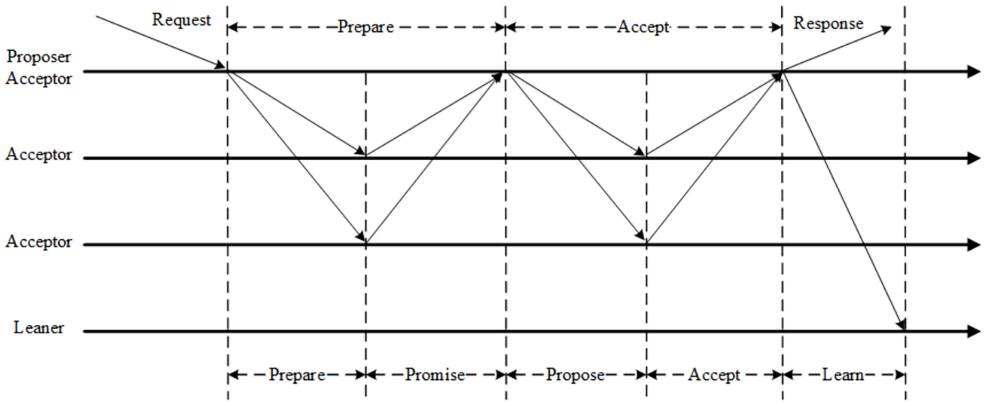

Figure 11. The message-passing and phase-based workflow of the Paxos consensus algorithm.

Lamport's seminal contributions to the field of consensus algorithms began with the proposal of Paxos, a pioneering approach to enhancing the efficiency of command execution while preserving the security and fault tolerance characteristics of classical Paxos. Building on this foundation, Lamport introduced the concept of generalized Paxos, a sophisticated algorithm that not only optimised the efficiency of command execution but also ensured the security and fault tolerance of classical Paxos [102]. Subsequently, Lamport's seminal contributions advanced the field through the introduction of a novel distinction between fast rounds and classical rounds, along with the innovative utilisation of uncoordinated recovery or coordinated recovery in the event of collisions. This pioneering approach led to the attainment of efficient consensus, thereby resolving the latency issue prevalent in traditional consensus algorithms in asynchronous systems [103]. In the field of distributed systems, Kończak proposed the FullSS, ViewSS, and EpochSS algorithms to address the issue of state recovery in the event of a Paxos state machine failure. The primary objective of these algorithms is to facilitate the rapid recovery of the state of the crashed replica, thereby ensuring the continued availability of the service [104]. In a related study, Marandi presented two Paxos-based efficient atomic broadcast protocols: M-Ring Paxos and U-Ring Paxos. The primary function of these protocols is to enhance

throughput [105]. Skrzypczak proposed RMWPaxos, a system that facilitates "in situ" learning of consensus decisions in a single distributed state. This approach circumvents the necessity for costly state management and complex logging operations. By circumventing the necessity for costly state management and intricate logging operations, it offers a cost-effective, highly scalable solution [106]. In a seminal paper, Srinivasan [107] proposed a new Paxos-based algorithm that has the potential to bring about a significant reduction in the time taken to process recovery. This is achieved by enabling processes executing lower rounds to make decisions regardless of the presence of higher rounds in the system, with a view to minimising the impact of recovered processes starting higher rounds. The time to process recovery is reduced from 17δ to 11δ, and even down to 5δ under certain conditions. Wang presented an RDMA-based Paxos protocol and its runtime system, APUS, and demonstrated through experimental means that APUS is fast, scalable, and easy to deploy [108]. MWOTIL proposed the LowPaxos protocol, which is specially designed for low-resource environments. The LowPaxos protocol improves the performance of state machine replication by selecting the best leader and adapting to environmental changes, and outperforms the adaptation in heterogeneous environments compared with traditional leader and leaderless protocols [109].

Paxos has the capacity to tolerate up to f node failures in a 2f + 1 node system, thereby ensuring strong consistency in distributed systems. It guarantees that all nodes ultimately reach the same state and maintains consistency even in the presence of failures. Paxos has a complete theory, does not rely on specific trust nodes, and is suitable for decentralisation. However, the algorithm design and implementation of Paxos is more complex, requiring multiple rounds of communication (proposal, acceptance, confirmation, etc.), and the performance may be affected under high-latency or unstable network conditions, and when the number of nodes increases dramatically, the communication overhead of Paxos rises significantly, which restricts its application in large-scale distributed systems. In conclusion, Paxos is a powerful consensus mechanism suitable for distributed systems requiring strong consistency and high availability. However, the inherent complexity and performance overhead of this system render it more suitable for deployment within private or enterprise-level scenarios as opposed to large-scale public networks. Furthermore, the primary function of this system is to provide crash fault tolerance (CFT) rather than Byzantine fault tolerance (BFT).

4.4. Federated Byzantine Agreement (FBA)

FBA, proposed by David Mazieres in 2015, is a consensus algorithm for open distributed systems. Figure 12 illustrates the FBA mechanism. FBA ensures the security and fairness of the system by balancing the rights and trust relationships of the nodes. FBA can guarantee the correct and reliable messaging in the case that the number of failed or malicious nodes does not exceed one-third of the total number of nodes. The advantages of FBA are high decentralisation and good network scalability, but the disadvantage is that the security and activity depend on the structure of the trust graph [110].

Stellar represents the inaugural secure and certifiable implementation of FBA, an open-source distributed payment network that facilitates the transfer and exchange of a broad spectrum of currencies and assets. In 2015, Stellar transitioned to FBA as its consensus algorithm, formally designated as the Stellar Consensus Protocol (SCP). The SCP enables each node to autonomously select the nodes to be trusted, thereby forming a federation based on the trust relationship. Notably, the SCP is capable of attaining consensus expeditiously within the network while ensuring security and decentralisation. Consequently, the SCP exhibits four properties concurrently: decentralised control, low latency, a flexible trust mechanism, and asymptotic security [111]. The feasibility of the

SCP has been formally demonstrated [112–114]. Innerbichler proposes the implementation of the FBA algorithm, which incorporates a hierarchical quorum structure. This system is capable of achieving consensus in the presence of malicious or failed nodes. In the event of such an occurrence, the FBA mechanism ensures the security and activity of the system and prevents it from entering a stalemate state [115]. As Kim et al. have previously observed, FBA is not as decentralised as it should be but is clearly a centralized blockchain [116]. The ZOI was thus designed with a reputation mechanism to incentivise all peers to become verifiers, thus improving the centralisation of the stellar system that measures the reputation of each verifier in a democratic way, thus gaining the trust of the rest of the network [117]. Florian proposed a series of concepts, including minimum blocking set, minimum splitting set, and top level, with the aim of characterising which groups of nodes may potentially compromise activity and security. Consequently, this facilitates the execution of a comprehensive risk assessment in FBAS-based systems, such as the Stellar Network [118]. Tumas investigated the robustness of the FBA mechanism in the face of targeted cyber-attacks. It was found that an attacker only needs to disconnect the top 9% most highly connected nodes to stop the blockchain. The robustness was improved by replacing the connections of some of the highly connected nodes with those of the lowly connected nodes. This resulted in an increase in network robustness to 52% and quorum robustness to 45% [119].

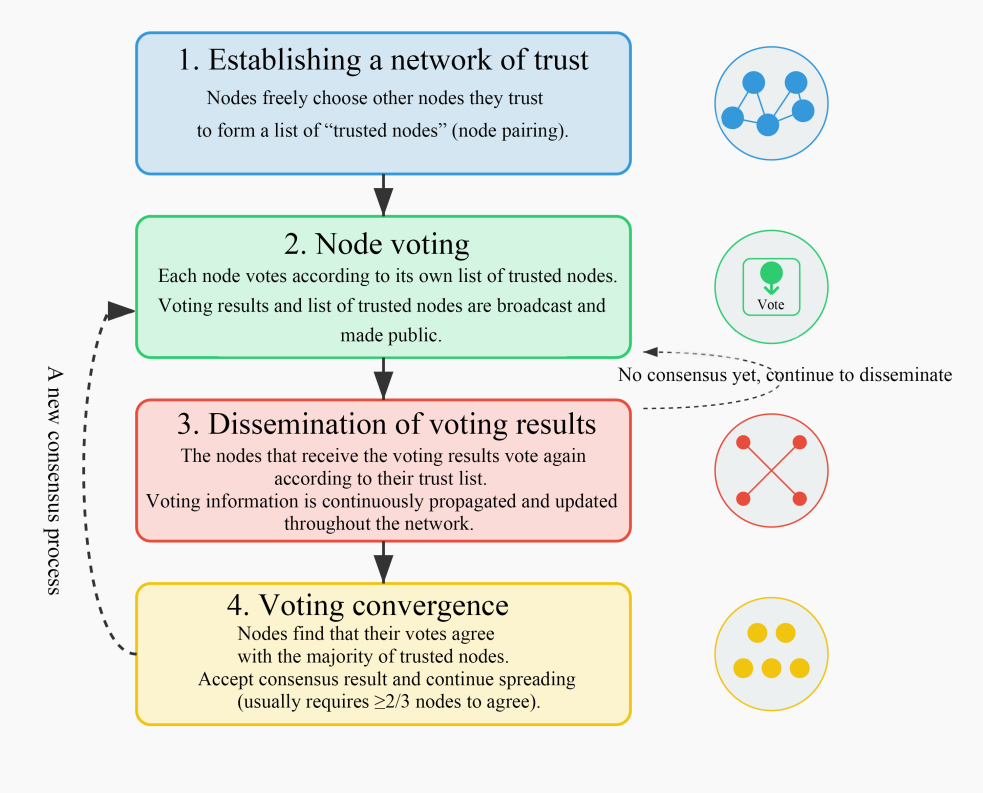

Figure 12. The four-step process of the FBA mechanism.

In conclusion, while the FBA model has been demonstrated to ensure the delivery of messages with a success rate of no more than one-third of those that are failed or malicious, its security and activity are nevertheless dependent on the structure of the trust graph. This issue has been addressed by some of the existing variants of the research. Despite the fact that FBA is characterised by a greater degree of decentralisation in comparison with traditional BFT, the establishment and maintenance of its trust model has the po-

tential to result in a certain degree of centralisation risk, particularly in the context of practical implementation.

4.5. Practical Byzantine Fault Tolerance (PBFT)

PBFT is a consensus algorithm in distributed systems that has been designed to solve the Byzantine fault problem. The flow of this consensus mechanism is illustrated in Figure 13. The latter occurs when nodes may send incorrect or malicious data. This approach was first proposed by Castro and Liskov in 1999 for scenarios such as Permissioned Blockchain. Since its introduction as the first practical Byzantine fault tolerance (BFT) replication solution in the partial synchronisation model, many BFT solutions have been built around its core two-stage paradigm. PBFT requires that the total number of nodes in the network, N, satisfies $N \geq 3f + 1$, ensuring that agreement can be reached despite the presence of up to f malicious nodes.

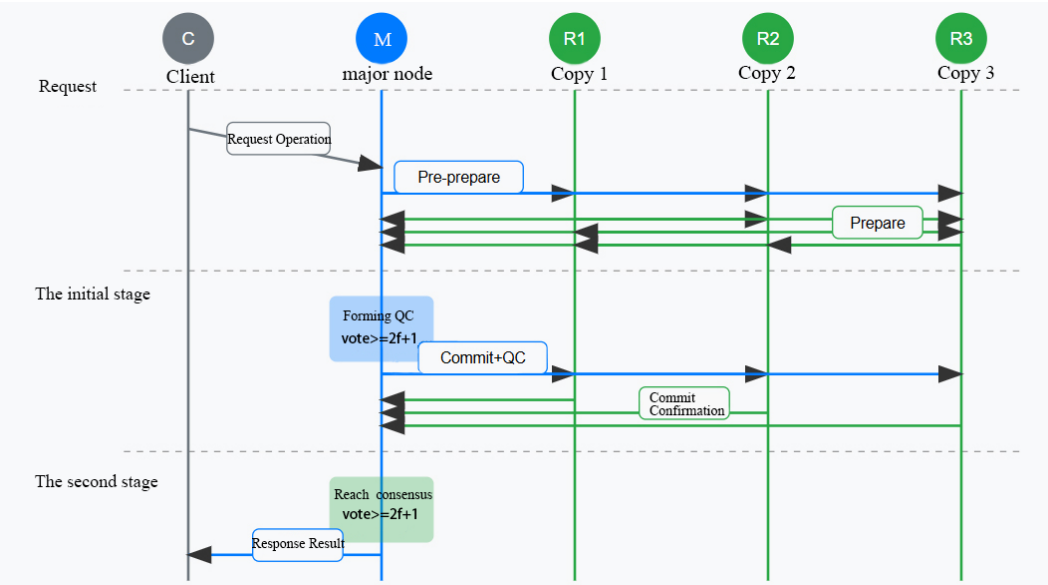

Figure 13. The request–response and multi-phase workflow of the PBFT consensus mechanism.

The variants of PBFT include the following: REBFT, Honey Badger BFT, RBFT, WBFT, s-PBFT, SBFT, Scalable Hierarchical PBFT, T-PBFT, IPBFT, APBFT, Casper-PBFT, and BFT-SMaRt. These variants have been optimised and improved in different ways in terms of decentralisation level, energy efficiency, scalability, throughput, etc., to address different application scenarios and challenges [120]. As illustrated in Figure 13, the performance metrics of the PBFT mechanism variants are summarised in Table 7.

PBFT serves as the cornerstone of Byzantine fault tolerance, and its variants have been shown to significantly improve performance by reducing communication complexity, optimising latency, and enhancing asynchronous support. Examples of systems that excel in terms of scalability and throughput include HotStuff, Tendermint, and BFT-SMaRt. HoneyBadgerBFT is uniquely suited for asynchronous networks, while Casper-PBFT and RBFT, on the other hand, fulfil the needs of public blockchain and enterprise scenarios, respectively. Nevertheless, there is still scope for improvement with regard to scalability, asynchronous fault tolerance, and cross-chain interoperability. The primary limitation of PBFT is its high communication complexity, which is expressed as $O(n^2)$. This inherent property of PBFT results in a significant decline in performance as the number of nodes increases. Consequently, it is more appropriate for permissioned chain environments with a limited number of nodes. Future research should concentrate on layered architectures,

hybrid consensus, and green technologies to promote the wide application of PBFT in Web3, IoT, and enterprise blockchains.

Table 7. PBFT-Variant performance indicators.

	Decentralization Level	Permissioned/ Permissionless	Energy Efficient	Scalability	Throughput
REBFT [121]	Semi-centralized	Both	Low	Medium	Low
Honey Badger BFT [122]	Semi-centralized	Permissionless	Moderate	Medium	High
RBFT [123]	Decentralized	Permissionless	Moderate	Strong	High
WBFT [124]	Decentralized	Permissioned	Moderate	Low	Medium
s-PBFT [125]	Decentralized	Permissioned	High	Medium	Low
SBFT [126]	Decentralized	Both	High	Strong	Medium
Scalable historical PBFT [127]	Decentralized	Permissioned	Moderate	Medium	Medium
T-PBFT [128]	Decentralized	Semi-permissioned	High	Medium	High
IPBFT [129]	Decentralized	Both	High	Strong	High
APBFT [130]	Semi-centralized	Permissioned	High	Strong	High
Casper-PBFT [131]	Decentralized	Permissionless	High	Strong	High
BFT-SMaRt [132]	Semi-centralized	Permissioned	High	Medium	Very High

4.6. Casper

In order to provide protection against both extended revisions and disastrous system failures, Vitalik Buterin and Virgil Griffith developed Casper, an overlay that offers additional protection against block reversals for virtually all proof-of-work chains [133]. Casper is a partial consensus mechanism that combines PoS and BFT, incorporating two slashing conditions, a correct fork selection rule, and a dynamic set of validators. The mechanism under discussion is capable of defending against long-range revision attacks and catastrophic crashes by introducing an inactive leakage mechanism. In addition, experimental work conducted by Moindrot investigated the various parameters of Casper, and it was found that consensus can be reached even under unfavourable conditions. The algorithm demonstrates exceptional robustness to disconnected nodes and reliably reaches consensus even under high latency [134]. In order to ensure security while keeping the protocol active, and to design effective reward and punishment mechanisms to incentivize participants to follow the protocol as well as to maintain the ultimate consistency of the protocol in the presence of network partitions, Buterin proposed a hybrid Casper Friendly Final Deterministic Gadgets (FFG) protocol for solving the problems during the transition from PoW to PoS in Ethernet [135]. In this study, Levrand employs an analysis of the Casper blockchain to reveal the current state of wealth inequality and degree of decentralisation. The analysis provides valuable insights into understanding wealth inequality and fairness issues in PoS blockchains by showing that, although Casper's reward mechanism leads to wealth inequality in the long run, in comparison to alternative PoS networks, this system has been demonstrated to be both more equitable and to accumulate wealth at a more gradual rate [136]. Veschetti conducted a formal analysis of blockchain consensus protocols utilising the PRISM+ model checker, thereby demonstrating that the Hybrid Casper protocol exhibited considerable resilience against both eclipse and majority attacks, thus proving effective in its resistance to these forms of attack [137].

The Casper mechanism has been demonstrated to exhibit particular strengths in terms of energy efficiency, security, and decentralisation through the combination of PoS and PBFT, rendering it a particularly suitable solution for Web3, DeFi, and enterprise blockchain applications. Nevertheless, the communication overhead, pledge centralisation risk, and protocol complexity of hyperscale networks limit their performance. In the future, Casper is expected to play a greater role in the blockchain ecosystem by optimising communication, enhancing asynchronous support, and mitigating centralisation issues.

4.7. Proof of Devotion (PoD)

The fundamental principle of the PoD algorithm is as follows: users who have surpassed the designated NR level are involved in the validators' selection process through the payment of a deposit, thereby becoming bookkeeping candidates. Each bookkeeping candidate engages in a competitive endeavour to secure the bookkeeping right through virtual mining. The user who successfully obtains the bookkeeping right assumes responsibility for generating blocks, acquiring block rewards, and generating income from transaction fees. In the event of a user engaging in malevolent behaviour, their deposit will be confiscated and redistributed to other candidates for the bookkeeping position [138]. The Byzantine problem can be resolved because the cost of malevolence is exorbitant.

PoD does not require a significant amount of computational resources, is more energy-efficient, and focuses more on the actual contribution of users to the network. The specific steps of the algorithm are illustrated in Figure 14. It reduces the over-reliance on token holdings and increases the consideration of the contribution dimension. Furthermore, it does not rely on a few delegated nodes but rather evaluates the behaviours of all the participants more broadly. However, the implementation of PoD may vary from project to project, and the specific rules need to be carefully designed, or else it may lead to non-transparency or unfairness. Furthermore, the efficacy of PoD is contingent on the level of activity and user participation within the blockchain ecosystem. It has been demonstrated that the mechanism may not function optimally in the absence of sufficient community participation [139]. This process assists in ascertaining the legitimacy of proposed blocks. Moreover, in order to eradicate the titular probabilistic monopoly that may eventuate in a monopoly, PoD grants bookkeeping ownership to designated nodes.

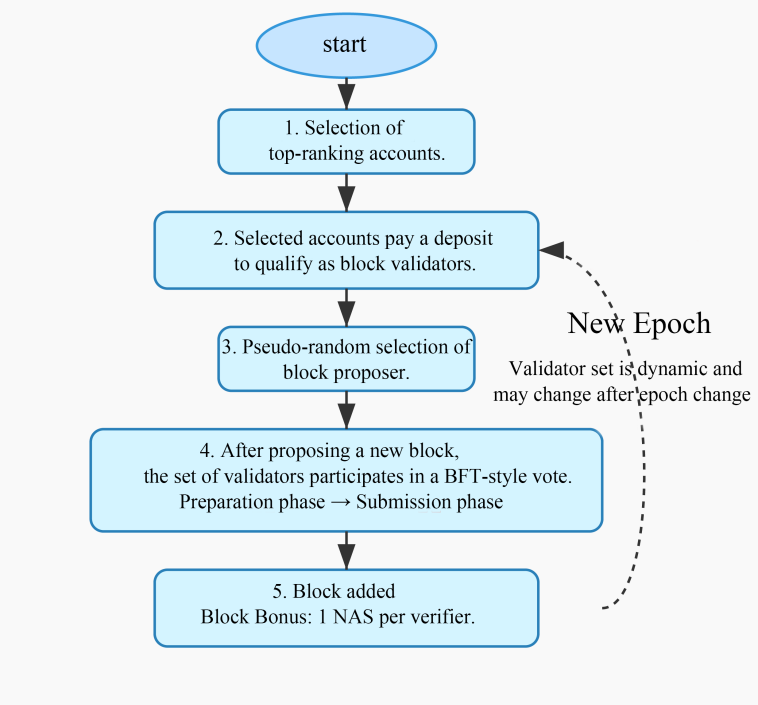

Figure 14. The step-by-step process of the PoD consensus algorithm.

The PoD model utilises the Nebulas Rank algorithm to quantify user contributions, thereby combining the advantages of both the PoS and the PoI models. Users may be rewarded for running nodes, participating in DApp development, or providing data indexing. The mechanism has been designed to establish a "positive feedback loop" with the objective of promoting the long-term development of the blockchain ecosystem. It is

evident that Nebula Chain utilised this technology; however, on 30 December 2024, the Nebula mainnet node officially ceased operation.

The objective of PoD is to achieve fairer decentralisation and higher security by linking user contributions to accounting rights while avoiding the high energy consumption of PoW. The primary benefit of this approach is that it utilizes economic penalty mechanisms to deter malicious behaviour, thereby addressing the Byzantine problem. Nevertheless, the efficacy of PoD is contingent on the transparency and fairness of its rule design, as well as the active participation of the community. In the event of inadequate design or an absence of community engagement, there is a possibility of the emergence of new centralisation risks or inefficiency.

4.8. Delegated Byzantine Fault Tolerance (DBFT)

Tyler Crain and Vincent Gramoli found that if processes are permitted to execute asynchronous rounds subsequent to receiving a specified number of messages without awaiting messages from potentially slow-responding coordinators, the resultant decentralisation is particularly appealing to blockchains. This is due to the fact that each node performs a similar function in enforcing the consensus, thereby rendering decisions inherently "democratic". The solution is scalable due to the avoidance of bottlenecks by balancing loads, thus giving rise to DBFT [140].

DBFT is a consensus algorithm that supports large-scale participation in consensus through proxy voting. In the context of Ant Neo, the inaugural public chain to be produced in China, token holders are empowered to exercise their democratic right by casting votes for their preferred validators. The selected group of validators, subsequently chosen based on their expertise and capabilities, employs the BFT algorithm to achieve consensus and generate a new block. It is noteworthy that the voting process in the Neo network is characterised by its real-time nature and its absence of personalisation, ensuring a level playing field for all participants.

It has been determined that Neo's DBFT mechanism requires between 15 and 20 s to generate a block. Transaction throughput is measured at approximately 1000 TPS, which is considered to be a satisfactory performance level for a public blockchain. With certain optimisations, DBFT has the potential to reach 10,000 TPS, thereby enabling support for large-scale commercial applications. DBFT provides $f = n - 13$ fault tolerance for consensus systems with one consensus node. This fault tolerance also covers security and availability, is immune to general and Byzantine errors, and is suitable for any network environment. DBFT has good finality, meaning that once a block has been finalised, it cannot be forked, and a transaction cannot be undone or rolled back. DBFT incorporates digital identity technology, meaning that the validator can be a real person or an institution. Consequently, the DBFT can be frozen, revoked, inherited, retrieved, and have token exchange rights based on a judicial decision that exists within itself. The Neo network has been designed to facilitate the registration of compliant financial assets, and to support this process when necessary. However, this approach is not without its drawbacks, as evidenced by the fact that there is a high level of competition among individuals to become the root chain. It is conceivable that there may be multiple root chains in it [141,142].

The present study proposes an approach to enhance the efficiency and security of the DBFT algorithm through the implementation of honest behaviour. This approach is intended to address the limitations of the existing DBFT algorithm, as outlined in [143]. In contrast, Jeon proposes a solution that involves the incorporation of a randomly selected set of verifiers into the DBFT framework, thereby enhancing the probability of identifying malicious activity [144]. This approach facilitates the participation of a substantial number of users without compromising system efficiency.

DBFT attains high efficiency, high throughput, and deterministic finality through the integration of the delegation mechanism and BFT, rendering it suitable for scenarios necessitating rapid confirmation and elevated security (e.g., enterprise blockchain, smart economy). Its low energy consumption and flexible governance further enhance the application potential. Nevertheless, there are several challenges to be addressed, including limited decentralisation, the risk of node election, and scalability bottlenecks. The governance complexity and reliance on node trust may have a detrimental effect on long-term stability.

Chapter Summary

The various designs of high-security consensus mechanisms aim to resist attacks and ensure data consistency. The mechanisms discussed in this chapter provide security while demonstrating different trade-offs in other performance areas.

The performance metrics of the consensus mechanism for strong security are summarised in Table 8.

Table 8. Strong-security consensus mechanism performance indicators.

	Throughput	TPS	CT	Security	FT	CoA	Energy Consumption	CPT	HR	Scalability	NS	CC
PoW [145–149]	Very Low	7	60 min	Very High	2f + 1	High	Very High	50–800 kw/h	ASIC	Low	-	O(n)/ O(n·logn)
Proof of Useful Work [97,150–154]	Very Low	35	2 h+	Very High	2f + 1	High	Very High	Very High	High-end GPUs	Low	-	O(n)/ O(n·logn)
Paxos [102,107,155–162]	Low	-	1 s+	High	2f + 1 (CFT)	Moderate	Low	Low	Standard Server	Low	Low	O(n)
FBA [110,112–114,154]	High	1.5 k–3 k	3–5 s	High	3f + 1	Very High	Low	0.222 w/h	Low	High	High	O(n,s,k)
PBFT [119,130,163]	Moderate	800+	Fast	High	3f + 1	Moderate	High	Moderate	Standard Server	Moderate	Low	$O(n^2)$
Casper [132–137]	High	1000+	-	High	3f + 1	High	Low	Low	Low	High	High	$O(n^2)$
Proof of Devotion [138,139]	Moderate	-	Fast	High	3f + 1	High	Low	Low	Standard Server	Moderate	-	$O(n^2)$
DBFT [140,141]	High	1k+	Fast	High	3f + 1	High	Low	Low	Standard Server	High	Low	$O(n^2)$

CT: Confirmation time. **FT:** Fault tolerance. **CPT:** Consumption per transaction. **NS:** Node scalability. **CoA:** Cost of attack. **HR:** Hardware requirements. **CC:** Communication complexity.

These mechanisms are centred around establishing trust and resisting malicious behaviour. PoW ensures security through enormous computational power, while BFT-type protocols (such as PBFT, DBFT, and Casper) tolerate Byzantine faults by achieving consensus among a majority of nodes. FBA builds a security model through a trust graph. Meanwhile, PoUW and PoD aim to enhance efficiency and fairness by introducing "useful work" or "proof of contribution", all the while ensuring security.

Security often involves trade-offs between efficiency, decentralization, and energy consumption.

Security vs. energy consumption: PoW offers the highest level of security but at the cost of extremely high energy consumption. PoUW attempts to mitigate this issue through "useful work," but still faces challenges in terms of efficiency and verification complexity. In contrast, PoS and BFT-based protocols (such as Casper and DBFT) ensure security while significantly reducing energy consumption.

The trade-off between security and scalability: PBFT offers strong security, but its communication complexity ($O(n^2)$) limits scalability as the number of nodes increases. Paxos faces similar issues. FBA and DBFT enhance scalability to some extent through delegation or federation mechanisms, but their degree of decentralization and reliance on trust models require attention.

Security vs. decentralization trade-off: PoW is highly decentralized but inefficient. FBA and DBFT achieve a certain degree of decentralization, but the construction and maintenance of their trust models may still lead to centralization risks. Casper and PoD

aim to find a better balance between decentralization and security. The core design and application scenario refinement analysis for strong security are summarised in Table 9.

Table 9. Strong-security consensus mechanism core design and application scenario refinement analysis.

Consensus Mechanisms	Core Design and Trade-Offs	Applicable Scenarios
PoW	Computing power competition ensures security, but energy consumption is extremely high and throughput is low.	Bitcoin, the public blockchain requiring the highest level of security.
PoUW	Useful work improves resource utilization, but verification is complex and confirmation takes a long time.	Blockchain combining scientific computing and data processing
Paxos	Message passing achieves strong consistency, but it is complex and only provides fault tolerance.	Small and medium-sized distributed systems, private chains
FBA	Trust graph construction is secure and highly decentralized, but relies on trust structures.	Open distributed systems, enterprise chains (such as Stella)
PBFT	The BFT protocol provides fast finality and high security, but node scalability is limited.	Permissioned blockchain, consortium blockchain
Casper	Proof of contribution + deposit penalties are energy-efficient and decentralized, but rule design and community participation are key.	Blockchain that incentivizes user contributions
DBFT	Proxy voting + BFT, high efficiency, high throughput, and deterministic final confirmation. Limited decentralization, node election risk.	Enterprise Blockchain, Smart Economy

PoW and PoUW excel in terms of security, but their high energy consumption and low throughput pose challenges for modern blockchain applications.

Paxos is a classic in distributed systems, offering strong consistency, but its complexity and limited support for crash tolerance restrict its application in public blockchains.

FBA and DBFT offer efficient and secure solutions in enterprise-level and consortium blockchain scenarios through their unique trust models and delegation mechanisms, but their degree of decentralization and reliance on trust remain ongoing concerns.

PBFT and its variants perform well in permissioned blockchains, providing fast finality and high security, but their scalability is limited by the number of nodes.

Casper represents the direction of public blockchains towards low energy consumption and high throughput, achieving a balance between security, energy efficiency, and decentralization through the combination of PoS and BFT.

PoD is an innovative attempt to enhance security and decentralization through user contributions, but its long-term effectiveness still requires further validation.

- Future trends:

The future development of high-security consensus mechanisms will focus on improving efficiency and scalability without compromising security. This includes optimising existing BFT protocols (e.g., reducing communication complexity), exploring new trust models (e.g., reputation-based mechanisms), and combining sharding and hybrid consensus technologies to overcome performance bottlenecks. At the same time, defending against quantum attacks will also become an important area of research.

5. Low Energy Consumption

Low-energy-oriented consensus mechanisms are designed to minimise the energy consumption and hardware resources necessary to operate the network. These mechanisms prioritise environmental sustainability and cost-effectiveness, thereby contributing to the development of sustainable energy technologies. It is evident that such mechanisms frequently act as a substitute for computationally intensive tasks, such as mining in PoW,

opting instead for methods of verification that are less onerous, including those related to equity and storage.

5.1. Proof of Stake (PoS)

The success of Bitcoin is predicated on the PoW mechanism, which, however, is energy-consuming, resulting in high operational costs. Consequently, Sunny and Scott proposed the PoS mechanism as an alternative to reduce energy dependence and increase economic efficiency [164]. However, the genesis of PoS can be traced back to 2011, when an individual operating under the pseudonym Quantum Mechanic first proposed proof of stake on the Bitcointalk forum, a prominent Bitcoin community forum [165]. The consensus mechanism employs a voting process to elect validators based on the number of tokens held, thereby reducing resource consumption. It is evident that PoW is contingent on the limited availability of computer hardware to impede the occurrence of witch attacks. Conversely, PoS is predicated on the tokens inherent within the blockchain itself [166]. As illustrated in Figure 15, the workflow of a proof-of-stake (PoS) consensus mechanism can be summarized in six key stages. First is the "Staking Phase", where users must stake a minimum of 32 ETH to become a validator or delegate their stake through liquid staking services like Lido. This is followed by the "Validator Selection Phase", where the system uses a randomness beacon (RANDAO) to randomly select validators for each epoch (approximately 6.4 min). The selected validators are assigned to a committee, with one designated as the block proposer and the others as attesters. The third stage is "Block Production", which occurs every 12 s slot. The selected proposer collects transactions to create a new block, potentially using MEV (Maximal Extractable Value) to optimize rewards. In the "Validation and Attestation Phase", the committee members verify the new block and submit attestations (votes). Finally, in the "Finality and Rewards Phase", a block becomes final and irreversible after receiving attestations from over two-thirds of the validators, which takes about 12.8 min. Successful participants receive rewards, while validators who are offline or exhibit malicious behaviour face penalties, including having their staked assets slashed. In the end, the final block will be added to the blockchain and the next slot process will be repeated. In a PoS ecosystem, in addition to the core role of validators, there are two equally crucial participants: delegators and relay nodes, who work together to maintain the network's security, decentralization, and efficiency.

In a PoS ecosystem, in addition to the core role of validators, there are two equally crucial participants: delegators and relay nodes, who work together to maintain the network's security, decentralization, and efficiency. Delegators hold cryptocurrency but lack the willingness or technical capability to run a validator node. They participate indirectly in staking by delegating their tokens to one or more reputable validators. This mechanism significantly lowers the barrier to entry for ordinary users, allowing them to support the network and earn rewards by contributing their stake. However, delegators must choose their validators carefully, as their staked tokens are also at risk of slashing if the chosen validator behaves maliciously or goes offline. In some complex PoS protocols, particularly in the MEV-Boost architecture following the Ethereum Merge, relay nodes play a critical intermediary role. They act as a bridge connecting block builders and block proposers. Builders are responsible for creating full blocks that contain potentially high-value transactions (such as MEV) and sending them to the relay node. The relay node, in turn, provides the highest-bidding block header (without the full transaction contents) to the proposer. This process protects the proposer's privacy and prevents malicious behaviour while ensuring the proposer receives the maximum possible reward. These three roles form an efficient collaborative workflow: delegators increase a validator's stake weight, the validator (as a proposer) uses the relay node to securely select and create the most profitable

block, and ultimately, all participants—both the validator and the delegators—share in the block rewards for their contributions. This complex network of interactions allows the PoS system to balance security with efficiency and inclusivity.

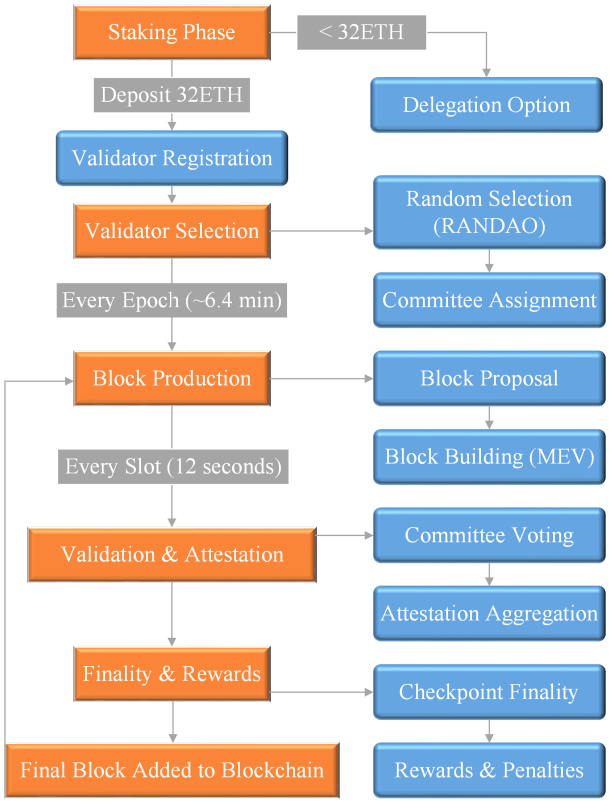

Figure 15. The detailed workflow of the PoS consensus mechanism as implemented on Ethereum.

As is typical of consensus mechanisms, there are numerous other PoS-based consensus mechanisms. These include the Ouroboros protocol, which is a pure PoS protocol [167], and the Chains-of-Activity protocol, where the selection of the leader is based on the tokens and the hash value of the previous block [168]. The Byzantine fault tolerance (BFT) protocol is utilised to facilitate the confirmation of each fixed-interval checkpoint block of the Casper protocol [133]. The Algorand protocol employs a cryptographic lottery mechanism to select the leader and committee members based on the nodes' private keys and seeds [169,170]. The Tendermint protocol utilises a BFT voting protocol to confirm blocks, wherein verifiers gain voting rights through deposits, etc. [166]. Li's data-driven approach indicates that Polkadot exhibits superior performance in terms of decentralisation and robust fairness. However, it is observed to have a limited number of verifiers. Tezos and Cardano demonstrate a greater degree of openness; nevertheless, the distribution of wealth is gradually becoming centralised, and small verifiers may not be adequately rewarded. Following an update, Casper's fairness has been enhanced, though it continues to exhibit fluctuations [171]. Mišić has proposed the implementation of Qualified Proof of Stake (QPoS) to address the issue of power concentration, with the aim of incentivising and penalising honest behaviour among voters and leaders. The performance of the system is analysed through the utilisation of an embedded Markov chain model [172].

The PoS mechanism has been demonstrated to achieve low energy consumption, high throughput, and scalability. These properties make it suitable for both decentralised blockchain and enterprise applications. Nevertheless, the principal challenges pertain to pledge centralisation risk, initial pledge cost, and finality latency. Addressing the challenges

posed by governance complexity and security issues necessitates the optimization of protocols and incentives [166].

5.2. Proof of Burn (PoB)

Karantias was the first to provide a formal definition of a proof-of-destruction protocol. The system comprises two functions: firstly, a function that generates a cryptocurrency address. In the event of a user sending funds to a designated address, the funds are permanently irrevocably destroyed. Secondly, it is imperative to incorporate a verification function that will ensure the address in question is indeed unspendable. The following properties are proposed for the destruction protocol: unspendability, which specifies that an address correctly verified as a destruction address cannot be used for spending; binding, which allows associating metadata with a specific destruction; and uncensorability, which specifies that a destruction address is indistinguishable from a regular cryptocurrency address [173].

Rodinko proposes a multi-currency auction model based on PoBs, which implements a secure and decentralised system for transferring funds from an existing cryptocurrency to a secure, decentralised upgrade of new tokens. The model eliminates the need for external information sources, circumvents the KYC (know your customer) procedure, and facilitates fair price discovery and fund allocation [174]. The Burn-to-Claim cross-chain interoperability protocol is meticulously modelled and validated, and the results demonstrate that the protocol is correct, secure, and weakly atomic under the assumptions, though strong atomism does not always hold [175]. This approach is analogous to PoW insofar as it involves the distribution of work or investment into the system. However, rather than energy, it utilises a supply of "spent" coins. The energy consumption of the proposed system is comparatively low, as evidenced by the findings of PoW [176].

Proof of destruction is a novel consensus mechanism that allocates mining rights by destroying tokens. This approach focuses on the high energy consumption of PoW and theoretically provides better decentralisation and market scarcity. Nevertheless, the industry is confronted with considerable challenges, including resource wastage, economic inequality, and intricacy. Furthermore, the feasibility of PoB in large-scale applications still requires further verification. In conclusion, PoB is a potentially valuable option, albeit one which must be assessed on a case-by-case basis in order to ascertain its applicability to a given project, and the specific needs and objectives thereof.

5.3. Proof of Elapsed Time (PoET)

PoET is a consensus algorithm that has been developed by Intel as an alternative to traditional proof-of-work algorithms. The PoET mechanism workflow diagram is illustrated in Figure 16. In PoET, as in Bitcoin's PoW, network participants are assigned mining rights based on random wait times rather than solving complex mathematical puzzles. This approach is designed to ensure fairness and prevent any participant from gaining an advantage over the others when verifying transactions and adding new blocks to the blockchain. The implementation of random wait times is intended to achieve two objectives. Firstly, it is expected that PoET will be able to achieve efficient consensus. Secondly, it is anticipated that energy consumption will be minimised. A high level of security is also maintained [176,177].

In recent years, Chen evaluated the security of PoET-based blockchain systems through theoretical analysis and mathematical modelling. This research identified vulnerabilities in the system in the event of trusted computing component failures. The author proposed a change to the probability distribution, allowing statistical tests to reject blocks generated by certain proportions of nodes. This, it is claimed, will improve the security of

the system [178]. Pal's proposal of a DC-PoET mechanism is of particular interest. This mechanism, through the implementation of a distributed coordination process, has the capacity to significantly reduce the probability of orphaned blocks. Furthermore, it has been demonstrated to enhance transaction throughput, exhibit high robustness in the case of uncompromised TEE, and demonstrate resilience against double-spending and selfish mining attacks [179]. Bowman proposed an ET consensus protocol based on PoET and PoL for trusted execution environments, and demonstrated that even in the case of a possible attack on TEE, it can still be realised through the introduction of the z test function, thereby enabling the achievement of a scalable and secure consensus [180]. B-LPoET, a lightweight PoET mechanism for private chains, was proposed, with the aim of improving the efficiency and security of distributed transaction execution through multithreading technology, and effectively reducing the delay and resource consumption of consensus execution while supporting more nodes [181]. Kumar proposed front-end IoT applications based on Bitcoin and PoET [182].

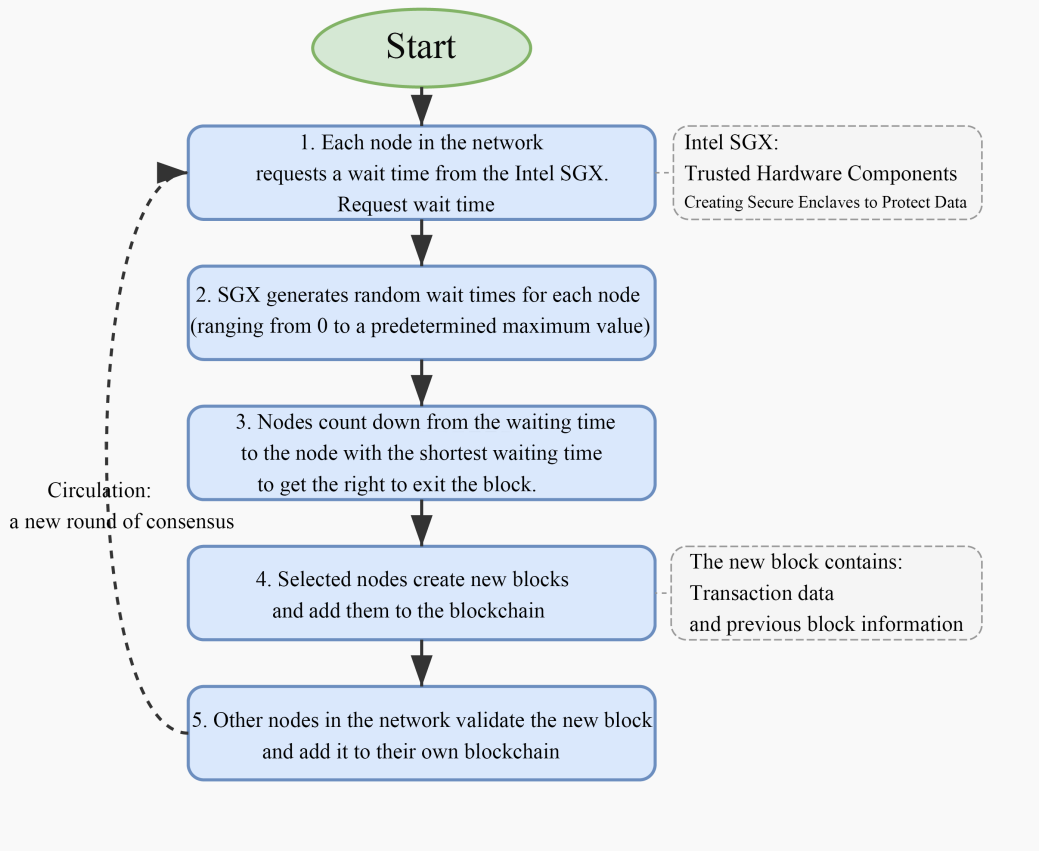

Figure 16. The workflow of the PoET consensus mechanism using Intel SGX.

In summary, the PoET mechanism has a low cost of participation; more people can easily join and thus decentralize; it is easier for all participants to verify that leaders are legitimately elected; and the cost of controlling the leader election process is proportional to the value gained from it. However, it should be noted that specific hardware must be used, and this is not applicable to public blockchains [182].

5.4. Proof of Capacity (PoC)

PoC was first proposed by Dziembowski et al., thereby establishing the theoretical foundation, otherwise referred to as proof of space. This is a disk space-based PoS that aims to reduce the consumption of computational resources by constructing graphs with high tiling complexity and using Merkle hash trees to achieve an efficient and secure PoS

scheme [183]. Subsequently, the Burstcoin team implemented this concept in a tangible blockchain project in 2014, thereby becoming the inaugural cryptocurrency to utilise PoW. Ateniese et al. presented two PoSpace protocols under ROM. PoSpace offers a novel defence against denial-of-service attacks by necessitating the provers to utilise a designated amount of space. The paper also provides a definition of weak PoSpace (wPoSpace) [184]. Jiang proposed a non-cooperative game-based storage offloading method for optimising resource allocation in PoC-based mobile blockchain mining [185]. Recently, Signum introduced proof of commitment (PoC+) decentralised consensus to the world as an evolution of proof of capacity (PoC) consensus. It is asserted that the mining process in the PoC+ consensus is so efficient and has such minimal hardware requirements that any consumer PC can be used for mining. It is noteworthy that the process of mining is imperceptible to the user, with the exception of the sporadic blinking of the hard disk drive (HDD) [186].

In general, PoC systems demonstrate superior energy efficiency, attributable to their reduced energy consumption in comparison to computing power. They exhibit a modest hardware threshold, necessitating only a standard hard disk for participation. Nevertheless, these systems allocate a substantial amount of storage capacity to redundant data, which can be exploited by an attacker to launch a man-in-the-middle attack by renting storage space. This results in wasted storage resources and compromised security.

5.5. Proof of Authority (PoA)

The concept of PoA was initially put forward by Gavin Wood, the founder of Ethereum. PoA involves the pre-selection of a limited number of trusted nodes (referred to as validators or authorities) with the function of validating transactions and creating new blocks. In contrast to PoW and PoS, PoA is predicated on the reputation and identity of these verifiers as opposed to computing power or pledged assets [187].

Angelis [188] has highlighted that PoA algorithms offer inadequate guarantees with regard to consistency, particularly in scenarios where data integrity is paramount. Alrubei proposed an honesty-based distributed proof-of-authorisation (HDPoA) mechanism based on the PoA mechanism. This enables efficient and secure network operation in resource-constrained IoT devices through scalable workloads [189].

The PoA model is predicated on the Byzantine fault-tolerant paradigm, a characteristic that ensures its security. The utilisation of a centralised reputation mechanism serves to enhance its efficiency, while its scalability is notable, albeit with the caveat that transactions require verification by an approval system. The centralised PoA configuration is particularly well suited for implementation in actual IoT devices [190]. It is evident that PoA exhibits superior energy efficiency, expedited operation, and enhanced security when contrasted with conventional PoW, PoS, and other consensus mechanisms [191]. However, it should be noted that PoA does possess a certain degree of centralisation.

Chapter Summary

The aim of low-energy consensus mechanisms is to minimise the energy and hardware resources required to operate the network while prioritising environmental sustainability and cost-effectiveness.

The performance metrics of the consensus mechanism for low energy consumption are summarised in Table 10.

These mechanisms typically achieve low energy consumption by avoiding computationally intensive tasks, such as mining in PoW, and instead using verification methods based on equity, time, or identity. PoS elects validators by staking tokens; PoET uses random waiting times in a trusted execution environment; PoC relies on storage space; and PoA is based on the reputation of pre-selected authoritative nodes.

Table 10. Low-energy consensus mechanism performance indicators.

	Throughput	TPS	CT	Security	FT	CoA	Energy Consumption	CPT	HR	Scalability	NS	CC
PoS [164–166,192–195]	High	30–1 k+	13–20 min	High	3f + 1	High	Very Low	2×10^{-2} kwh	NO	High	High	$O(n^2)$
PoB [174–177,180]	Low	-	-	Moderate	3f + 1	Moderate	Very Low	2×10^{-6} kwh	NO	Low	-	$O(n^2)$
PoET [45,176]	High	Very High	Moderate	Moderate	3f + 1	Moderate	Low	Close to 0	SGX chips	Moderate	Moderate	O(n)
PoC [183,186]	Low	30	10 min	Moderate	3f + 1	Moderate	Low	10^{-6} kwh	High-capacity hard drives	Moderate	Moderate	O(n)/ O(n·logn)
PoA [187–192,196]	High	1300+	3–5 s	Moderate	3f + 1	-	Very Low	10^{-6} kwh	NO	Low	Low	$O(n^2)$

CT: Confirmation time. **FT:** Fault tolerance. **CPT:** Consumption per transaction. **NS:** Node scalability. **CoA:** Cost of attack. **HR:** Hardware requirements. **CC:** Communication complexity.

Low energy consumption often involves trade-offs between decentralization, security, or dependence on specific hardware.

Energy consumption and decentralization trade-offs: PoS has extremely low energy consumption but poses risks of staking centralization. PoA is highly energy-efficient and fast but has the highest degree of centralization. PoET and PoC perform well in terms of energy consumption, but PoET depends on specific hardware, while PoC faces storage waste and security vulnerabilities.

Energy consumption and security trade-off: PoS and PoA maintain high security while being energy-efficient, but PoC and PoB face their own security challenges, such as PoC being susceptible to storage rental attacks, and PoB facing resource waste and economic inequality issues.

Hardware dependency: while PoET and PoC are energy-efficient, they have specific requirements for SGX chips and high-capacity hard drives, respectively, which may limit their universality in certain scenarios.

The core design and application scenario refinement analysis for low energy consumption are summarised in Table 11.

Table 11. Low-energy-consumption consensus mechanism core design and application scenario refinement analysis.

Consensus Mechanisms	Core Design and Trade-Offs	Applicable Scenarios
PoS	Staking tokens to elect validators significantly reduces energy consumption but poses risks of centralization and initial staking costs.	Modern public blockchains (such as Ethereum 2.0) and enterprise applications
PoB	Destruction of tokens allocates mining rights, which is energy efficient but faces issues of resource waste and economic inequality.	Theoretical research, specific token issuance
PoET	Message passing achieves strong consistency, but it is complex and only provides fault tolerance.	Small and medium-sized distributed systems, private chains
PoC	Trust graph construction is secure and highly decentralized but relies on trust structures.	Open distributed systems, enterprise chains (such as Stellar)
PoA	The BFT protocol provides fast finality and high security, but node scalability is limited.	Permissioned blockchain, consortium blockchain

PoS has become the mainstream consensus algorithm, favoured for its high throughput and low energy consumption, with Ethereum 2.0 being a prime example. It is particularly well suited for modern blockchains.

PoA performs well in enterprise scenarios and permissioned chains, but it carries a higher risk of centralization.

PoET and PoC offer innovative solutions, but they must address hardware dependency and security issues. PoET is vulnerable to storage rental attacks, while PoC relies on trusted hardware.

- Future trends:

The future development of low-energy consensus mechanisms will focus on further enhancing their decentralization and security while maintaining their inherent energy efficiency advantages. This includes optimizing existing protocols and exploring new incentive mechanisms and governance models to mitigate centralization risks. In addition, combining privacy protection technologies such as zero-knowledge proofs will enable their application in more privacy-sensitive scenarios.

6. Flexible Scaling

Flexible scale-oriented consensus mechanisms are designed to adapt to changes in network size (e.g., an increase in the number of nodes or cross-chain collaboration) while maintaining performance and stability. Such mechanisms achieve efficient scaling through modular design, sharding techniques, or dynamic trust models.

6.1. Egalitarian Paxos (EPaxo)

EPaxos is a leaderless consistency algorithm where any replica has the capacity to commit logs, which generally necessitates one or two network round trips for a single log commit. It synthesises the merits of Basic Paxos and Multi-Paxos by introducing a dynamic ordering that balances both efficiency and availability.

In 2012, Moraru proposed the Egalitarian Paxos system with the objective of achieving optimal commit latency when tolerating one and two failures in the WAN. The proposed system eliminates the bottleneck of a single leader and evenly balances the load across all replicas. Furthermore, EPaxos facilitates graceful degradation of performance and continuous availability in the event of a slow replica or system crash, thereby enhancing system scalability and availability for distributed computing environments necessitating high throughput and low latency [197,198]. Moraru subsequently accomplished a distributed system that exhibited high availability, high performance, and stable performance through leaderless design, load balancing, and fast commit latency. This accomplishment served to substantiate the correctness of EPaxos, a distributed consensus protocol that was the first to achieve all three objectives concurrently [199]. Sutra then explores possible problems with the implementation of the egalitarian Paxos protocol, particularly regarding how to handle command dependencies when switching between ballots. It is noted that the repetitive consensus process of EPaxos must rely on two ballot variables to keep track of the progress of each replica, otherwise security may be compromised [200]. Junior proposed a method for utilising EPaxos's dependency information to execute SMR commands in parallel, a strategy that has the potential to enhance throughput, particularly in scenarios where the cost of command execution is high or the conflict rate is elevated [201].

In summary, Egalitarian Paxos is a high-performance, decentralised consensus algorithm that is particularly well suited to low-conflict, high-concurrency distributed systems. The primary benefits of this approach are its leaderless design and concurrency processing capabilities. However, it is important to note that there are limitations in terms of complexity, conflict sensitivity, and network requirements. Consequently, EPaxos is more appropriate for private or enterprise-level scenarios characterised by high performance and decentralisation requirements. However, it should be noted that trade-offs may be necessary in environments characterised by high conflict or resource constraints.

6.2. Raft

In 2014, Diego Ongaro and John Ousterhout proposed Raft, a simplified version of the Paxos algorithm. Raft is characterised by its ease of understanding and implementation, as well as its suitability for distributed systems and master–slave replication. A key distinction

of Raft is its lack of support for "evil nodes", which is a feature that sets it apart from other algorithms. This characteristic leads to its frequent utilisation within private chains. A fundamental tenet of the Paxos algorithm is its decentralised nature, as it does not incorporate a leader proposer role. This attribute is a hallmark of a pure, decentralised distributed algorithm. The fundamental Paxos algorithm is notable for the absence of a leader proposer role, thus ensuring complete decentralisation. However, it is important to note that the algorithm is not without its drawbacks, which include the capacity for only a single-value consensus, the occurrence of live locks, and a high network overhead. It is evident that the Multi-Paxos algorithm (a decentralised algorithm to a leader-based algorithm) is based on the leader proposer. Furthermore, the Raft algorithm can be regarded as a further optimisation of Multi-Paxos, primarily by the addition of two restrictions:

- Firstly, the sequentiality of log addition must be considered. Raft necessitates the sequential addition of logs, whereas Multi-Paxos permits concurrent addition, obviating the requirement for logs to be in sequence. Consequently, logs may be absent.
- Secondly, the selection of master restrictions: Raft is a protocol that requires only the node with the most recent logs to be elected leader because logs are added serially. This enables Raft to confirm the most recent node based on the logs [202]. Howard demonstrated that the Raft protocol outperforms existing Paxos variants in terms of comprehensibility, correctness, and performance by implementing and evaluating the Raft consensus algorithm [203].

In the Multi-Paxos algorithm, the addition of logs occurs concurrently, which hinders the identification of the node with the most recent logs. Consequently, Multi-Paxos is unable to select a leader proposer node, and it must generate the other logs after it becomes the leader node [204]. Howard then compared the Paxos and Raft algorithms and found that Raft is more efficient in the leader election process because it allows only the servers with the latest logs to become the leader, thus reducing the exchange of log entries [205].

On this basis, Yang proposed the cell-based Raft algorithm (CBR) to reduce the burden on the leader node by reducing the number of messages and determining the optimal cell size through a federated learning model to improve the throughput and stability of the system [206]. Lu proposed the P-Raft consensus algorithm, which significantly improves the coalitional chain consensus efficiency and security. The algorithm can effectively elect the node with optimal performance as the leader, reduce the probability of invalid election, shorten the average election period, and enhance the robustness of the blockchain network [207].

The proposed RaBFT algorithm by Bai has been shown to enhance the efficiency of leader election and system stability through the integration of a verifiable secret sharing technique and a dynamic committee mechanism. This approach has been demonstrated to reduce the frequency of elections, leading to substantial improvements in throughput and consensus latency. Additionally, the algorithm has exhibited notable efficacy in the context of Byzantine fault tolerance [208]. Fu's contributions include the introduction of epoch and log segment index concepts, along with the proposal of the AdRaft algorithm. This algorithm is designed to optimize the leader election and log replication phases, thereby enhancing throughput and latency of the blockchain network. The AdRaft algorithm incorporates a vote-change mechanism and an allocation idea, contributing to its effectiveness [209]. Abdorrahimi proposed a pioneering approach that integrates blockchain technology and the Raft consensus algorithm to enhance the reliability and efficacy of e-healthcare systems, particularly in the domain of doctors' prescriptions. This approach ensures data security and consistency, thereby contributing to the optimisation of medical decision-making processes [210]. Luo et al. conducted a study that examined the performance of PBFT- and RAFT-based consensus networks in non-ideal wireless environments,

with a particular focus on the impact of millimetre-wave and terahertz signals. The results of this study indicated that RAFT-based consensus networks demonstrated a higher success rate than PBFT-based networks while exhibiting reduced energy consumption [211]. A simple yet accurate analytical model is proposed to analyse the split probability of a distributed network, demonstrating the impact of three parameters, namely network size, packet loss rate, and election timeout period, on the availability [212]. Woos proposes a methodology to plan for changes during formal verification, and successfully applies it to the validation of the Raft consistency protocol. The application of information hiding, custom induction principles, higher-order affine primitives, and structural tactics has been demonstrated to result in a substantial reduction in the redo work due to definition and theorem changes [213]. Li et al. proposed a probabilistic failure model for RAFT consensus, incorporating Markovian properties to calculate reliability in the presence of failures at the nodes and communication links. This model was tested through simulations, which yielded linear relationships between reliability and tolerance gains, on the one hand, and failure rate and threshold of faulty nodes, on the other. Furthermore, the positive correlation of identified faulty nodes on consensus reliability was analysed [214].

In summary, Raft is a simple, efficient, and easily implementable consensus algorithm for distributed systems that require strong consistency and moderate size. The primary benefits of this approach are its comprehensibility and its capacity to withstand failures. Consequently, it is extensively employed in enterprise-level contexts. However, leader bottlenecks, concurrency performance limitations, and dependence on the network have been shown to result in underperformance of algorithms such as Egalitarian Paxos in highly concurrent or extremely dynamic scenarios. Raft has been identified as ideal for scenarios such as private or federated chains, distributed databases, etc., but it may need to be improved or other mechanisms introduced in public chains or large-scale decentralised systems.

6.3. Sharded Consensus

Sharded Consensus is a consensus mechanism in a distributed system or blockchain that improves system scalability and throughput by dividing the network into multiple smaller subsets. This avoids duplicated overhead on communication, storage, and computation for each full node to process transactions or tasks in parallel. Each subset operates in accordance with the consensus protocol while concurrently undertaking partial storage of transactions in an autonomous manner. This approach is intended to alleviate the demands placed upon a solitary global consensus, thereby facilitating the attainment of a throughput that exhibits linear growth, and a network size that is likewise able to scale in a linear fashion [215].

In a recent study, Dang examined the fundamental disparities in failure models between conventional distributed databases and blockchain systems. Utilising Trusted Execution Environments (TEEs), Dang proposed the design of fault-scalable consensus protocols and efficient shard formation protocols. Furthermore, Dang put forward a cross-shard transaction coordination protocol that facilitates the execution of general blockchain workloads. This coordination protocol employs a Byzantine fault-tolerant reference committee to prevent a malicious coordinator [216]. Tao proposes a consensus group selection algorithm based on inter-node transmission latency (CGSTD), with the objective of optimising consensus group selection in a sharded blockchain. The aim is to reduce the overall consensus latency and enhance security. The efficacy of CGSTD is demonstrated through experimental findings that indicate its superior performance in terms of total consensus latency, increase in consensus latency of the sharded blockchain, node storage requirements, and consensus group participation [217]. In order to achieve a minimised remapping of node assignments, and thus reduce the average time complexity of the system while ensuring its security,

Chen combines the Jump Consistent Hash algorithm and the Signature Anchorhash algorithm. The cross-split transaction processing scheme has been designed to address the potential malicious behaviour of both nodes and leaders within the split. The incentive mechanism has been designed to take into account the differences between different splits and node types, thus ensuring greater fairness and efficacy in comparison with traditional mechanisms [218]. Wu et al. proposed a novel KBFT algorithm, which was found to be more reasonable and effective than traditional incentive mechanisms. This was achieved by combining the K-archetypal clustering algorithm and the BLS-based multi-signature Byzantine fault-tolerant algorithm. The results of the study demonstrated that this combination significantly improves the scalability and throughput of large-scale federated chains while reducing communication complexity. In addition, the algorithm designs a simple and efficient credit mechanism and supervision mechanism, which further ensures the security and reliability of the system [219]. Liu et al. significantly enhance the blockchain's scalability and throughput by implementing adaptive slice management, secure and atomic cross-slice transaction processing, efficient slice state synchronization, and dispute resolution mechanism. These innovations substantially improve the performance and security of blockchain sharding [220]. In this study, Kogias investigates the limitations of ByzCoin in open, adversarial networks and proposes a new consensus protocol, MOTOR. This protocol is designed to balance robustness and scalability by using deterministic cryptography that is resistant to DoS attacks, introducing a random spinning leader mechanism, and designing an incentive-compatible reward mechanism. The study addresses the shortcomings of existing solutions [221].

In essence, Sharded Consensus represents a pivotal technological solution to the scalability challenges confronting blockchain, thereby markedly enhancing throughput and efficiency through parallel processing and dynamic sharding. This approach is particularly well suited to scenarios characterised by high transaction volumes, including public and alliance chains. The primary benefits of this approach are its high scalability, low latency, and resource efficiency. However, significant challenges must be addressed, including complexity, security risk, cross-slice communication, and load balancing. Additionally, the implementation process is intricate and the degree of decentralization must be carefully considered. In the context of large-scale, concurrency-intensive distributed systems, slice consensus emerges as the optimal solution. However, it is imperative to optimise the security and consistency mechanisms in conjunction with specific scenarios to ensure optimal performance and reliability.

6.4. DAG-Based Mechanism

In the domain of blockchain technology, consensus algorithms such as POW, POS, and PBFT are widely utilised. These algorithms function by operating on the infrastructure of the chain ledger, which serves as the primary data storage and transaction monitoring system. The chained data structure necessitates the generation of blocks in a serial manner, thereby constraining the scalability of consensus algorithms. The notion of a directed acyclic graph (DAG) was initially proposed as a consensus algorithm by a scholar from the Hebrew University of Israel in 2013. Subsequently, members of the NXT community have advocated for the utilisation of the DAG structure for the purpose of storing blocks, with the objective of facilitating concurrent packing and execution of blocks [222]. At present, public chains such as IOTA, Byteball, and Nano utilise a DAG (directed acyclic graph) as their ledger structure, employing disparate consensus algorithms. In addition, a significant number of scholars have contributed to the development of the DAG consensus, including the Conflux mechanism proposed by Yao Zhizhi's team [223], the SPECTRE mechanism proposed by scholars from the Hebrew University of Jerusalem, Israel [224], and the

Phantom consensus mechanism proposed by this team [225], as well as the Hashgraph proposed by swirlds [226]. The layered DAG blockchain architecture is utilised in VANET in edge computing environments to address security and trust issues. Pervez et al. discuss seven different communities working on blockchain based on directed acyclic graph (DAG) and propose what an optimal world DAG blockchain architecture framework should be [227].

Subsequently, Cao et al. proceeded to analyse the advantages and disadvantages of PoW, PoS, and DAG mechanisms. DAG-based consensus has been shown to exhibit lower transaction fees and resource consumption while also achieving higher transaction throughput. However, the confirmation delay of DAG consensus in real IoT scenarios, when the traffic load varies over time, is significantly affected by the traffic load as well as centrality issues [228]. Raikwar et al. demonstrated that DAG architecture boasts significant advantages in terms of improving scalability and flexibility. However, it is important to note that DAG architecture is also faced with challenges in terms of complexity and security [229]. Furthermore, Wang has highlighted that a DAG blockchain system has considerable potential in improving scalability and performance. However, it is essential to emphasise that the system still requires enhancement in terms of security, consistency, and finality [230]. In addition, Qu has proposed a novel consensus mechanism, TidyBlock, for DAG-based blockchain in the Internet of Things (IoT). The construction of a visible network, the design of transaction collation and block selection algorithms, the facilitation of fast on-chain storage and straightforward data searching, post-processing, and upper tier applications are all achieved through this methodology [231]. The enhancement of security is achieved through the utilisation of a DAG blockchain and the adaptation of parallel operations in VANET, superseding the use of single-threaded chained blockchain. In order to meet specific power consumption requirements in VANET and to consider typical licensed network models, a node reputation TSA mechanism is developed to replace the widely used PoW consensus. Meanwhile, a layered architecture was designed to decentralise major transactions to LCs to improve blockchain efficiency and reduce system overhead [232]. Wu explored the application of a distributed consensus mechanism in wirelessly connected autonomous systems (CASs) and proposed a directed acyclic graph (DAG)-based messaging structure to solve the problem of message loss and unpredictable forwarding delays [233]. Deng proposed a federated blockchain consensus algorithm that is both highly concurrent and scalable. This algorithm is based on segmented directed acyclic graphs (DAGs) and back propagation neural networks (BPNNs) with the aim of improving system throughput and reducing the time complexity of global retrieval. The algorithm's segmented DAG structure performs excellently in terms of throughput and response time while also offering high security and scalability [234]. In this paper, the author puts forward a lightweight and efficient distributed ledger consensus algorithm based on Bayesian inference for distributed ledger graphs (DAGs). This is proposed as a solution to the security and scalability problems in Internet of Things (IoT) systems. The efficiency and resistance to attacks of this algorithm when processing a large number of transactions is also discussed [234]. Zhang and Li proposed a parallel consensus mechanism based on the DAG lattice structure and the PBFT that solves the problems caused by inefficient PBFT consensus and node mobility in the Internet of Vehicles (IoV) [235]. Sasikumar et al. proposed a DAG-based blockchain consensus mechanism applied to the industrial Internet of Things (IIoT) [236]. Li and Huang proposed an efficient DAG-based blockchain architecture for solving the scalability and throughput problems caused by the single-chain structure of traditional blockchain in IoT applications [237]. Ding and Sato conceptualised a decentralised database platform (DAGbase) employing a layered architecture and a DAG-based consensus mechanism to address the security challenges posed by distributed

database platforms when confronted with threats from centralised entities while ensuring operational efficiency and cost-effectiveness [238]. Zhou et al. proposed a permissionless blockchain system, designated as DLattice. This system employs a dual DAG architecture and the DPoS-BA-DAG (PANDA) protocol to achieve efficient data protection and data tokenisation [239].

Furthermore, a DAG-based distributed ledger technology for generating, verifying, and confirming power transactions in smart grids was proposed by Park and Kim. The aim of this technology is to achieve low-latency power transactions in smart grids [240].

Dai et al. proposed GradedDAG, an asynchronous Byzantine fault-tolerant consensus protocol based on the DAG structure, with the objective of reducing the latency of existing DAG protocols [241]. This was followed by LightDAG, which significantly reduces latency by replacing the RBC protocol with lightweight broadcast protocols such as CBC and PBC [242]. Jovanovic's approach involved the use of an unauthenticated DAG, along with the submission of multiple leader blocks in each asynchronous DAG round, resulting in the achievement of both high latency and low latency. MAHI-MAHI, on the other hand, operates under asynchronous conditions, achieving high throughput and low latency. In a geographically distributed environment, MAHI-MAHI is capable of processing 350,000 transactions per second while maintaining a latency of less than two seconds [243]. Danezis et al. designed and evaluated a memory pooling protocol, Narwhal, which is able to tolerate asynchronous networks and maintain high-performance memory pooling in the event of failures. In conjunction with the asynchronous consensus protocol Tusk, which exhibits zero message overhead, the Narwhal protocol achieves 160,000 transactions per second with approximately three seconds of latency [70]. In their study, Cao et al. [71] compared the performance of three blockchain consensus mechanisms: proof of work (PoW), proof of stake (PoS), and directed acyclic graph (DAG). Their findings revealed that PoW and PoS are susceptible to transaction loss under high loads, while DAG demonstrates resilience in this regard.

Subsequently, Liu et al. [72] proposed the Serein algorithm, a novel approach aimed at enhancing the scalability and throughput of blockchain systems. This algorithm employs a functional partitioning of nodes, adopts a pipeline structure, and leverages the DAG structure to optimize system performance. In the research study by Xiang et al., the Jointgraph algorithm was shown to outperform the existing Hashgraph algorithm in terms of throughput and latency. This was achieved by the introduction of a supervisory node and the optimisation of the consensus process [244]. Subsequently, Xiang also proposed the Teegraph consensus algorithm, which combines TEE and DAG techniques to address the issue of efficient data sharing among IoT devices [245].

DAG-based consensus algorithms have been shown to provide high throughput, low latency, and low fees through parallel transaction processing and decentralised architecture, rendering them particularly suitable for IoT, instant payments, and high-performance blockchain scenarios. Their primary benefits are scalability and energy efficiency, which surpass those of traditional mechanisms such as Paxos and Raft. Nevertheless, the inherent complexity of DAGs, in conjunction with their vulnerability to security challenges, weak consistency, and network dependency, imposes significant limitations on their application in scenarios characterised by strong consistency or high conflict. In comparison with Sharded Consensus, DAG exhibits greater natural parallelism, yet it possesses weaker load balancing capabilities. The DAG algorithm continues to evolve rapidly, and future optimisations such as IOTA's Coordicide or Hedera's Decentralised Governance are anticipated to further augment its capacity for public chains and particular distributed systems, contingent upon trade-offs in security and maturity.

Chapter Summary

Flexible-scaling consensus mechanisms are designed with adaptability at their core, with the objective of maintaining performance and stability even as network size and complexity increase. This category comprises a variety of approaches, ranging from modular designs such as EPaxos and Raft to advanced scaling solutions including Sharded Consensus and DAG-based mechanisms.

The performance metrics of the consensus mechanism for flexible scaling are summarised in Table 12.

Table 12. Flexible-extension consensus mechanism performance indicators.

	Throughput	TPS	CT	Security	FT	CoA	Energy Consumption	CPT	HR	Scalability	NS	CC
Egalitarian Paxos [246–248]	Moderate	3 k+	100–500 ms	High	2f + 1 (CFT)	Moderate	Low	Low	Standard server	High	High	O(n·logn)
Raft [202,207,249]	Moderate	1 k+	500 ms	Moderate	2f + 1 (CFT)	Moderate	Low	Low	LH I/O	High	-	O(n)
Sharded Consensus [250–253]	Very High	10 K+	-	Low	-	Low	Low	Low	Low	Very High	10 k+	O(n·k)
DAG-based Consensus [63,254–258]	Very High	4 k+	1–2 s	Moderate	2f + 1	10^{-4} kWh	Very Low	Low	Low	Very High	1000 k+	O(1)–O(n)

CT: Confirmation time. **FT:** Fault tolerance. **CPT:** Consumption per transaction. **NS:** Node scalability. **LH I/O:** Low-latency, high-throughput disk I/O. **CoA:** Cost of attack. **HR:** Hardware requirements. **CC:** Communication complexity.

The primary function of these mechanisms is to break down the monolithic structure of traditional blockchains in order to achieve scalability. The strategies employed by these systems include leaderless architectures (e.g., EPaxos), simplified leader election (e.g., Raft), network partitioning (e.g., Sharding), and parallel transaction processing (e.g., DAG). The underlying philosophy of this approach is to distribute the workload, minimise global consensus overhead, and allow for concurrent operations. This is intended to overcome the inherent scalability limitations of single-chain, serial-processing models.

While designed for scalability, these mechanisms often involve complex trade-offs with other performance attributes:

Scalability vs. Consistency/Complexity: EPaxos offers high availability and throughput in low-conflict scenarios but faces limitations in high-conflict environments due to its inherent complexity. Raft prioritizes simplicity and strong consistency within a leader–follower model, making it suitable for moderate-sized systems but less adept at handling large-scale public networks where leader bottlenecks and high concurrency can degrade performance.

Scalability vs. Security/Decentralization: Sharding dramatically improves scalability by partitioning the network, but it introduces new security challenges, such as cross-shard attack vectors and the "single-shard takeover" problem. Managing shard state consistency and ensuring fair load balancing also add significant complexity. DAGs offer high parallelism and low latency, but their inherent "weak consistency" can pose challenges for finality guarantees and security compared to linear blockchains. The trust models and decentralization levels in these flexible-scaling mechanisms need careful consideration, as the pursuit of efficiency can sometimes lead to a reduction in the number of participants or increased complexity in maintaining a truly decentralized state.

The core design and application scenario refinement analysis for flexible scaling are summarised in Table 13.

Paxos and Raft are robust for environments requiring strong consistency within a contained number of nodes, making them ideal for enterprise-level applications where throughput needs to be high but decentralization is not the absolute top priority. Their

relative simplicity (Raft) and leaderless efficiency (EPaxos) position them well for private blockchain deployments.

Table 13. Flexible-scaling consensus mechanism core design and application scenario refinement analysis.

Consensus Mechanisms	Core Design and Trade-Offs	Applicable Scenarios
Egalitarian Paxos	Leaderless architecture for high availability and low latency in WAN. Complexity and conflict sensitivity are limitations.	Private/Enterprise systems, low-conflict environments
Raft	Simplicity and strong consistency via leader. Leader bottleneck and concurrency limits in high-scale scenarios.	Private/Federated chains, distributed databases
Sharded Consensus	Network partitioning for parallel processing. Introduces security risks (cross-shard attacks) and management complexity.	Public chains (Ethereum 2.0), high-volume DApps
DAG-based Consensus	Parallel transaction processing for high throughput and low fees. Challenges in consistency, finality, and potential centralization.	IoT, instant payments, high-performance blockchains

Sharded Consensus and DAG-based Consensus are at the forefront of tackling the scalability challenge for public blockchains. They achieve significantly higher transaction volumes by processing transactions in parallel. However, Sharded Consensus is complicated to manage, while DAG-based Consensus has challenges in consistency, and both introduce new security and load balancing complexities that are actively being researched.

- Future Trends:

The future of flexible-scaling mechanisms lies in hybrid approaches that combine the strengths of different techniques. For example, integrating sharding with BFT-style finality gadgets or using DAGs as a transactional layer over a more robust base consensus could address current limitations. Further research will focus on dynamic sharding, cross-shard communication optimization, and novel incentive mechanisms to ensure both performance and security in increasingly large and complex decentralized networks.

7. Discussion and Future Directions

In order to discuss the regulatory and ethical implications of consensus mechanisms, this paper focuses on privacy, governance, and fairness. Regulatory and ethical considerations, and future trends are the two main topics of this chapter.

7.1. Regulatoryand Ethical Considerations

The evolution of blockchain consensus mechanisms is not just a technical one; it has significant regulatory and ethical implications, especially concerning privacy, governance, and fairness [259]. While many mechanisms aim to solve the blockchain trilemma, they often introduce new trade-offs that can affect these critical areas.

7.1.1. Privacy

The transparency of public blockchains, while a security feature, can compromise user privacy. Transactions on protocols like Ripple and Bitcoin are publicly announced, exposing a user's transaction behaviour (e.g., time and amount). This can lead to de-anonymization attacks, as seen in research on the Ripple system.

Some consensus mechanisms, especially those in private or consortium chains like PoA, inherently offer more privacy by limiting participation to known, trusted entities. However, this comes at the cost of decentralization. Future solutions will likely involve integrating privacy-enhancing technologies like zero-knowledge proofs (ZKPs) directly into the consensus layer to allow for private transactions on public chains without sacrificing

network security or decentralization [111,260,261]. For instance, some current research focuses on combining BFT-style protocols with ZKPs for privacy-sensitive applications.

7.1.2. Governance and Centralization

Many high-throughput and low-energy mechanisms, in an effort to scale, compromise on decentralization. The DPoS mechanism, for example, improves efficiency by electing a small number of representatives, but this can lead to a concentration of power. The Hedera Hashgraph protocol, while fast and efficient, has shown a tendency towards centralization. The Ripple Protocol's reliance on a limited number of pre-selected validators makes it highly efficient but fundamentally centralized, limiting its applicability to permissioned environments. This centralization risk can lead to governance issues where a small group can influence network decisions or even manipulate the system for their benefit.

Consensus mechanisms are attempting to address this by introducing more sophisticated governance models. Research on PoD, for example, aims to link accounting rights to user contributions rather than just token holdings, promoting a broader evaluation of participants and reducing over-reliance on a few wealthy nodes. Similarly, variants of PBFT, such as DBFT, are exploring ways to democratize the voting process and broaden participation. The challenge is to implement these systems transparently and fairly to avoid new forms of concentration or unfairness.

7.1.3. Fairness and Operational Complexity

The search for efficiency can also lead to issues of fairness and increased operational complexity. In some systems, like a modified version of Tendermint, the original reward mechanism was found to be unfair, and it required minor adjustments to ensure equitable rewards for validators [262]. In proof of stake, while it lowers the barrier to entry for users via delegation, the design of the reward system can still lead to wealth inequality in the long run. Meanwhile, protocols like PoUW, while aiming to reduce energy waste, introduce significant technical complexity and validation costs that can be difficult to manage in practice.

The trend is towards a hybrid approach. For example, Casper combines proof of stake with BFT to achieve a better balance between energy efficiency, security, and fairness, although it is not without its own complexities. The implementation of random selection mechanisms, as seen in Roll-DPoS and other variants, helps to reduce the possibility of collusion and improves resistance to attacks, thereby enhancing both security and fairness. The ongoing challenge is to create systems that are both highly performant and governed by transparent, auditable, and truly decentralized rules that reward honest behaviour and penalize malicious actions without becoming overly complex or introducing new points of failure.

In conclusion, addressing these regulatory and ethical challenges requires moving beyond the traditional performance metrics of throughput and latency.

7.2. Future Directions

The advent of self-executing contracts of consensus mechanisms, implemented through lines of code, has precipitated a paradigm shift in the execution of traditional contracts. These mechanisms have been shown to automate and embed contracts within the blockchain network, thereby redefining the landscape of contractual execution. They ensure the transparency and immutability of contract rules enforced by blockchain network participants, and they allow building decentralised applications (DApps) on the blockchain network, facilitating applications in several areas such as [263]. It is the authors' opinion that the trend of future development is roughly twofold. Firstly, on the technical side, there is an improvement in performance, including privacy, decentralisation, security, through-

put, scalability, and energy efficiency, as previously mentioned. Secondly, on the application side, there is an integration with various advanced technologies, such as web3 [78], supply chain, IoT [264], AI [12,265], healthcare, crowdfunding funds, and quantum technology, etc.

Firstly, from a technical standpoint, it is probable that blockchain will become progressively more efficient. Platforms such as Ethereum are already transitioning from "proof of work" to "proof of stake" in order to reduce energy consumption, and it is anticipated that further innovations, including sharding and Layer 2 solutions (e.g., Rollups), will emerge in the future to enhance transaction efficiency. Rollups have been employed to enhance transaction speed and scalability. It is conceivable that a more widespread consensus may be reached with regard to the implementation of mechanisms that would facilitate the acceleration of the blockchain and the conservation of resources.

In terms of application, the use of blockchain is expected to extend to a greater number of industries. It is evident that the field of finance continues to dominate, with decentralised finance (DeFi) emerging as a notable development. This shift has prompted the entry of traditional financial institutions into the domain [266]. Recent initiatives by prominent figures, such as Musk's Dogecoin and Trump's Trumpcoin, serve as illustrative examples. Notably, Trumpcoin does not adhere to the conventional virtual currency model reliant on mining. Nevertheless, it signifies a recognition of the ongoing expansion of the virtual economy, driven by the innovative financial instruments offered by virtual currencies. It is anticipated that the domains of supply chain management, digital identity verification, and copyright protection will witness a significant escalation in the implementation of blockchain technology. Moreover, there is a strong possibility that even domains such as voting systems and medical data management will be impacted by this technological advancement. In addition, the combination of non-fungible tokens (NFTs) and the metaverse has the potential to facilitate increased asset trading within the virtual realm [267–269].

Regulation is a pivotal variable in this context. In the coming years, governments may introduce clearer rules. The adoption of blockchain technology by various entities, including public services, may be a subject of interest. Conversely, the implementation of more stringent regulations, such as those imposed on cryptocurrencies, is a potential outcome. This will have a direct impact on the rate of globalization of blockchain. However, this paradigm is not without its challenges, including privacy issues (the balancing act of ensuring both the openness and transparency of on-chain data and the protection of privacy), cross-chain interoperability (the effective collaboration across different blockchains), and the potential threat of quantum computing (the potential compromise of current encryption algorithms) [270]. Nevertheless, it is plausible that these may also be the driving force behind the iteration of the technology.

8. Conclusions

The present paper provides a comprehensive and systematic analysis of the performance of consensus mechanisms in blockchain technology. The aim of the paper is to provide a clear framework for researchers and practitioners to understand the performance characteristics, strengths, and weaknesses of existing consensus mechanisms and their future development potential. By categorising and evaluating the performance of consensus mechanisms, the paper reveals the applicability of blockchain technology in different domains and the current challenges it faces.

It is evident that consensus mechanisms can be categorised into four distinct performance-oriented categories: high throughput, high security, low energy consumption, and flexible scalability. Each category encompasses a range of both established and emerging consensus mechanisms. The paper employs key metrics, including throughput, confirmation time, security, energy consumption, and scalability, to evaluate the

performance of these mechanisms. By analysing these metrics, the paper reveals the performance trade-offs between different consensus mechanisms. To illustrate this point, we may consider the relative strengths and weaknesses of PoW, PoS, and PoA. PoW offers high levels of security but also results in extremely high energy consumption. By contrast, PoS and PoA excel in terms of energy consumption but may compromise on the degree of decentralisation.

The paper also observes that blockchain consensus mechanisms are evolving from PoW to more environmentally friendly and efficient alternatives such as PoS and DPoS, a trend driven by a combination of technological and environmental factors. This transition is further substantiated by market data from 2018 to 2025, which indicates a gradual substitution of PoW's predominance by PoS and alternative mechanisms.

Furthermore, the paper emphasises the application-specific nature of consensus mechanisms. For instance, PoA is well suited for IoT scenarios, PBFT is appropriate for permissioned blockchains, and DAG consensus demonstrates optimal performance in high-throughput public blockchains. Recent technological advances, including HL-DPoS, HotStuff, and Casper, have achieved substantial advancements in throughput, security, and energy efficiency. However, these innovations continue to confront challenges related to scalability and decentralization. The paper also discusses in detail the limitations and challenges of each type of consensus mechanism, such as PoH and Avalanche in high-throughput mechanisms that may face security risks or dependence on network conditions, and DPoS and Tendermint that may lead to decentralization due to dependence on a few nodes. The high energy consumption of PoW in high-security mechanisms is not sustainable, PBFT is less scalable when the number of nodes increases, and FBA and DBFT are highly dependent on the trust structure. In the realm of low-energy consumption mechanisms, PoET necessitates dedicated hardware support. The PoC is susceptible to storage attacks, and the PoB encounter challenges related to resource wastage and complexity. In the context of flexible scalability mechanisms, slice consensus and DAG consensus have been identified as exhibiting challenges pertaining to security, complexity, and load balancing. Furthermore, DAG has been observed to potentially encounter consistency issues in scenarios characterised by high levels of conflict.

At this juncture, an evaluation of the prevailing consensus mechanisms reveals that no individual protocol can fully satisfy all business requirements. Prior to the selection of the DLT models and consensus protocols involved, it is imperative to meticulously consider their business needs and deployment environments in order to comprehend their performance requirements for consensus mechanisms. In environments where there is a lack of trust and where there is a perception of hostility, it is essential that consensus is complex and that it includes incentives and severe penalties for participant nodes. This is in order to ensure the integrity of the network and to prevent fraudulent nodes from appearing on the network. Consequently, the security of public DLTs is achieved at the expense of speed and scalability. Conversely, within private environments characterised by the presence of trusted participating nodes, consensus protocols can be straightforward and do not necessitate incentives. This is due to the fact that participating organisations possess a commercial interest in protecting and securing the network, enabling them to prioritise aspects such as speed and scalability.

It is anticipated that blockchain consensus mechanisms will continue to drive the digital ecosystem as the technology continues to evolve. The development of next-generation consensus mechanisms is of paramount importance in order to facilitate a more decentralised, efficient, and sustainable blockchain system. Such mechanisms will be achieved by optimising existing mechanisms, developing hybrid solutions, integrating with emerging technologies, and addressing key challenges. Standardisation efforts and advances in

cross-chain interoperability will also further promote the popularisation and adoption of blockchain technology.

In summary, this paper provides a systematic summary of the current research on blockchain consensus mechanisms, and it also indicates future development trends. Through continuous innovation and exploration, blockchain technology will play a greater role in the fields of finance, healthcare, the Internet of Things, Web3, and so forth, and will promote profound changes in the digital economy and social governance.

Author Contributions: Conceptualization, Z.S.; methodology, Z.S. and X.-B.C.; software, Z.S.; validation, Z.S., Q.Q. and X.-B.C.; formal analysis, Q.Q.; investigation, Z.S.; resources, Z.S.; data curation, Z.S.; writing—original draft preparation, Z.S.; writing—review and editing, Z.S.; visualization, Z.S.; supervision, Q.Q. and X.-B.C.; project administration, X.-B.C.; funding acquisition, X.-B.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China under Grant 71571091 and Grant 71771112.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: We wish to thank all data providers. We also wish to thank all colleagues, reviewers, and editors who provided valuable suggestions.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Diffie, W. New Directions in Cryptography. *IEEE Trans. Inform. Theory* **1976**, *22*, 472–492. [CrossRef]
2. Shostak, R.; Pease, M.; Lamport, L. The Byzantine Generals Problem. *ACM Trans. Program. Lang. Syst.* **1982**, *4*, 382–401. [CrossRef]
3. Merkle, W.; Stephan, F. Trees and Learning. In Proceedings of the 9th Annual Conference on Computational Learning Theory, Desenzano del Garda, Italy, 28 June–1 July 1996; pp. 270–279.
4. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. 2008. Available online: https://assets.pubpub.org/d8wct41f/31611263538139.pdf (accessed on 3 August 2025).
5. Motepalli, S.; Jacobsen, H.A. Decentralizing Permissioned Blockchain with Delay Towers. *arXiv* **2022**, arXiv:2203.09714. [CrossRef]
6. Wang, W.; Hoang, D.T.; Hu, P.; Xiong, Z.; Wang, P. A Survey on Consensus Mechanisms and Mining Strategy Management in Blockchain Networks. *IEEE Access* **2019**, *7*, 22328–22370. [CrossRef]
7. Narayanan, A.; Bonneau, J.; Felten, E.; Miller, A.; Goldfeder, S. *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*; Princeton University Press: Princeton, NJ, USA, 2016; pp. 1–350.
8. Etherscan. Available online: https://etherscan.io/ (accessed on 3 August 2025).
9. Comprehensive Guide to Companies Involved in Blockchain & Energy. Available online: https://www.solarplaza.com/ (accessed on 3 August 2025).
10. Islam, S.; Islam, M.J.; Hossain, M.; Noor, S.; Kwak, K.-S.; Islam, S.M.R. A Survey on Consensus Algorithms in Blockchain-Based Applications: Architecture, Taxonomy, and Operational Issues. *IEEE Access* **2023**, *11*, 39066–39082. [CrossRef]
11. Hazari, S.S.; Mahmoud, Q.H. Comparative Evaluation of Consensus Mechanisms in Cryptocurrencies. *Internet Technol. Lett.* **2019**, *2*, e100. [CrossRef]
12. Bamakan, S.M.H.; Motavali, A.; Bondarti, A.B. A Survey of Blockchain Consensus Algorithms Performance Evaluation Criteria. *Expert Syst. Appl.* **2020**, *154*, 113385. [CrossRef]
13. Paykari, N.; Lyons, D.M.; Rahouti, M. Enhancing Visual Homing in Robotics: A Study on Blockchain Integration and Consensus Algorithms. *Distrib. Ledger Technol.* **2025**, *4*, 6. [CrossRef]
14. Liu, X.; Yu, W. A Review of Research on Blockchain Consensus Mechanisms and Algorithms. In Proceedings of the 9th International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS), Okinawa, Japan, 21–23 November 2024; Volume 9, pp. 1–10.
15. Ismail, L.; Materwala, H. A Review of Blockchain Architecture and Consensus Protocols: Use Cases, Challenges, and Solutions. *Symmetry* **2019**, *11*, 1198. [CrossRef]

16. Ferdous, M.S.; Chowdhury, M.J.M.; Hoque, M.A.; Colman, A. Blockchain Consensus Algorithms: A Survey. *arXiv* **2020**, arXiv:2001.07091. [CrossRef]
17. Shrimali, B.; Patel, H.B. Blockchain State-of-the-Art: Architecture, Use Cases, Consensus, Challenges and Opportunities. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 6793–6807. [CrossRef]
18. Zhou, S.; Li, K.; Xiao, L.; Cai, J.; Liang, W.; Castiglione, A. A Systematic Review of Consensus Mechanisms in Blockchain. *Mathematics* **2023**, *11*, 2248. [CrossRef]
19. Chand, S.; Liu, Y.A.; Stoller, S.D. Formal Verification of Multi-Paxos for Distributed Consensus. In Proceedings of the FM 2016: Formal Methods, Limassol, Cyprus, 9–11 November 2016; Volume 9995, pp. 1–16.
20. Liang, Z.; Jabrayilov, V.; Charapko, A.; Aghayev, A. MultiPaxos Made Complete. *arXiv* **2024**, arXiv:2405.11183. [CrossRef]
21. Du, H.; Hilaire, D.J.S. Multi-Paxos: An Implementation and Evaluation. 2009. Available online: https://dada.cs.washington.edu/research/tr/2009/09/UW-CSE-09-09-02.PDF (accessed on 3 August 2025).
22. Lin, W.; Jiang, H.; Zhao, N.; Zhang, J. An Optimized Multi-Paxos Protocol with Centralized Failover Mechanism for Cloud Storage Applications. In Proceedings of the Collaborative Computing: Networking, Applications and Worksharing, London, UK, 19–22 August 2019; Volume 14, pp. 610–625.
23. Luo, Y.H.; Chen, Y.Q.; Chen, Q.; Liang, Q.L. A New Election Algorithm for DPos Consensus Mechanism in Blockchain. In Proceedings of the 7th International Conference on Digital Home (ICDH), Aizuwakamatsu, Japan, 29–31 March 2019; pp. 116–120.
24. Saad, S.M.S.; Radzi, R.Z.R. Comparative Review of the Blockchain Consensus Algorithm Between Proof of Stake (POS) and Delegated Proof of Stake (DPOS). *Int. J. Innov. Comput.* **2020**, *10*, 2. [CrossRef]
25. Larimer, D. Delegated Proof-of-Stake (DPOS). Bitshare Whitepaper. 2014, *unpublished*.
26. Li, Y.; Xia, C.; Li, C.; Zhao, Y.; Chen, C.; Wang, T. HL-DPoS: An Enhanced Anti-Long-Range Attack DPoS Algorithm. *Comput. Netw.* **2024**, *249*, 110473. [CrossRef]
27. Wei, Y.; Xu, Q.; Peng, H. An Enhanced Consensus Algorithm for Blockchain. *Sci. Rep.* **2024**, *14*, 17701. [CrossRef]
28. Bachani, V.; Bhattacharjya, A. Preferential Delegated Proof of Stake (PDPoS)—Modified DPoS with Two Layers towards Scalability and Higher TPS. *Symmetry* **2023**, *15*, 4. [CrossRef]
29. Fan, X.; Chai, Q. Roll-DPoS: A Randomized Delegated Proof of Stake Scheme for Scalable Blockchain-Based Internet of Things Systems. In Proceedings of the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous '18), New York, NY, USA, 5–7 November 2018; pp. 482–484.
30. Sun, Y.; Yan, B.; Yao, Y.; Yu, J. DT-DPoS: A Delegated Proof of Stake Consensus Algorithm with Dynamic Trust. *Procedia Comput. Sci.* **2021**, *187*, 371–376. [CrossRef]
31. Lin, H.; Du, J. SP-DEWOA: An Evolutionary Distributed Witness Node Election Method for Delegated Proof of Stake. *IEEE Internet Things J.* **2025**, *12*, 3003–3016. [CrossRef]
32. Tan, P.; Wan, L.; He, P.; Li, X. Blockchain Architecture for Lightweight Storage. *Appl. Sci.* **2025**, *15*, 1446. [CrossRef]
33. Xu, G.; Liu, Y.; Khan, P.W. Improvement of the DPoS Consensus Mechanism in Blockchain Based on Vague Sets. *IEEE Trans. Ind. Inform.* **2019**, *16*, 4252–4259. [CrossRef]
34. Kwon, J. Tendermint: Consensus Without Mining. 2014. Available online: https://www.weusecoins.com/assets/pdf/library/Tendermint%20Consensus%20without%20Mining.pdf (accessed on 3 August 2025).
35. Lagaillardie, N.; Djari, M.A.; Gürcan, Ö. A Computational Study on Fairness of the Tendermint Blockchain Protocol. *Information* **2019**, *10*, 378. [CrossRef]
36. Amoussou-Guenou, Y.; Del Pozzo, A.; Potop-Butucaru, M.; Tucci-Piergiovanni, S. Correctness and Fairness of Tendermint-Core Blockchains. *arXiv* **2018**, arXiv:1805.08429. [CrossRef]
37. Buchman, E. Tendermint: Byzantine Fault Tolerance in the Age of Blockchains. Ph.D. Thesis, University of Guelph, Guelph, ON, Canada, 2016.
38. Buchman, E.; Guerraoui, R.; Komatovic, J.; Milosevic, Z.; Seredinschi, D.-A.; Widder, J. Revisiting Tendermint: Design Tradeoffs, Accountability, and Practical Use. In Proceedings of the 52nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks-Supplemental Volume (DSN-S), Baltimore, MD, USA, 27–30 June 2022; pp. 11–14.
39. Karamachoski, J.; Gavrilovska, L. Extended Performance Evaluation of the Tendermint Protocol. In Proceedings of the ETAI 2021, San Diego, CA, USA, 1–5 August 2021; pp. 1–5.
40. Abraham, I.; Gueta, G.G.; Malkhi, D.; Reiter, M.K.; Yin, M.F. HotStuff: BFT Consensus in the Lens of Blockchain. *arXiv* **2019**, arXiv:1803.05069. [CrossRef]
41. Garay, J.A.; Kiayias, A.; Leonardos, N. The Bitcoin Backbone Protocol: Analysis and Applications. *J. ACM* **2024**, *71*, 1–49. [CrossRef]
42. Force-Locking Attack on Sync Hotstuff. Available online: https://eprint.iacr.org/2019/ (accessed on 3 August 2025).
43. Abraham, I.; Malkhi, D.; Nayak, K.; Ren, L.; Yin, M. Sync Hotstuff: Simple and Practical Synchronous State Machine Replication. In Proceedings of the 2020 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 18–21 May 2020; pp. 106–118.

44. Gao, M.; Wang, Z.; Lu, G. Enhancing Consensus Security and Privacy with Multichain Ring Signatures Based on HotStuff. *Electronics* **2023**, *12*, 4632. [CrossRef]
45. Jalalzai, M.M.; Niu, J.; Feng, C.; Gai, F. Fast-Hotstuff: A Fast and Robust BFT Protocol for Blockchains. *IEEE Trans. Dependable Secur. Comput.* **2023**, *21*, 2478–2493. [CrossRef]
46. Kang, D.; Gupta, S.; Malkhi, D.; Sadoghi, M. HotStuff-1: Linear Consensus with One-Phase Speculation. *Proc. ACM Manag.* **2025**, *3*, 1–29. [CrossRef]
47. Wang, R.; Yuan, M.; Wang, Z.; Li, Y. Improved Fast-Response Consensus Algorithm Based on HotStuff. *Sensors* **2024**, *24*, 5417. [CrossRef]
48. Niu, J.; Wang, M.; Gai, F.; Jalalzai, M.M.; Feng, C.; Zhang, Y. Chained HotStuff Under Performance Attack. *IEEE Trans. Dependable Secur. Comput.* **2025**, *22*, 3737–3750. [CrossRef]
49. Solana. Available online: https://solana.com/ (accessed on 23 March 2024).
50. Yakovenko, A. Solana: A New Architecture for a High Performance Blockchain v0.8.13. Available online: https://coincode-live.github.io/static/whitepaper/source001/10608577.pdf (accessed on 3 August 2025).
51. Amores-Sesar, I.; Cachin, C.; Tedeschi, E. When Is Spring Coming? A Security Analysis of Avalanche Consensus. *arXiv* **2022**, arXiv:2210.03423. [CrossRef]
52. Amores-Sesar, I.; Cachin, C.; Schneider, P. An Analysis of Avalanche Consensus. In Proceedings of the International Colloquium on Structural Information and Communication Complexity, Vietri sul Mare, Italy, 27–29 May 2024; pp. 27–44.
53. Snowflake to Avalanche: A Novel Metastable Consensus Protocol Family for Cryptocurrencies. Available online: https://avalabs.org/whitepaper (accessed on 3 August 2025).
54. Rocket, T.; Yin, M.; Sekniqi, K.; van Renesse, R.; Sirer, E.G. Scalable and Probabilistic Leaderless BFT Consensus Through Metastability. *arXiv* **2019**, arXiv:1906.08936.
55. Ketchum, A.; Williams, M. On Pseudo-Profound Bullshit in the Avalanche Whitepaper. 2019. Available online: https://chan-relay.snowblossom.org/channel/85ywfwa2quwkcd3d765ljcg5f5exqg8t7fjfzs9a/reviews/2019-11-11-On%20pseudo-profound%20bullshit%20in%20the%20Avalanche%20whitepaper.pdf (accessed on 3 August 2025).
56. Kniep, Q.; Laval, M.; Sliwinski, J.; Wattenhofer, R. Quantifying Liveness and Safety of Avalanche's Snowball. In Proceedings of the European Symposium on Research in Computer Security, Toulouse, France, 22–26 September 2025; pp. 260–275.
57. Ullah, S.; Ullah, Z.; Waqas, A. Efficiency and Reliability of Avalanche Consensus Protocol in Vehicular Communication Networks. In Proceedings of the 19th Conference on Computer Science and Intelligence Systems (FedCSIS), Belgrade, Serbia, 8–11 September 2024; pp. 1–6.
58. Doddipatla, L. Avalanche: A Secure Peer-to-Peer Payment System Using Snowball Consensus Protocols. *TechRxiv* **2025**, *20*. [CrossRef]
59. Baird, L.; Harmon, M.; Madsen, P. Hedera: A Public Hashgraph Network & Governing Council. *White Pap.* **2019**, *1*, 9–10.
60. Alahmad, M.; Alshaikhli, I.; Alkandari, A.; Alshehab, A.; Islam, M.; Alnasheet, M. Influence of Hedera Hashgraph over Blockchain. *J. Eng. Sci. Technol.* **2022**, *17*, 3475–3488.
61. Amherd, L.; Li, S.N.; Tessone, C.J. Centralised or Decentralised? Data Analysis of Transaction Network of Hedera Hashgraph. *arXiv* **2023**, arXiv:2311.06865. [CrossRef]
62. Roh, E.; Choi, J.; Park, Y.; Seo, S.W. Hashgraph-Based Model Parameter Management for Reliable and Secure Deep Learning. In Proceedings of the 2025 IEEE International Conference on Consumer Electronics (ICCE), Osaka, Japan, 23–26 September 2025; pp. 1–5.
63. Schwartz, D.; Youngs, N.; Britto, A. The Ripple Protocol Consensus Algorithm. *Ripple Labs Inc. White Pap.* **2014**, *5*, 151.
64. Ripple. Available online: https://ripple.com/ (accessed on 6 April 2025).
65. Armknecht, F.; Karame, G.O.; Mandal, A.; Youssef, F.; Zenner, E. Ripple: Overview and Outlook. In Proceedings of the Trust and Trustworthy Computing: 8th International Conference (TRUST 2015), Heraklion, Greece, 24–26 August 2015; pp. 163–180.
66. Ripple, T.P. Ripple Protocol Consensus Algorithm Review. *Self-Published* 2015. *unpublished*.
67. Di Luzio, A.; Mei, A.; Stefa, J. Consensus Robustness and Transaction De-Anonymization in the Ripple Currency Exchange System. In Proceedings of the 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS), Atlanta, GA, USA, 5–8 June 2017; pp. 140–150.
68. Baseera, A.; Alsadhan, A.A. Enhancing Blockchain Security Using Ripple Consensus Algorithm. *Comput. Mater. Contin.* **2022**, *73*, 3. [CrossRef]
69. Li, L. Using the Ripple Consensus Algorithm to Achieve Transparent Cross-Border Payments in E-Commerce. In Proceedings of the 6th International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM 2024), Frankfurt, Germany, 20–21 October 2024; Volume 2024; pp. 295–299.
70. Danezis, G.; Kokoris-Kogias, L.; Sonnino, A.; Spiegelman, A. Narwhal and Tusk: A DAG-Based Mempool and Efficient BFT Consensus. In Proceedings of the Seventeenth European Conference on Computer Systems, Rennes, France, 5–8 April 2022; pp. 34–50.

71. Cao, B.; Zhang, Z.; Feng, D.; Zhang, S.; Zhang, L.; Peng, M.; Li, Y. Performance Analysis and Comparison of PoW, PoS and DAG Based Blockchains. *Digit. Commun. Netw.* **2020**, *6*, 480–485. [CrossRef]
72. Liu, Y.; Chen, J.; Zhang, M.; Shi, S.; Wang, F. Serein: A Parallel Pipeline-Based DAG Schema for Consensus in Blockchain. *IET Blockchain* **2024**, *4*, 681–690. [CrossRef]
73. What Is Delegated Proof of Stake (DPoS)? 2025. Available online: https://www.osl.com/hk-en/academy/article/what-is-delegated-proof-of-stake-dpos (accessed on 3 August 2025).
74. Chen, Y.; Liu, F. Research on Improvement of DPoS Consensus Mechanism in Collaborative Governance of Network Public Opinion. *Peer-to-Peer Netw. Appl.* **2022**, *15*, 1849–1861. [CrossRef]
75. Li, C.; Xu, R.; Duan, L. Characterizing Coin-Based Voting Governance in DPoS Blockchains. In Proceedings of the International AAAI Conference on Web and Social Media, Limassol, Cyprus, 5–8 June 2023; Volume 17, pp. 1148–1152.
76. Cason, D.; Fynn, E.; Milosevic, N.; Milosevic, Z.; Buchman, E.; Pedone, F. The Design, Architecture and Performance of the Tendermint Blockchain Network. In Proceedings of the 2021 40th International Symposium on Reliable Distributed Systems (SRDS), Chicago, IL, USA, 20–23 September 2021; pp. 23–33.
77. Tendermint. 2025. Available online: https://github.com/tendermint/tendermint/wiki/introduction (accessed on 3 August 2025).
78. Li, C.; Xu, R.; Palanisamy, B.; Duan, L.; Shen, M.; Liu, J.; Wang, W. Blockchain Takeovers in Web 3.0: An Empirical Study on the TRON-Steem Incident. *ACM Trans. Web.* **2025**, *19*, 1–23. [CrossRef]
79. Shahsavari, Y.; Zhang, K.; Talhi, C. Performance Modeling and Analysis of HotStuff for Blockchain Consensus. In Proceedings of the 2022 Fourth International Conference on Blockchain Computing and Applications (BCCA), San Antonio, TX, USA, 5–7 September 2022; pp. 135–142.
80. Zhao, S.; Wu, Y.; Wang, Z. HotStuff-2 vs. HotStuff: The Difference and Advantage. *arXiv* **2024**, arXiv:2403.18300.
81. Everything You Need to Know About Avalanche (AVAX): Buying, Storing, and Using. 2025. Available online: https://paxful.com/university/zh-hans/what-is-avax-avalanche-crypto (accessed on 3 August 2025).
82. Ethereum vs. Avalanche: 2025 Comparison. 2025. Available online: https://tokentax.co/blog/ethereum-vs-avalanche (accessed on 3 August 2025).
83. Avalanche Consensus—Does It Perform as Promised. 2025. Available online: https://www.researchgate.net/publication/389598083 (accessed on 3 August 2025).
84. Hedera. 2025. Available online: https://hedera.com (accessed on 3 August 2025).
85. Kohli, V.; Chakravarty, S.; Chamola, V.; Sangwan, K.S.; Zeadally, S. An Analysis of Energy Consumption and Carbon Footprints of Cryptocurrencies and Possible Solutions. *Digit. Commun. Netw.* **2023**, *9*, 79–89. [CrossRef]
86. Han, R.; Shapiro, G.; Gramoli, V.; Xu, X. On the Performance of Distributed Ledgers for Internet of Things. *Internet Things* **2020**, *10*, 100087. [CrossRef]
87. Tibuleac, S.; Filer, M.; Grindstaff, S.; Atlas, D. Design and Optimization of Multi-Haul DWDM Networks. In Proceedings of the SPIE 6354, Network Architectures, Management, and Applications IV, Gwangju, Republic of Korea, 5–7 September 2006; pp. 529–538.
88. Tumas, V.; Rivera, S.; Magoni, D.; State, R. Topology Analysis of the XRP Ledger. In Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing, Tallinn, Estonia, 27–31 March 2023; pp. 1277–1284.
89. Amores-Sesar, I.; Cachin, C.; Mićić, J. Security Analysis of Ripple Consensus. *arXiv* **2020**, arXiv:2011.14816. [CrossRef]
90. Dwork, C.; Naor, M. Pricing via Processing-or-Combatting Junk Mail. In Proceedings of the Lecture Notes in Computer Science, Stockholm, Sweden, 9–14 August 1993; Volume 740, pp. 1–16.
91. Jakobsson, M.; Juels, A. Proofs of Work and Bread Pudding Protocols. In Proceedings of the Secure Information Networks: Communications and Multimedia Security, Leuven, Belgium, 20–21 September 1999; pp. 258–272.
92. Rani, P.; Bhambay, R. A Comparative Survey of Consensus Algorithms Based on Proof of Work. In *Emerging Technologies in Data Mining and Information Security: Proceedings of IEMIS 2022, Volume 1*; Springer Nature: Singapore, 2022; pp. 261–268.
93. Lashkari, B.; Musilek, P. A Comprehensive Review of Blockchain Consensus Mechanisms. *IEEE Access* **2021**, *9*, 43620–43652. [CrossRef]
94. Verium the Reserve: The World's First CPU Mineable Digital Commodity. Available online: https://vericoin.info/verium-digital-reserve/ (accessed on 3 August 2025).
95. Voulgaris, S.; Fotiou, N.; Siris, V.A.; Polyzos, G.C.; Jaatinen, M.; Oikonomidis, Y. Blockchain Technology for Intelligent Environments. *Future Internet* **2019**, *11*, 213. [CrossRef]
96. Purple Protocol—A Scalable Platform for Decentralized Applications and Tokenized Assets. Available online: https://github.com/purplecoin (accessed on 3 August 2025).
97. Hoffmann, F. Challenges of Proof-of-Useful-Work (PoUW). In Proceedings of the 2022 IEEE 1st Global Emerging Technology Blockchain Forum: Blockchain & Beyond (iGETblockchain), Irvine, CA, USA, 7–11 November 2022; pp. 1–5.

98. King, S. Primecoin: Cryptocurrency with Prime Number Proof-of-Work. 2013. Available online: https://bravenewcoin.com/wp-content/uploads/2023/11/b72fce6c-66c4-47e8-8772-f5d7af815450.pdf (accessed on 3 August 2025).
99. Ileri, A.M.; Ozercan, H.I.; Gundogdu, A.; Senol, A.K.; Oezkaya, M.Y.; Alkan, C. Coinami: A Cryptocurrency with DNA Sequence Alignment as Proof-of-Work. *arXiv* **2016**, arXiv:1602.03031. [CrossRef]
100. Baldominos, A.; Saez, Y. Coin.ai: A Proof-of-Useful-Work Scheme for Blockchain-Based Distributed Deep Learning. *Entropy* **2019**, *21*, 723. [CrossRef] [PubMed]
101. Lamport, L. Paxos Made Simple. *ACM SIGACT News (Distrib. Comput. Column)* **2001**, *32*, 51–58.
102. Lamport, L. Generalized Consensus and Paxos. 2005. Available online: https://www.microsoft.com/en-us/research/publication/generalized-consensus-and-paxos/ (accessed on 3 August 2025).
103. Lamport, L. Fast Paxos. *Distrib. Comput.* **2006**, *19*, 79–103. [CrossRef]
104. Kończak, J.; Wojciechowski, P.T.; Santos, N.; Żurkowski, T.; Schiper, A. Recovery Algorithms for Paxos-Based State Machine Replication. *IEEE Trans. Dependable Secur. Comput.* **2019**, *18*, 623–640. [CrossRef]
105. Jalili Marandi, P.; Primi, M.; Schiper, N.; Pedone, F. Ring Paxos: High-Throughput Atomic Broadcast. *Comput. J.* **2017**, *60*, 866–882. [CrossRef]
106. Skrzypczak, J.; Schintke, F.; Schütt, T. RMWPaxos: Fault-Tolerant In-Place Consensus Sequences. *IEEE Trans. Parallel Distrib. Syst.* **2020**, *31*, 2392–2405. [CrossRef]
107. Srinivasan, S.; Kandukoori, R. A Paxos Based Algorithm to Minimize the Overhead of Process Recovery in Consensus. *Acta Inform.* **2019**, *56*, 433–446. [CrossRef]
108. Wang, C.; Jiang, J.; Chen, X.; Yi, N.; Cui, H. Apus: Fast and Scalable Paxos on RDMA. In Proceedings of the 2017 Symposium on Cloud Computing, Santa Clara, CA, USA, 24–27 September 2017; pp. 94–107.
109. Mwotil, A.; Anderson, T.; Kanagwa, B.; Stavrinos, T.; Bainomugisha, E. LowPaxos: State Machine Replication for Low Resource Settings. *IEEE Access* **2024**, *12*, 91272–91288. [CrossRef]
110. Mazieres, D. The Stellar Consensus Protocol: A Federated Model for Internet-Level Consensus. *Stellar Dev. Found.* **2015**, *32*, 1–45.
111. Stellar. 2025. Available online: https://developers.stellar.org/docs/ (accessed on 3 August 2025).
112. García-Pérez, Á.; Gotsman, A. Federated Byzantine Quorum Systems. In Proceedings of the 22nd International Conference on Principles of Distributed Systems (OPODIS 2018), Hong Kong, China, 17–19 December 2018; pp. 17:1–17:16.
113. Losa, G.; Gafni, E.; Mazieres, D. Stellar Consensus by Instantiation. In Proceedings of the 33rd International Symposium on Distributed Computing (DISC 2019), Budapest, Hungary, 14–18 October 2019; pp. 27:1–27:15.
114. García-Pérez, Á.; Schett, M.A. Deconstructing Stellar Consensus. In Proceedings of the 23rd International Conference on Principles of Distributed Systems (OPODIS 2019), Neuchâtel, Switzerland, 17–19 December 2019; pp. 5:1–5:16.
115. 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.
116. Kim, M.; Kwon, Y.; Kim, Y. Is Stellar as Secure as You Think? In Proceedings of the 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Stockholm, Sweden, 17–19 June 2019; pp. 377–385.
117. Zoi, Z.A. Study of Consensus Protocols and Improvement of the Federated Byzantine Agreement (FBA) Algorithm. Master's Thesis , Universitat Politècnica de Catalunya, Barcelona, Spain, 2019.
118. Florian, F.; Henningsen, S.; Ndolo, C.; Scheuermann, B. The Sum of Its Parts: Analysis of Federated Byzantine Agreement Systems. *Distrib. Comput.* **2022**, *35*, 399–417. [CrossRef]
119. Tumas, V.; Rivera, S.; Magoni, D.; State, R. Federated Byzantine Agreement Protocol Robustness to Targeted Network Attacks. In Proceedings of the 2023 IEEE Symposium on Computers and Communications (ISCC), Tunis, Tunisia, 9–12 July 2023; pp. 443–449.
120. Kant, K.; Pandey, S.; Shanker, U. Addressing Blockchain Efficiency: A Study on Super Node-Based Consensus Mechanisms. In Proceedings of the 2024 1st International Conference on Advanced Computing and Emerging Technologies (ACET), Ghaziabad, India, 23–24 August 2024; pp. 1–6.
121. Distler, T.; Cachin, C.; Kapitza, R. Resource Efficient Byzantine Fault Tolerance. *IEEE Trans. Comput.* **2015**, *65*, 2807–2819. [CrossRef]
122. Miller, A.; Xia, Y.; Croman, K.; Shi, E.; Song, D. The Honey Badger of BFT Protocols. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; pp. 31–42.
123. Lei, K.; Zhang, Q.; Xu, L.; Qi, Z. Reputation-Based Byzantine Fault-Tolerance for Consortium Blockchain. In Proceedings of the 2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS), Singapore, 11–13 December 2018; pp. 604–611.
124. Qin, H.; Cheng, Y.; Ma, X.; Li, F.; Abawajy, J. Weighted Byzantine Fault Tolerance Consensus Algorithm for Enhancing Consortium Blockchain Efficiency and Security. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 8370–8379. [CrossRef]

125. He, L.; Hou, Z. An Improvement of Consensus Fault Tolerant Algorithm Applied to Alliance Chain. In Proceedings of the 2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 12–14 July 2019; pp. 1–4.
126. Gueta, G.G.; Abraham, I.; Grossman, S.; Malkhi, D.; Pinkas, B.; Reiter, M.K.; Tomescu, A. SBFT: A Scalable and Decentralized Trust Infrastructure. *arXiv* **2018**, arXiv:1804.01626.
127. Feng, L.; Zhang, H.; Chen, Y.; Lou, L. Scalable Dynamic Multi-Agent Practical Byzantine Fault-Tolerant Consensus in Permissioned Blockchain. *Appl. Sci.* **2018**, *8*, 10.
128. Gao, S.; Yu, T.; Zhu, J.; Cai, W. T-PBFT: An EigenTrust-Based Practical Byzantine Fault Tolerance Consensus Algorithm. *China Commun.* **2019**, *16*, 111–123. [CrossRef]
129. Eyal, I.; Sirer, E.G. Majority Is Not Enough: Bitcoin Mining Is Vulnerable. *Commun. ACM* **2018**, *61*, 95–102. [CrossRef]
130. Navaroj, G.I.; Julie, E.G.; Robinson, Y.H. Adaptive Practical Byzantine Fault Tolerance Consensus Algorithm in Permission Blockchain Network. *Int. J. Web Grid Serv.* **2022**, *18*, 62–82. [CrossRef]
131. Moindrot, O.; Bournhonesque, C. Proof of Stake Made Simple with Casper. Master's Thesis, ICME, Stanford University, Stanford, CA, USA, 2017.
132. Liao, J.; Gong, B.; Sun, W.; Zhang, F.W.; Ning, Z.Y.; Au, H.M. BFTRAND: Low-Latency Random Number Provider for BFT Smart Contracts. In Proceedings of the 2024 54th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Brisbane, Australia, 24–27 June 2024; pp. 389–402.
133. Buterin, V.; Griffith, V. Casper the Friendly Finality Gadget. *arXiv* **2017**, arXiv:1710.09437.
134. Buterin, V.; Reijsbergen, D.; Leonardos, S.; Piliouras, G. Incentives in Ethereum's Hybrid Casper Protocol. *Int. J. Netw. Manag.* **2020**, *30*, e2098. [CrossRef]
135. Bandara, E.; Shetty, S.; Mukkamala, R.; Liang, X.; Foytik, P.; Ranasinghe, N.; Zoysa, D.K. Casper: A Blockchain-Based System for Efficient and Secure Customer Credential Verification. *J. Bank. Financ. Technol.* **2022**, *6*, 43–62. [CrossRef]
136. Levrand, T. Wealth Inequality in CasperLabs' Proof-of-Stake Blockchain. 2022. Available online: https://capuana.ifi.uzh.ch/publications/PDFs/22758_Master_ThomasLevrand_PoSCasper.pdf (accessed on 3 August 2025).
137. Veschetti, A. A Formal Analysis of Blockchain Consensus. 2023. Available online: https://amsdottorato.unibo.it/id/eprint/10835/ (accessed on 3 August 2025).
138. Agha, A.; Otsu, K.; Morrell, B.; Fan, D.D.; Thakker, R.; Santamaria-Navarro, A. NeBula: TEAM CoSTAR's Robotic Autonomy Solution that Won Phase II of DARPA Subterranean Challenge. *Field Robot.* **2022**, *2*, 1432–1506. [CrossRef]
139. Shahaab, A.; Lidgey, B.; Hewage, C.; Khan, I. Applicability and Appropriateness of Distributed Ledgers Consensus Protocols in Public and Private Sectors: A Systematic Review. *IEEE Access* **2019**, *7*, 43622–43636. [CrossRef]
140. Crain, T.; Gramoli, V.; Larrea, M.; Raynal, M. DBFT: Efficient Leaderless Byzantine Consensus and Its Application to Blockchains. In Proceedings of the IEEE 17th International Symposium on Network Computing and Applications (NCA), Cambridge, MA, USA, 1–3 November 2018; pp. 1–8.
141. Zhang, J.; Rong, Y.; Cao, J.; Rong, C.; Bian, J.; Wu, W. DBFT: A Byzantine Fault Tolerance Protocol with Graceful Performance Degradation. *IEEE Trans. Dependable Secur. Comput.***2022**, *19*, 3387–3400. [CrossRef]
142. Zhan, Y.; Wang, B.; Lu, R.; Yu, Y. DRBFT: Delegated Randomization Byzantine Fault Tolerance Consensus Protocol for Blockchains. *Inf. Sci.* **2021**, *559*, 8–21. [CrossRef]
143. Christofi, G. Study of Consensus Protocols and Improvement of the Delegated Byzantine Fault Tolerance (DBFT) Algorithm. Master's Thesis, Universitat Politècnica de Catalunya, Barcelona, Spain, 2019.
144. Jeon, S.; Doh, I.; Chae, K. RMBC: Randomized Mesh Blockchain Using DBFT Consensus Algorithm. In Proceedings of the International Conference on Information Networking (ICOIN), Chiang Mai, Thailand, 10–12 January 2018; pp. 712–717.
145. Proof of Stake (PoS) vs. Proof of Work (PoW). 2025. Available online: https://hedera.com/learning/consensus-algorithms/proof-of-stake-vs-proof-of-work (accessed on 3 August 2025).
146. A Deep Dive into Blockchain Scalability. 2025. Available online: https://crypto.com/en/university/blockchain-scalability (accessed on 3 August 2025).
147. Proof of Work Is Not What Makes Blockchains Unscalable. 2025. Available online: https://ethereumclassic.org/ (accessed on 3 August 2025).
148. The Komodo: Documentation Orientation. Available online: https://komodoplatform.com/en/technology/delayed-proof-of-work/ (accessed on 3 August 2025).
149. Bitcoin. 2025. Available online: https://bitcoin.org/ (accessed on 3 August 2025).
150. Chong, Z.K.; Ohsaki, H.; Ng, B. Proof of Useful Intelligence (PoUI): Blockchain Consensus Beyond Energy Waste. *arXiv* **2025**, arXiv:2504.17539. [CrossRef]
151. Orlicki, J.I. PoGO: A Scalable Proof of Useful Work via Quantized Gradient Descent and Merkle Proofs. *arXiv* **2025**, arXiv:2504.07540. [CrossRef]
152. Vieira, G.M.D.; Buzato, L.E. The Performance of Paxos and Fast Paxos. *arXiv* **2013**, arXiv:1308.1358. [CrossRef]

153. Lee, S.; Kim, S. Short Selling Attack: A Self-Destructive but Profitable 51% Attack on PoS Blockchains; Cryptology ePrint Archive. 2020. Available online: https://eprint.iacr.org/2020/ (accessed on 3 August 2025).
154. What Is Federated Byzantine Agreement (FBA). 2025. Available online: https://crypto.news/what-is-federated-byzantine-agreement-fba/ (accessed on 3 August 2025).
155. Ding, X.; Lu, H.; Cheng, L. CE-PBFT: An Optimized PBFT Consensus Algorithm for Microgrid Power Trading. *Electronics* **2024**, *13*, 1942. [CrossRef]
156. Moraru, I.; Andersen, D.; Kaminsky, M. There Is More Consensus in Egalitarian Parliaments. In Proceedings of the 24th ACM Symposium on Operating Systems Principles (SOSP 2013), Farmington, PA, USA, 3–6 November 2013; pp. 358–372.
157. Tollman, S.; Park, S.J.; Ousterhout, J. EPaxos Revisited. In Proceedings of the 18th USENIX Symposium on Networked Systems Design and Implementation (NSDI 21), Berkeley, CA, USA, Virtual, 12–14 April 2021; pp. 613–632.
158. Zhang, X.; Xue, M.; Miao, X. A Consensus Algorithm Based on Risk Assessment Model for Permissioned Blockchain. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 8698009. [CrossRef]
159. Soltani, P.; Ashtiani, F. Technical Report: Analytical Modeling and Throughput Computation of Blockchain Sharding. *arXiv* **2022**, arXiv:2210.04599. [CrossRef]
160. Consensus: Paxos(etcd) vs. Nakamoto(Bitcoin). 2025. Available online: https://gyuho.dev/consensus-systems/paxos-etcd-vs-nakamoto-bitcoin/ (accessed on 3 August 2025).
161. Whittaker, M.; Giridharan, N.; Szekeres, A.; Hellerstein, J.M.; Stoica, I. Bipartisan Paxos: A Modular State Machine Replication Protocol. *arXiv* **2020**, arXiv:2003.00331.
162. Multi-Paxos. 2025. Available online: https://github.com/Tencent/phxpaxos (accessed on 3 August 2025).
163. Zhong, W.; Feng, W.; Huang, M.; Feng, S. ST-PBFT: An Optimized PBFT Consensus Algorithm for Intellectual Property Transaction Scenarios. *Electronics* **2023**, *12*, 325. [CrossRef]
164. King, S.; Nadal, S. Ppcoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake. 2012. Available online: https://bitcoin.peryaudo.org/vendor/peercoin-paper.pdf (accessed on 3 August 2025).
165. Saleh, F. Blockchain Without Waste: Proof-of-Stake. *Rev. Financ. Stud.* **2021**, *34*, 1156–1190. [CrossRef]
166. Nguyen, C.T.; Hoang, D.T.; Nguyen, D.N.; Niyato, D.; Nguyen, H.T.; Dutkiewics, E. Proof-of-Stake Consensus Mechanisms for Future Blockchain Networks: Fundamentals, Applications and Opportunities. *IEEE Access* **2019**, *7*, 85727–85745. [CrossRef]
167. 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.
168. Bala, K.; Kaur, P.D. A Novel Game Theory Based Reliable Proof-of-Stake Consensus Mechanism for Blockchain. *Trans. Emerg. Telecommun. Technol.* **2022**, *33*, e4525. [CrossRef]
169. Chen, J.; Micali, S. Algorand. *arXiv* **2016**, arXiv:1607.01341.
170. Chen, J.; Micali, S. Algorand: A Secure and Efficient Distributed Ledger. *Theoret. Comput. Sci.* **2019**, *777*, 155–183. [CrossRef]
171. Li, S.N.; Spychiger, F.; Tessone, C.J. Reward Distribution in Proof-of-Stake Protocols: A Trade-Off Between Inclusion and Fairness. *IEEE Access* **2023**, *11*, 134136–134145. [CrossRef]
172. Mišić, J.; Mišić, V.B.; Chang, X. QPoS: Decentralized Stake-Based Leader and Voter Selection in a PBFT System with Mobile Voters. *IEEE Trans. Netw. Sci. Eng.* **2025**, *12*, 653–668. [CrossRef]
173. Karantias, K.; Kiayias, A.; Zindros, D. Proof-of-Burn. In Proceedings of the Financial Cryptography and Data Security: 24th International Conference (FC 2020), Kota Kinabalu, Malaysia, 10–14 February 2020; pp. 523–540.
174. Rodinko, M.; Oliynykov, R.; Nastenko, A. Decentralized Proof-of-Burn Auction for Secure Cryptocurrency Upgrade. *Blockchain Res. Appl.* **2024**, *5*, 100170. [CrossRef]
175. Pillai, B.; Hóu, Z.; Biswas, K.; Muthukkumarasamy, V. Formal Verification of the Burn-to-Claim Blockchain Interoperable Protocol. In Proceedings of the International Conference on Formal Engineering Methods, Brisbane, Australia, 21–24 November 2023; pp. 249–254.
176. Menon, A.A.; Saranya, T.; Sureshbabu, S.; Mahesh, A.S. A Comparative Analysis on Three Consensus Algorithms: Proof of Burn, Proof of Elapsed Time, Proof of Authority. In *Computer Networks and Inventive Communication Technologies: Proceedings of Fourth ICCNCT 2021*; Springer: Singapore, 2022; pp. 369–383.
177. Tschorsch, F.; Scheuermann, B. Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies. *IEEE Commun. Surv. Tuts.* **2016**, *18*, 2084–2123. [CrossRef]
178. 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 Stabilization, Safety, and Security of Distributed Systems: 19th International Symposium (SSS 2017), Boston, MA, USA, 5–8 November 2017; pp. 282–297.
179. Pal, A.; Kant, K. DC-PoET: Proof-of-Elapsed-Time Consensus with Distributed Coordination for Blockchain Networks. In Proceedings of the 2021 IFIP Networking Conference (IFIP Networking), Virtual, 21–24 June 2021; pp. 1–9.
180. Bowman, M.; Das, D.; Mandal, A.; Montgomery, H. On Elapsed Time Consensus Protocols. In Proceedings of the Progress in Cryptology–INDOCRYPT 2021, Jaipur, India, 12–15 December 2021; pp. 559–583.

181. Khan, A.A.; Dhabi, S.; Yang, J.; Alhakami, W.; Bourouis, S.; Yee, P.L. B-LPoET: A Middleware Lightweight Proof-of-Elapsed Time (PoET) for Efficient Distributed Transaction Execution and Security on Blockchain Using Multithreading Technology. *Comput. Electr. Eng.* **2024**, *118*, 109343. [CrossRef]
182. Kumar, M.A.; Radhesyam, V.; SrinivasaRao, B. Front-End IoT Application for the Bitcoin Based on Proof of Elapsed Time (PoET). In Proceedings of the 2019 Third International Conference on Inventive Systems and Control (ICISC), Coimbatore, India, 10–11 January 2019; pp. 646–649.
183. 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.
184. Ateniese, G.; Bonacina, I.; Faonio, A.; Galesi, N. Proofs of Space: When Space Is of the Essence. In Proceedings of the Security and Cryptography for Networks: 9th International Conference (SCN 2014), Amalfi, Italy, 3–5 September 2014; pp. 538–557.
185. Jiang, S.; Wu, J. A Game-Theoretic Approach to Storage Offloading in PoC-Based Mobile Blockchain Mining. In Proceedings of the Twenty-First International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing, Virtual, 11–14 October 2020; pp. 171–180.
186. The Foundation for Our Sustainable Future. Available online: https://signum.network/whitepaper (accessed on 3 August 2025).
187. Wood, G. Polkadot: Vision for a Heterogeneous Multi-Chain Framework. *White Pap.* **2016**, *21*, 4662.
188. Aniello, L.; Baldoni, R.; Lombardi, F.; Sassone,V. PBFT vs Proof-of-Authority: Applying the CAP Theorem to Permissioned Blockchain. In *CEUR Workshop Proceedings*; CEUR-WS: Aachen, Germany, 2018; Volume 2058.
189. Alrubei, S.; Ball, E.; Rigelsford, J. HDPoA: Honesty-Based Distributed Proof of Authority via Scalable Work Consensus Protocol for IoT-Blockchain Applications. *Comput. Netw.* **2022**, *217*, 109337. [CrossRef]
190. Fahim, S.; Rahman, S.K.; Mahmood, S. Blockchain: A Comparative Study of Consensus Algorithms PoW, PoS, PoA, PoV. *Int. J. Math. Sci. Comput.* **2023**, *3*, 46–57. [CrossRef]
191. Joshi, S. Feasibility of Proof of Authority as a Consensus Protocol Model. *arXiv* **2021**, arXiv:2109.02480. [CrossRef]
192. Consensus Mechanism. 2025. Available online: https://docs.kaia.io/learn/consensus-mechanism (accessed on 3 August 2025).
193. Eos. 2025. Available online: https://eos.io/ (accessed on 3 August 2025).
194. Ethereum. 2025. Available online: https://ethereum.org/ (accessed on 3 August 2025).
195. Cargocoin. 2025. Available online: https://cargocoin.io/ (accessed on 3 August 2025).
196. Ambrous. 2025. Available online: https://ambrosus.com/ (accessed on 3 August 2025).
197. Yadav, A.K.; Singh, K.; Amin, A.H.; Almutairi, L.; Alsenani, T.R.; Ahmadian, A. A Comparative Study on Consensus Mechanism with Security Threats and Future Scopes: Blockchain. *Comput. Commun.* **2023**, *201*, 102–115. [CrossRef]
198. Moraru, I.; Andersen, D.G.; Kaminsky, M. Egalitarian Paxos. In Proceedings of the ACM Symposium on Operating Systems Principles, St. Louis, MO, USA, 21–24 October 2012; pp. 1–12.
199. Moraru, I.; Andersen, D.G.; Kaminsky, M. *A Proof of Correctness for Egalitarian Paxos*; Technical Report; Parallel Data Laboratory, Carnegie Mellon University: Pittsburgh, PA, USA, 2013.
200. Sutra, P. On the Correctness of Egalitarian Paxos. *Inf. Process. Lett.* **2020**, *156*, 105901. [CrossRef]
201. Ceolin, T.; Dotti, F.; Pedone, F. Parallel State Machine Replication from Generalized Consensus. In Proceedings of the 2020 International Symposium on Reliable Distributed Systems (SRDS), Shanghai, China, 21–24 September 2020; pp. 133–142.
202. Hu, J.; Liu, K. Raft Consensus Mechanism and the Applications. *J. Phys. Conf. Ser.* **2020**, *1544*, 012079. [CrossRef]
203. Howard, H. *Arc: Analysis of Raft Consensus*; Technical Report; University of Cambridge, Computer Laboratory: Cambridge, UK, 2014.
204. Ongaro, D.; Ousterhout, J. In Search of an Understandable Consensus Algorithm. In Proceedings of the 2014 USENIX Annual Technical Conference (USENIX ATC 14), Philadelphia, PA, USA, 17–20 June 2014; pp. 305–319.
205. Howard, H.; Mortier, R. Paxos vs Raft: Have We Reached Consensus on Distributed Consensus? In Proceedings of the 7th Workshop on Principles and Practice of Consistency for Distributed Data, Heraklion, Greece, 27 April 2020; pp. 1–9.
206. Yang, D.; Doh, I.; Chae, K. Cell Based Raft Algorithm for Optimized Consensus Process on Blockchain in Smart Data Market. *IEEE Access* **2022**, *10*, 85199–85212. [CrossRef]
207. Lu, S.; Zhang, X.; Zhao, R.; Chen, L.; Li, J.;Yang, G. P-Raft: An Efficient and Robust Consensus Mechanism for Consortium Blockchains. *Electronics* **2023**, *12*, 2271. [CrossRef]
208. Bai, F.; Li, F.; Shen, T.; Zeng, K.; Zhang, X.; Zhang, C. RaBFT: An Improved Byzantine Fault Tolerance Consensus Algorithm Based on Raft. *J. Supercomput.* **2024**, *80*, 21533–21560. [CrossRef]
209. Fu, W.; Wei, X.; Tong, S. An Improved Blockchain Consensus Algorithm Based on Raft. *Arab. J. Sci. Eng.* **2021**, *46*, 8137–8149. [CrossRef]
210. Abdorrahimi, B.; Nekouie, A.; Rahmani, A.M.; Lansky, J.; Nulicek, V.; Hosseinzadeh, M.; Moattar, M.H. Blockchain Technology and Raft Consensus for Secure Physician Prescriptions and Improved Diagnoses in Electronic Healthcare Systems. *Sci. Rep.* **2024**, *14*, 15692. [CrossRef] [PubMed]

211. Luo, H.; Yang, X.; Yu, H.; Sun, G.; Lei, B.; Guizani, M. Performance Analysis and Comparison of Nonideal Wireless PBFT and RAFT Consensus Networks in 6G Communications. *IEEE Internet Things J.* **2023**, *11*, 9752–9765. [CrossRef]
212. Huang, D.; Ma, X.; Zhang, S. Performance Analysis of the Raft Consensus Algorithm for Private Blockchains. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *50*, 172–181. [CrossRef]
213. Woos, D.; Wilcox, J.R.; Anton, S.; Tatlock, Z.; Ernst, M.D.; Anderson, T. Planning for Change in a Formal Verification of the Raft Consensus Protocol. In Proceedings of the 5th ACM SIGPLAN Conference on Certified Programs and Proofs (CPP 2016), St. Petersburg, FL, USA, 20–22 January 2016; pp. 154–165.
214. Li, Y.; Fan, Y.; Zhang, L.; Crowcroft, J. RAFT Consensus Reliability in Wireless Networks: Probabilistic Analysis. *IEEE Internet Things J.* **2023**, *10*, 12839–12853. [CrossRef]
215. Yu, G.; Wang, X.; Yu, K.; Ni, W.; Zhang, J.A.; Liu, R.P. Survey: Sharding in Blockchains. *IEEE Access* **2020**, *8*, 14155–14181. [CrossRef]
216. Dang, H.; Dinh, T.T.A.; Loghin, D.; Chang, E.C.; Lin, Q.; Ooi, B.C. Towards Scaling Blockchain Systems via Sharding. In Proceedings of the 2019 International Conference on Management of Data, Amsterdam, The Netherlands, 30 June–5 July 2019; pp. 123–140.
217. Tao, L.; Lu, Y.; Fan, Y.; Tan, C.W.; Wei, Z. Optimized Consensus Group Selection Focused on Node Transmission Delay in Sharding Blockchains. *IEEE Trans. Comput. Soc. Syst.* **2024** ,*12*, 3. [CrossRef]
218. Chen, R.; Wang, L.; Peng, C.; Zhu, R. An Effective Sharding Consensus Algorithm for Blockchain Systems. *Electronics* **2022**, *11*, 2597. [CrossRef]
219. Wu, X.; Jiang, W.; Song, M.; Jia, Z.; Qin, J. An Efficient Sharding Consensus Algorithm for Consortium Chains. *Sci. Rep.* **2023**, *13*, 20. [CrossRef] [PubMed]
220. Liu, A.; Chen, J.; He, K.; Du, R.; Xu, J.; Wu, C. Dynashard: Secure and Adaptive Blockchain Sharding Protocol with Hybrid Consensus and Dynamic Shard Management. *IEEE Internet Things J.* **2024** , *12*, 5462–5475. [CrossRef]
221. Kokoris-Kogias, E. Robust and Scalable Consensus for Sharded Distributed Ledgers. 2019. Available online: https://eprint.iacr.org/2019/676.pdf (accessed on 3 August 2025).
222. Lu, X.; Jiang, C.; Wang, P. A Survey on Consensus Algorithms of Blockchain Based on DAG. In Proceedings of the 2024 6th Blockchain and Internet of Things Conference, Fukuoka, Japan, 19–21 July 2024; pp. 50–58.
223. Li, C.; Li, P.; Zhou, D.; Yang, Z.; Wu, M.; Yang, G.; Xu, W.; Long, F.; Yao, A.C.C. A Decentralized Blockchain with High Throughput and Fast Confirmation. In Proceedings of the 2020 USENIX Annual Technical Conference (USENIX ATC 20), Boston, MA, USA, 15–17 July 2020; pp. 515–528.
224. Sompolinsky, Y.; Lewenberg, Y.; Zohar, A. Spectre: A Fast and Scalable Cryptocurrency Protocol. Cryptology ePrint Archive. 2016. Available online: https://eprint.iacr.org/2016/1159 (accessed on 3 August 2025).
225. Sompolinsky, Y.; Wyborski, S.; Zohar, A. PHANTOM GHOSTDAG: A Scalable Generalization of Nakamoto Consensus. In Proceedings of the 3rd ACM Conference on Advances in Financial Technologies, Arlington, VA, USA, 26–28 September 2021; pp. 57–70.
226. Baird, L. *The Swirlds Hashgraph Consensus Algorithm: Fair, Fast, Byzantine Fault Tolerance*; Technical Report SWIRLDS-TR-2016-01; Swirlds: Dallas, TX, USA, 2016; Volume 34, pp. 9–11.
227. Pervez, H.; Muneeb, M.; Irfan, M.U.; Haq, I.U. A Comparative Analysis of DAG-Based Blockchain Architectures. In Proceedings of the 2018 12th International Conference on Open Source Systems and Technologies (ICOSST), Lahore, Pakistan, 19–21 December 2018; pp. 27–34.
228. Cao, B.; Li, Y.; Zhang, L.; Mumtaz, S.; Zhou, Z.; Peng, M. When Internet of Things Meets Blockchain: Challenges in Distributed Consensus. *IEEE Netw.* **2019**, *33*, 133–139. [CrossRef]
229. Raikwar, M.; Polyanskii, N.; Müller, S. SoK: DAG-Based Consensus Protocols. In Proceedings of the 2024 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Dublin, Ireland, 27–31 May 2024; pp. 1–18.
230. Wang, Q.; Yu, J.; Chen, S.; Xiang, Y. SoK: DAG-Based Blockchain Systems. *ACM Comput. Surv.* **2023**, *55*, 1–38. [CrossRef]
231. Qu, X.; Wang, S.; Li, K.; Huang, J.; Cheng, X. TidyBlock: A Novel Consensus Mechanism for DAG-Based Blockchain in IoT. *IEEE Trans. Mobile Comput.* **2024**, *24*, 722–735. [CrossRef]
232. Dong, Z.; Wu, H.; Li, Z.; Mi, D.; Popoola, O.; Zhang, L. Trustworthy VANET: Hierarchical DAG-Based Blockchain Solution with Proof of Reputation Consensus Algorithm. In Proceedings of the 2023 IEEE International Conference on Blockchain (Blockchain), Hainan, China, 17–21 December 2023; pp. 127–132.
233. Wu, H.; Yue, C.; Zhang, L.; Li, Y.; Imran, M.A. When Distributed Consensus Meets Wireless Connected Autonomous Systems: A Review and A DAG-Based Approach. *IEEE Netw.* **2024**, *39*, 261–269. [CrossRef]
234. Deng, X.; Li, K.; Wang, Z.; Liu, H. A Novel Consensus Algorithm Based on Segmented DAG and BP Neural Network for Consortium Blockchain. *Secur. Commun. Netw.* **2022**, *2022*, 1060765. [CrossRef]
235. Zhang, X.; Li, R.; Zhao, H. A Parallel Consensus Mechanism Using PBFT Based on DAG-Lattice Structure in the Internet of Vehicles. *IEEE Internet Things J.* **2022**, *10*, 5418–5433. [CrossRef]

236. Sasikumar, A.; Senthilkumar, N.; Subramaniyaswamy, V.; Kotecha, K.; Indragandhi, V.; Ravi, L. An Efficient, Provably-Secure DAG Based Consensus Mechanism for Industrial Internet of Things. *Int. J. Interact. Des. Manuf.* **2023**, *17*, 2197–2207. [CrossRef]
237. Li, L.; Huang, D.; Zhang, C. An Efficient DAG Blockchain Architecture for IoT. *IEEE Internet Things J.* **2022**, *10*, 1286–1296. [CrossRef]
238. Ding, Y.; Sato, H. Dagbase: A Decentralized Database Platform Using DAG-Based Consensus. In Proceedings of the 2020 IEEE 44th Annual Computers, Software, and Applications Conference (COMPSAC), Madrid, Spain, 13–17 July 2020; pp. 798–807.
239. Zhou, T.; Li, X.; Zhao, H. DLattice: A Permission-Less Blockchain Based on DPoS-BA-DAG Consensus for Data Tokenization. *IEEE Access* **2019**, *7*, 39273–39287. [CrossRef]
240. Park, S.; Kim, H. DAG-Based Distributed Ledger for Low-Latency Smart Grid Network. *Energies* **2019**, *12*, 3570. [CrossRef]
241. Dai, X.; Zhang, Z.; Xiao, J.; Yue, J.; Xie, X.; Jin, H. GradedDAG: An Asynchronous DAG-Based BFT Consensus with Lower Latency. In Proceedings of the 2023 42nd International Symposium on Reliable Distributed Systems (SRDS), Marrakech, Morocco, 27–29 September 2023; pp. 107–117.
242. Dai, X.; Wang, G.; Xiao, J.; Guo, Z.; Hao, R.; Xie, X. LightDAG: A Low-Latency DAG-Based BFT Consensus Through Lightweight Broadcast. In Proceedings of the 2024 IEEE International Parallel and Distributed Processing Symposium (IPDPS), San Francisco, CA, USA, 27–31 May 2024; pp. 998–1008.
243. Jovanovic, P.; Kogias, L.K.; Kumara, B.; Xonnino, A.; Tennage, P.; Zablotchi, Z. Mahi-Mahi: Low-Latency Asynchronous BFT DAG-Based Consensus. *arXiv* **2024**, arXiv:2410.08670.
244. Xiang, F.; Huaimin, W.; Peichang, S.; Xue, O.; Xunhiu, Z. Jointgraph: A DAG-Based Efficient Consensus Algorithm for Consortium Blockchains. *Softw. Pract. Exp.* **2021**, *51*, 1987–1999. [CrossRef]
245. Fu, X.; Wang, H.; Shi, P.; Zhang, X. Teegraph: A Blockchain Consensus Algorithm Based on TEE and DAG for Data Sharing in IoT. *J. Syst. Archit.* **2022**, *122*, 102344. [CrossRef]
246. Tennage, P.; Desjardins, A.; Kogias, E.K. Mandator and Sporades: Robust Wide-Area Consensus with Efficient Request Dissemination. *arXiv* **2022**, arXiv:2209.06152. [CrossRef]
247. Scalable but Wasteful or Why Fast Replication Protocols Are Actually Slow. 2025. Available online: https://charap.co/scalable-but-wasteful-or-why-fast-replication-protocols-are-actually-slow/ (accessed on 3 August 2025).
248. Sedlmeir, J.; Buhl, H.U.; Fridgen, G.; Keller, R. The Energy Consumption of Blockchain Technology: Beyond Myth. *Bus. Inf. Syst. Eng.* **2020**, *62*, 599–608. [CrossRef]
249. Gochain. 2025. Available online: https://gochain.io/go (accessed on 3 August 2025).
250. Proof of Authority (PoA) Meaning. 2025. Available online: https://www.ledger.com/academy/glossary/proof-of-authority (accessed on 3 August 2025).
251. Liu, Y.; Liu, A.; Lu, Y.; Pan, Z.; Li, Y.; Vian, S.; Conti, M. Kronos: A Secure and Generic Sharding Blockchain Consensus with Optimized Overhead; Cryptology ePrint Archive: 2024. Available online: https://eprint.iacr.org/2024/ (accessed on 3 August 2025).
252. Baageel, H.; Rahman, M.M. Leveraging Sharding-Based Hybrid Consensus for Blockchain. *Comput. Mater. Contin.* **2024**, *81*, 1. [CrossRef]
253. Wang, D.; Zhang, X. Secure Ride-Sharing Services Based on a Consortium Blockchain. *IEEE Internet Things J.* **2021**, *8*, 2976–2991. [CrossRef]
254. Li, M.; Luo, X.; Xue, K.; Xue, Y.; Sun, W.; Li, J. A Secure and Efficient Blockchain Sharding Scheme via Hybrid Consensus and Dynamic Management. *IEEE Trans. Inf. Forensics Secur.* **2024**, *19*, 5911–5924. [CrossRef]
255. Xie, S.; Kang, D.; Lyu, H.; Sadoghi,M. Fides: Scalable Censorship-Resistant DAG Consensus via Trusted Components. *arXiv* **2025**, arXiv:2501.01062. [CrossRef]
256. Khan, M.; Hartog, F.D.; Hu, J. Toward Verification of DAG-Based Distributed Ledger Technologies through Discrete-Event Simulation. *Sensors* **2024**, *24*, 1583. [CrossRef] [PubMed]
257. Sompolinsky, Y.; Sutton, M. The DAG Knight Protocol: A Parameterless Generalization of Nakamoto Consensus; Cryptology ePrint Archive: 2022. Available online: https://eprint.iacr.org/2022/ (accessed on 3 August 2025).
258. Dag-vs-Blockchain. 2025. Available online: https://crustlab.com/blog/dag-vs-blockchain (accessed on 3 August 2025).
259. Blocktivity. 2025. Available online: https://blocktivity.info/ (accessed on 3 August 2025).
260. Burstiq. 2025. Available online: https://www.burstiq.com/ (accessed on 3 August 2025).
261. Hyperledger. 2025. Available online: https://hyperledger-fabric.readthedocs.io/ (accessed on 3 August 2025).
262. Guardtime. 2025. Available online: https://guardtime.com/publications/ (accessed on 3 August 2025).
263. Fateminasab, S.S.; Bahrepour, D.; Tabbakh, S.R.K. A Fair Non-Collateral Consensus Protocol Based on Merkle Tree for Hierarchical IoT Blockchain. *Sci. Rep.* **2025**, *15*, 3645. [CrossRef]
264. Zhang, H.; Zhao, Y. SatBFT: An Efficient and Scalable Consensus Protocol for Blockchain-Enabled Space-Air-Ground Integrated Network. *IEEE Trans. Cogn. Commun. Netw.* **2025**. [CrossRef]

265. Liu, J.; Xie, M.; Chen, S.; Ma, C.; Gong, Q. An Improved DPoS Consensus Mechanism in Blockchain Based on PLTS for the Smart Autonomous Multi-Robot System. *Inf. Sci.* **2021**, *575*, 528–541. [CrossRef]
266. Jain, A.K.; Gupta, N.; Gupta, B.B. A Survey on Scalable Consensus Algorithms for Blockchain Technology. *Cyber Secur. Appl.* **2025**, *3*, 100065. [CrossRef]
267. Kushilevitz, E. Communication Complexity. In *Advances in Computers*; Zelkowitz, M.V., Ed.; Elsevier: Amsterdam, The Netherlands, 1997; Volume 44, pp. 331–360.
268. Bains, P. Blockchain Consensus Mechanisms: A Primer for Supervisors. *FinTech Notes* **2022**, *2022*, A001. [CrossRef]
269. Delegated Proof of Stake (DPoS): A Comprehensive Guide. 2025. Available online: https://primexbt.com/for-traders/what-is-delegated-proof-of-stake-dpos/ (accessed on 3 August 2025).
270. Blockchain Comparison. 2025. Available online: https://docs.venly.io/docs/blockchain-networks-comparison (accessed on 3 August 2025).