symmetry

MDPI

Review

Blockchain Consensus: An Overview of Alternative Protocols

Damilare Peter Oyinloye [1,2], Je Sen Teh [1,*], Norziana Jamil [3,*] and Moatsum Alawida [1,4]

1 School of Computer Sciences, Universiti Sains Malaysia, Pulau Pinang 11800, Malaysia; damilare.oyinloye@student.usm.my (D.P.O.); moatsum.alawida@ku.ac.ae (M.A.)
2 Department of Computer Science, Kwara State University, Malete 241104, Nigeria
3 Department of Computing, College of Computing and Informatics, Universiti Tenaga Nasional, Selangor 43000, Malaysia
4 Center of Cyber-Physical Systems, Khalifa University, Abu Dhabi 127788, United Arab Emirates
* Correspondence: jesen_teh@usm.my (J.S.T.); norziana@uniten.edu.my (N.J.)

Abstract: Blockchain networks are based on cryptographic notions that include asymmetric-key encryption, hash functions and consensus protocols. Despite their popularity, mainstream protocols, such as Proof of Work or Proof of Stake still have drawbacks. Efforts to enhance these protocols led to the birth of *alternative* consensus protocols, catering to specific areas, such as medicine or transportation. These protocols remain relatively unknown despite having unique merits worth investigating. Although past reviews have been published on popular blockchain consensus protocols, they do not include most of these lesser-known protocols. Highlighting these alternative consensus protocols contributes toward the advancement of the state of the art, as they have design features that may be useful to academics, blockchain practitioners and researchers. In this paper, we bridge this gap by providing an overview of alternative consensus protocols proposed within the past 3 years. We evaluate their overall performance based on metrics such as throughput, scalability, security, energy consumption, and finality. In our review, we examine the trade-offs that these consensus protocols have made in their attempts to optimize scalability and performance. To the best of our knowledge, this is the first paper that focuses on these alternative protocols, highlighting their unique features that can be used to develop future consensus protocols.

Keywords: blockchain; consensus protocol; distributed ledger technology; energy; finality; scalability; security; throughput

Citation: Oyinloye, D.P.; Teh, J.S..; Alawida, M.; Jamil, N. Blockchain Consensus: An Overview of Alternative Protocols. *Symmetry* **2021**, *13*, 1363. https://doi.org/10.3390/sym13081363

Academic Editor: Kuo-Hui Yeh

Received: 21 June 2021
Accepted: 9 July 2021
Published: 27 July 2021

1. Introduction

Since its inception, blockchain technology and its applications have been of great interest in the financial sector. One of the most popular use cases of blockchain, cryptocurrencies, relies on cryptographic algorithms, such as asymmetric-key encryption and hash functions, to facilitate secure financial transactions between various parties or nodes [1]. Blockchain can be viewed as a fully replicated distributed database system that keeps a record of all transactions in a network. All of these transactions are also stored by every contributing node or unit within the network itself [2]. It has a distributed consensus protocol running on every participating node, managing message exchanges and local decision making to enforce network consistency. Consensus protocols are a set of rules that participating nodes in a network use to decide whether a transaction is valid [3]. This ensures that all participants collectively maintain a common transaction ledger.

On cryptocurrency platforms, blockchain keeps track of currency transactions in a successive and orderly form to achieve a tamper-proof record [4]. With the rise of Ethereum and other blockchain-based distributed computing platforms, transactions also include data that are required to execute smart contracts. Blockchain as a term connotes that transaction records between participating nodes in a system or network are stored in a data format known as a "block". A sequential set of these connected blocks arranged in ascending order is referred to as the chain of blocks (i.e., "blockchain"). Permission-less

blockchain networks do not need a centralized authorizing figure; interested nodes can join without restrictions, and it permits a large number of participating nodes in the consensus process [5,6].

Bitcoin is without a doubt the most mainstream or well-known use case of blockchain technology. It was introduced as an alternative form of currency with the aim of overcoming the limitations of fiat money [7]. Bitcoin's underlying blockchain mechanism keeps a record of cryptocurrency transactions among participating nodes. Each node involved in the transaction possesses two keys: a private key and a public key [8]. The transaction address of a participating node is the hash value of the public key, which also serves as the node's identity. As multiple private–public key pairs can be generated, nodes can thus have more than one identity, leading to attacks such as a Sybil attack [9]. Transactions are signed with a node's private key and are sent to all other nodes in the network for verification. With a decentralized consensus architecture, the participating nodes come to an agreement on both the sequence and validity of all transactions. The records of these transactions are stored in blocks.

The current block of any block sequence has a timestamp and is connected to prior blocks by cryptographic hash values. Participating nodes cannot delete a block but can append new ones. The chaining of these blocks results in a shared, distributed database with an immensely growing record of transactions that are irreversible and immutable. It is difficult for anyone to tamper with block information without other nodes detecting the changes [10]. Unlike the days of centralized ledgers in the custody of a single point of authority, blockchains are essentially decentralized databases that are collaboratively managed by multiple participating entities. One important element of blockchain technology is its underlying consensus mechanism, known as the consensus protocol. These protocols decide which node is allowed to add a new block to the chain. Blockchain consensus protocols can be classified into two distinct classes: (1) proof-based consensus protocols, which require that entities provide proof of effort or resource expenditure, and (2) voting-based consensus protocols, which require participating entities in the network to exchange their results of verifying new blocks or transactions before making the final decision about which node is allowed to commit a new block to the chain [11]. Well-established consensus protocols include Proof of Work (PoW) [12], Proof of Stake (PoS) [13] and their variants [14].

1.1. Contribution and Scope

Over the years, numerous lesser-known alternative blockchain consensus protocols have been proposed, some of which are for specific areas of application, such as medicine or electric vehicles. Although many of these protocols are currently of academic interest rather than for practical applications, they have unique features that may be useful for the advancement of blockchain technologies. Past reviews have mostly covered conventional consensus protocols, evaluating them based on various metrics, such as throughput, scalability and security. In this paper, we present a survey of alternative consensus protocols that have been proposed within the past three years (as of April 2021), evaluating their overall performance based on metrics depicted in Figure 1. Our sources include academic papers, white papers, and official documentation of cryptocurrencies. White papers and official documentation were used mainly for the mainstream protocols, whereas alternative protocols were taken from academic papers. These academic papers were extracted from Scopus, using the search keywords "*blockchain*, *consensus*, *protocol*", and protocols that were solely based on mainstream protocols (PoW, PoS) were eliminated. We define *alternative protocols* as those that are lesser known and do not fall directly into the same category as the popular consensus protocols listed in Section 3 of our paper.

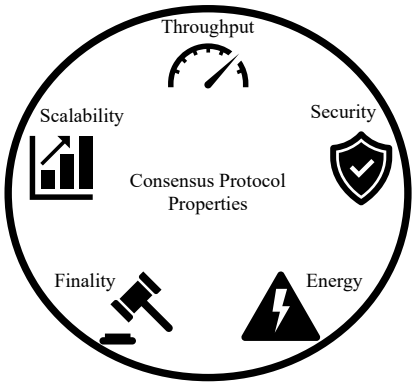

Figure 1. Consensus protocol properties.

The aim of this paper is to provide, at a glance, the advances in alternative consensus protocols to both new and experienced researchers alike. The paper also contains an introduction to widely adopted, conventional consensus protocols, such as PoW and PoS, for the benefit of readers who are new to the area, and also to define the scope of what we consider to be *popular* or *conventional* protocols as opposed to *alternative* protocols. As such, coverage of these well-known protocols will not delve too deeply into their details, which are already widely available in many other sources. In addition, the purpose of this paper is not to advocate the adoption of these alternative protocols, but rather to highlight their unique features that can be useful to researchers in the blockchain space. A summary of our contributions are as follows:

- An introduction to widely adopted, conventional consensus protocols, such as PoW and PoS, for the benefit of readers who are new to the area.
- A survey of alternative consensus protocols that have been proposed within the past three years (as of April 2021).
- Categorization of alternative consensus protocols based on their properties as depicted in Figure 2.
- An evaluation of the overall performance of these alternative protocols based on metrics depicted in throughput, security, energy consumption, finality, and scalability.
- A critical analysis of alternative protocols based on their properties, advantages, and disadvantages.

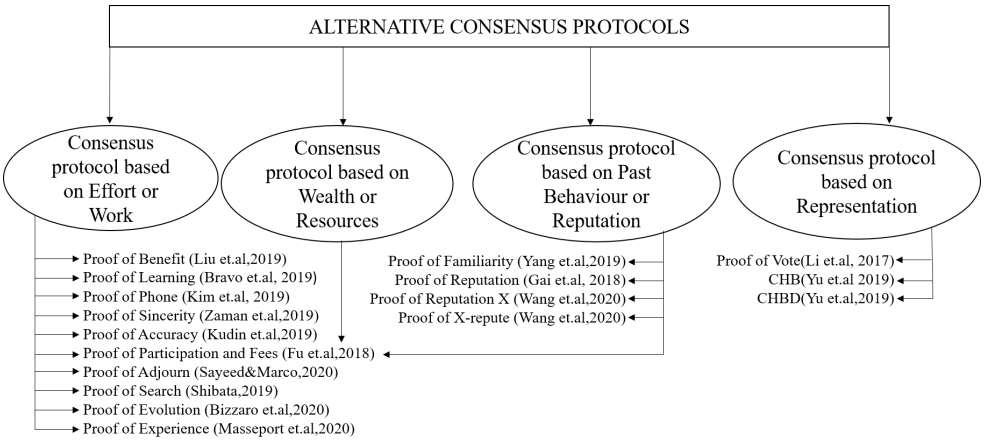

Figure 2. Taxonomy of alternative consensus protocols.

1.2. Outline

The rest of this paper is organized as follows: In Section 2, we briefly cover past review and survey papers on consensus protocols, followed by a discussion of well-known

consensus protocols and their applications in Section 3. Next, we delve into an in-depth discussion of the alternative consensus protocols in Section 4, and provide a thorough performance comparison in terms of throughput, scalability, energy consumption, finality and security implications in Section 5. A discussion of our findings, open problems and future work in the area is provided in Section 6 followed by Section 7, which concludes this review with some final remarks.

2. Past Reviews

This section covers previously published surveys and review papers about consensus protocols in chronological order. In 2016, Vukolic evaluated two major genres of blockchain consensus protocols, namely, Proof of Work (PoW) and Byzantine Fault Tolerance (BFT)-based protocols, with respect to their transaction capacity, scalability limits, power consumption, correctness proof, consensus finality, network synchrony assumptions and security [15]. PoW-based protocols scale well with network size and allow for permissionless access, but they support a very limited transaction capacity. On the other hand, they show that BFT protocols have a higher transaction capacity but a permissioned network for identity management is required and is less scalable. Apart from highlighting their major differences, this paper also suggests the hybrid use of BFT and PoW to enhance the overall performance of the consensus protocols.

Cachin et al. assessed the resilience of consensus protocols, including the ones used by permissioned blockchain networks when exposed to faults and adversarial nodes in their 2017 review [16]. The authors advocated that blockchain developers follow standard practices in computer security and cryptography, which includes relying on reviews (public and expert), discussions, detailed models, broad validation and formal proofs. Additionally, in 2017, Shehar Banor et al. presented a study of some classical blockchain consensus protocols [17]. It discussed the important components of major classical consensus protocols, including consensus protocols based on PoW, Proof of X (PoX) protocols that replace PoW with more energy-efficient alternatives, and protocols that are combinations or variations of classical consensus protocols. The paper analyzed these protocols based on metrics such as performance, security and design properties. These metrics were used to systematize the key themes in the protocol categories described above.

Wang et al. provided a broad overview of the organization of blockchain networks in 2018 [18]. Their paper highlighted the unique characteristics of incentivized consensus protocols in a typical blockchain network, with a focus on both the perspectives of distributed consensus system design and incentive mechanism design. They covered adoption strategies for self-organization by the individual nodes in blockchain networks using game theory. Emerging applications of blockchain were also highlighted.

In 2019, Yang Xiao et al. presented a review of mainstream blockchain consensus protocols [19]. Their review introduced key definitions and presented analytical results for the classical theory of fault tolerance. Subsequently, it identified five elements that are vital to any blockchain network, which include block proposal, validation, finalization, an incentive component and information propagation. Some of the popular blockchain consensus protocols were analyzed and compared based on these components. Their findings provided a clearer insight into the strengths, suitability and drawbacks with regards to fault tolerance, scalability and drawbacks. Notably, many of these protocols were still under development and were subject to major changes at the time of publication. In the same year, Shikah et al. conducted a survey on some of the major blockchain consensus protocols [20]. They analyzed factors that affect the overall performance and security of consensus protocols, provided a classification of blockchain consensus protocols in addition to detailed comparisons in terms of their strengths and drawbacks.

More recently, Ismail and Materwala surveyed the evolution of blockchain architecture and consensus protocols [21]. They classified blockchain architectures into three different distinct types and presented a table mapping these architectures to corresponding development platforms. They also introduced a classification and comparison of mainstream

consensus protocols. A critical analysis of several issues prevailing in blockchain and possible solutions were proposed for the issues. Future directions toward the development of an energy-efficient and scalable blockchain architecture and protocols were proposed. In 2020, Lepore et al. presented a review of popular consensus protocols, which included a relatively new protocol known as pure PoS [22]. Their work provided an in-depth comparison of PoW, PoS and the Pure PoS with throughput and scalability as metrics. Some basic cryptographic and blockchain technology were also introduced. A performance analysis using the consistency-availability-partition tolerance (CAP) theorem and performance comparison was presented.

Past reviews have mostly covered well-established consensus protocols as summarized in Table 1. However, there has been limited analysis of alternative or lesser-known consensus protocols, many of which have been proposed within the past three years. These protocols may have features, advantages or even shortcomings that may be useful to researchers and practitioners for enhancing existing protocols or introducing new ones.

Table 1. Summary of past reviews.

Author	Consensus Protocols	Evaluation Metrics
This Paper	**15 Alternative Protocols (Section 4)**	**Energy Consumption, Scalability, Finality, Security, Throughput**
Xiao et al. [19]	PoW, Hybrid PoW-BFT, Chain-based PoS, Committee-based PoS, BFT-based, PoS, DPoS, PoA, PoET, PoTS, Proof of Reputation (PoR, Ripple protocol	Block proposal, block validation, information propagation, block finalization, incentive mechanism, fault tolerance, transaction capacity
Vukolic [15]	PoW, BFT	Node identity management, consensus finality, scalability (No. of nodes), scalability (No. of clients), performance (throughput, latency), power consumption, fault tolerance
Banor et al. [17]	PoW, PoS, PoR	Security (transaction censorship resistance, DoS resistance, adversary), performance (throughput, scalability, latency and experimental setup)
Wang et al. [18]	PoW, Proof of Exercise, Proof of Retrievability, PoC, Proof of Human Work	Origin of hardness, design goal, implementation description, zero-knowledge proof properties, simulation of a random function, features of puzzle design
Shikah et al. [20]	PoW, PoS, DPoS, PoA, PoI, Proof of Luck, PBFT, Raft	Node identity management, data model, electing miners, energy-saving ability, fault tolerance (Byzantine, crash, 51% attack), transaction fee, block reward, performance (verification speed, throughput, block creation speed), scalability, double spending
Ismail and Materwala [21]	PoW, DPoW, PoS, DPoS, Proof of Stake Velocity (PoSV), PoB, PoC, PoH, PoI, Proof of Believability, PoA, PoET, Proof of Activity, PBFT, DBFT, FBA, DPoS+BFT, Raft	Scalability, complexity, cost effectiveness, energy efficiency
Lepore et al. [22]	PoW, PoS, Pure PoS	Throughput, scalability

3. Popular Consensus Protocols

Consensus protocols can be considered the backbone of a blockchain network as they enforce consistency and integrity, leading to tamper-proof and immutable properties. In this section, we cover some well-known consensus protocols for the benefit of newer researchers in the blockchain space. These protocols fall within two broadly defined groups: proof-based and voting-based protocols. Proof-based protocols comprise Proof of Work (PoW), delayed Proof of Work (dPoW), Proof of Importance (PoI), Proof of Elapsed Time (PoET), Proof of Capacity (PoC), Proof of Authority (PoA) and Proof of Burn (PoBr) whereas

voting-based protocols include Proof of Stake (PoS), Delegated Proof of Stake (DPoS), Ripple Protocol (RP), Practical Byzantine Fault Tolerance (PBFT), delegated Byzantine Fault Tolerance (DBFT) and Federated Byzantine Agreement (FBA). A summary of these protocols is available in Table 2.

3.1. Proof of Work

One of the most widely adopted consensus protocols since the introduction of Bitcoin in 2009 is PoW [23]. In PoW, network actors or participants use computational power to win the right of adding new blocks to the blockchain. In the event that a new block is detected and added to the network, the node that discovered the block is entitled to receive the predefined reward or transaction fees [24]. Computational power is expended to solve a hash puzzle, the solution of which is included in the resulting block. The process of solving the puzzle is computationally intensive and stochastic in nature. Finding a solution is likened to a miner discovering gold; hence, the hashing process is referred to as mining and the node performing it is known as a miner [25]. Any willing and interested entity is able to participate in mining utilizing its own computing power, which leads to PoW consuming an immense amount of energy [26]. For example, the energy required to run the Bitcoin and Ethereum networks is equivalent to powering 3.5 million and 1 million households, respectively [12]. Apart from the energy consumption, other drawbacks of PoW include low throughput and decentralization. For an adversary to circumvent the PoW protocol and disrupt the consensus, it must amass accumulated superior computing assets compared to the combined computing asset of all honest mining entities. This is referred to as the 51% attack [27]. Many believe that it is not economically feasible to execute this attack over a blockchain network; however, the advent of mining pools and the use of application-specific integrated circuits (ASICs) have shown otherwise. Bitcoin, Ethereum, Litecoin and Dogecoin are cryptocurrencies based on PoW.

3.2. Delayed Proof of Work

The dPoW consensus protocol is used by Komodo, a multi-chain platform [28]. As the name implies, a multi-chain platform leverages the security provided by a secondary blockchain (in this case, the security from solving hash puzzles in PoW) to secure blocks in the main blockchain. This process is facilitated by 64 notary nodes elected yearly, whose purpose is to write to the PoW blockchain. By doing so, dPoW avoids additional energy consumption and overhead costs. The dPoW consensus protocol utilizes the assigned PoW blockchain to save the Komodo transactions [28]. Apart from being cost effective, dPoW is also resilient against the 51% attack, as an adversary must attack both the main and secondary chains to be successful.

3.3. Proof of Stake

PoS is an alternative consensus protocol that retains the advantages of PoW while overcoming some of its weaknesses. In PoS, a participating entity must have some stake (cryptocurrency) in the system in order to mine or validate block transactions. If a participant owns 10% of the stake (coins), then the probability of such a participant mining the next block is 10%. The block proposal mechanism in the PoS consensus protocol is as follows: blocks are created by the staking nodes, also referred to as validators, which are then allowed to participate in the block creation process by staking a certain amount of currency [29]. The protocol then selects the entity or node that will mine the next block based on their stake. All participating nodes in a PoS consensus protocol network must prove ownership of a certain amount of stake locked in the network [30]. Subsequently, a pseudorandom mechanism is used to select a leader or block proposer. A committee formed by selecting nodes based on their stake locked in the network decides on the validity of the block proposed by the leader. The selected validators cast a vote to approve or reject the proposed block. The proposed block is only accepted and appended to the blockchain if the majority of the committee members cast votes of approval, usually two-thirds of the

committee size [31]. A 51% attack becomes economically infeasible, as an adversary must own 51% of the overall cryptocurrency. However, nothing-at-stake is a major drawback of PoS, where multiple chains can be voted on by block generators because there is nothing to lose. Achieving a consensus is prevented by this act, which leads to forks. Ouroboros [32], Peercoin, Gridcoin and Nxt are some examples of cryptocurrencies based on PoS.

3.4. Delegated Proof of Stake

DPoS is another voting-based consensus protocol that is derived from PoS. DPoS involves an electoral process analogous to a board of directors, whereby board members are limited in number and are elected by the populace. Additionally, they have the mandate of their electorates to exercise their rights. Apart from the amount of staked cryptocurrency, members with voting rights are picked through election and replacement [33]. Wealth staked in the protocol during voting rounds is locked in smart contracts. Transaction validation, new block creation, network operations, and maintenance are performed by the group of elected delegates. These delegates are the block producers (BPs), who are rewarded accordingly for work done. There is also a group of backup BPs who receive smaller rewards as well. For any negative action (such as collusion) or a missed turn, a BP may be voted off the list. The staked wealth or coin in the member's smart contract is frozen or confiscated as a penalty. DPoS mechanism addresses fundamental drawbacks, such as the nothing-at-stake problem, the long-range attack, and the weak subjectivity of the basic PoS system [34]. DPoS is more energy-efficient and has a high throughput (EOS produces one block every 0.5 s). One of the major drawbacks of DPoS is that it tends toward centralization, and participating members with large stakes in the network can vote themselves into becoming validators. BitShares, Steemit, EOS, Lisk and Ark are based on DPoS.

3.5. Proof of Authority

PoA is a variant of the PoS protocol [35]. In PoA, trusted nodes known as validators are selected to validate transactions and blocks. Due to the limited number of validators required, the network has high throughput, is highly scalable and has nearly zero processing fees. A validator also does not need to stake any of its assets, as with PoS, but rather its own reputation. By staking reputation (something that needs to be earned over time by participating in a network), this overcomes the main problem of PoS, where the staking of wealth leads to wealthy participants receiving most of the incentives in the network. However, PoA tends toward strong centralization, as the whole network is in the hands of a small number of actors.

3.6. Proof of Importance

PoI was introduced by NEM to grant nodes permission to create new blocks, a process referred to as harvesting [36]. Only nodes with a specified amount of cryptocurrency balance are eligible for harvesting (for NEM, this minimum balance is XEM 10,000). Nodes with less than this balance amount have a zero importance score. Nodes with zero importance scores do not stand a chance of being selected to produce blocks. This will ensure that selected nodes are only those with a reasonable amount of wealth locked in the network, and as such, will benefit from protecting the network, due to their own vested interests. Importance is taken into consideration in the harvesting process, where nodes with higher importance scores are more likely to be selected to create new blocks and receiving transaction fees. Importance is calculated based on outgoing transactions with a specified minimum value made by the node within a specific time period (for NEM, transactions of at least XEM 1000 in the last 43,200 blocks or approximately 30 days). The block producer selection process is essentially randomized and time based, where nodes need to determine if their current *hit* value (calculated from the hash of the previous block and the node's public key) falls below a *target* value (calculated from the time since the last block, the importance of the node, and current difficulty level).

3.7. Practical Byzantine Fault Tolerance

PBFT is a consensus protocol based on replication between known parties that can tolerate a failure of up to one-third of the parties [37,38]. It is an algorithm for solving a Byzantine fault resulting from a failure in achieving consensus caused by the Byzantine Generals Problem (BGP) [39]. In general, an elected leader (primary node) creates an ordered list of transactions that is broadcast to other validation nodes, who then execute them. After transactions have been executed, validation nodes compute the hash code for the new block which is then broadcast to their peers. If two thirds of the received hash codes are the same, the block is committed to the node's local copy of the blockchain. PBFT ensures network fault tolerance and allows thousands of operations per second with a negligible increase in waiting time. However, one of the major drawbacks of PBFT is the ability to practically implement the algorithm, due to the enormous amount of calculations required [30]. Tendermint [40], the Diem blockchain [41], Hashgraph [42], and Hyperledger Fabric [43] achieve consensus based on PBFT.

3.8. Ripple Protocol

The Ripple or XRP ledger consensus protocol is a Byzantine fault-tolerant consensus protocol [44,45]. A set of validators (also known as a unique node list) selected by network participants evaluate transactions. These validators should be honest nodes that are unlikely to collude with one another. When a large percentage of these validators agree upon a given set of transactions, it is included in the next ledger version. If not, these validators modify their transaction proposals to align them with other trusted validators. This may include the inclusion or removal of new transactions. This is an iterative process that is performed until consensus is reached. Consensus can be reached as long as there are at least 80% validators that are not faulty. However, if this requirement is not fulfilled, the progress entire network merely stalls rather than being susceptible to attacks.

3.9. Delegated Byzantine Fault Tolerance

DBFT is a consensus protocol analogous to a country's governance system, where there are citizens, delegates, and speakers to ensure that the country (network) is governed efficiently (functioning optimally) [46]. This protocol relies on a voting system to choose delegates and speakers, similar to PoS. Citizens (ordinary nodes) in DBFT vote for delegates (bookkeeping nodes), where each ordinary node is entitled to vote, regardless of their wealth. Speakers are then randomly chosen from these delegates. Delegates are responsible for tracking and recording citizen demands (network transactions) in the ledger. To verify a block, the randomly selected speaker proposes its block and broadcasts it to other delegates, who then match the speaker's block with theirs to verify its validity. Before the proposed block can be accepted and added to the network, at least two-thirds of the delegates must agree on it. However, if less than two-thirds of the delegates agree, a new speaker is randomly selected and the entire process restarts.

3.10. Federated Byzantine Agreement

Stellar is designed to facilitate transactions between tokens from different issuers. Its underlying consensus protocol, the Stellar Consensus Protocol (SCP) is based on the Federated Byzantine Agreement (FBA), which is a generalization of the Byzantine Agreement (BA) that supports open membership [47]. Token issuers can designate validators to enforce transaction finality, whereby validators from different token issuers must come to a consensus before anyone can commit to the transaction history.

To facilitate the consensus process, the notion of quorum slices is introduced by SCP. Quorums are subsets of nodes where each pair of quorums have non-empty intersections. In other words, each pair of quorum have at least one overlapping node. Quorums play a vital role in reaching an agreement in a distributed system, whereby state updates are performed only when there is a quorum of nodes in agreement. Traditionally, quorums are fixed and uniform but in SCP, traditional quorums cannot be applied because there may be

cases of nodes that are unaware of each other's existence. Instead, each node v can declare its own quorum slices (sets of nodes) with the following requirements:

- If all nodes in a slice are in agreement about the system state, v assumes that they are right.
- System information can be obtained by v in a timely fashion from one of the quorum slices at any given time.

The slices of a node v can contain nodes that v needs to be in agreement with. The nodes in these slices also have their own quorum slices. Agreement between nodes from different quorum slices (that are unaware of each other's existence) is then inferred via common nodes, which then form a quorum.

The SCP protocol has two phases: nomination and balloting. During the nomination phase, nodes nominate candidate values to commit to the ledger. By *echoing* the nomination of its peers, all nodes eventually converge to a particular value known as a candidate. This candidate is then tied to a ballot, which is voted on by nodes. There can be multiple ballots for multiple candidates being voted on at once. Consensus is reached when nodes can find a quorum that accepts a particular ballot.

3.11. Proof of Elapsed Time

In 2016, Intel introduced PoET as an alternate to PoW. Hyperledger's Sawtooth project (HYP16) [48] presently runs on the PoET protocol. Rather than competing based on computing resources or the staking of cryptocurrency, PoET implements a competing scheme built on a random back-off mechanism that has been previously adopted in local area networks, specifically for medium access control procedures [49]. In general, participating nodes request a randomly allocated wait time, whereby the node that was assigned the shortest waiting period is chosen to be a block leader. Apart from publishing the new block, the block leader must also provide evidence of its shortest wait time, and that it has not broadcast its block prior to the expiration of its wait time [17]. PoET is more decentralized due to the low cost of participation, which allows more entities to participate easily. It is also much easier for participating entities to verify whether a leader has been legitimately elected and that the cost of leader election is proportional to the value gained from it. However, it is not suitable for public blockchains and cannot be mass adopted because it requires the use of specialized hardware by Intel. Apart from that, it also requires trust in the third party hardware itself, which goes against one of the fundamental ideas of blockchain, which is that it should be "trustless".

3.12. Proof of Burn

Rather than investing in physical resources to perform mining operations, such as in PoW, PoBr intentionally burns cryptocurrency as a means to select block leaders [50]. Powerful computational resources are not required for the block validation process in PoBr-based networks, hence powerful mining hardware such as ASICs are not necessary. The act of burning currency can be seen as an act of purchasing a *virtual* mining rig. It allows a node to show commitment to the network by taking a short-term loss for long-term gain. Coins can be burned by sending them to a predetermined burn address, and cannot be recovered. These burn transactions result in burn hashes that are analogous to the hash values used to determine block leaders in PoW. Slimcoin is one example of a cryptocurrency that uses PoBr as its consensus mechanism [51].

3.13. Proof of Capacity

PoC is a protocol that involves allocating a non-trivial amount of memory or disk space to solve a problem presented by a service provider [52]. PoC is a piece of data that is sent by a Prover to a Verifier to show that the Prover has reserved a certain amount of space. For example, a participant requesting a particular service must reserve or dedicate a certain amount of disk space, which is then verified by a verification process. The verification process has to be timely, efficient and must consume a reasonable amount of disk space.

The reserved memory space showcases the node's commitment to the network, which ensures that the node requesting a service is genuine. In practice, it is difficult for a node requesting a service to pass the verification process if it has not reserved the memory space as claimed. PoC is often implemented using hard-to-pebble graphs. The verifying node asks the requesting node to perform labeling of a hard-to-pebble graph, to which the requesting node must commit. The verifying node prompts the requesting node to randomly open several locations in the reserved disk space as proof [53,54]. Burstcoin, Chia [55] and SpaceMint [54] are based on PoC.

Table 2. Cryptocurrencies and Consensus Protocols.

Consensus Protocol	Cryptocurrencies
PoW	Bitcoin (2009), Litecoin (2011), Namecoin (2011), Peercoin (2012), Dogecoin (2013), Primecoin (2013), Auroracoin (2014), Mazacoin (2014), Monero(2014), Dash (2014), Titcoin (2014), Verge (2014), Vertcoin (2014), Ethereum (2015), Tether (2015), Zcash (2016), Ethereum Classic (2015), Bitcoin Cash (2017)
dPoW	Komodo (2014)
PoS	Nxt (2013), Gridcoin (2013), Potcoin (2014), Steem (2014), Tezos (2014), Ouroboros (2016), Algorand (2017)
DPoS	EOS (2017)
PoA	Ethereum Kovan (2019)
RP	Ripple (2013)
PoS	Dash (2014)
POI	NEM (2014)
PBFT	Tendermint (2014), Hyperledger Fabric (2015), Diem (2020)
DBFT	NEO (2014)
FBA	Stellar (2014)
PoET	Hyperledger Sawtooth (2015)
PoBr	Slim Coin (2014)
PoC	SpaceMint (2014)

4. Alternative Protocols

In this section, we discuss 15 alternative consensus protocols that have been proposed within the past 3 years. Some of these protocols have been proposed for a specific application, such as Proof of Familiarity (PoF), Proof of Benefit (PoB), Proof of Participation and Fees (PoPF), Proof of Vote (PoV), and CHB and CHBD consensus protocols, while others are for more general-purpose use, such as Proof of Reputation (PoR), Proof of Reputation *X* (PoRX), Proof of Phone (PoP), Proof of Learning (PoL), Proof of Search (PoSe) Proof of Sincerity (PoSn), Proof of Adjourn (PoAj), Proof of Evolution (PoE), Proof of Experience (PoEx) and Proof of Accuracy (PoA). These alternative consensus protocols can be broadly categorized based on the requirement for the selection of a block publisher. Some of the alternative protocols can fall under more than one category. These categories include the following:

- Consensus protocol based on Effort or Work (CPE).
- Consensus protocol based on Wealth or Resources (CPW).
- Consensus Protocol based on Past Behavior or Reputation (CPPB).
- Consensus Protocol based on Representation (CPR).

A discussion about these categories and the alternative protocols that fall under them is provided in Section 6.

4.1. Consensus Protocols Based on Effort or Work

4.1.1. Proof of Benefit

The PoB consensus protocol was introduced to manage energy transactions for electric vehicles (EV) [56]. It can be used in permissioned or permissionless blockchain scenarios. PoB is part of an online benefit generating algorithm, ONPoB, meant to minimize power

load fluctuations when EVs are charged or discharged. The PoB consensus protocol generates a benefit number, λ, for the blockchain system based on EV charging and discharging demands. Higher λ values imply a desirable effect on minimizing power load fluctuations. In PoB, the participating nodes are the EVs (and by extension, their drivers), who are responsible for extending and maintaining the blockchain network.

Analogous to how PoW performs leadership election by finding a solution to a hash puzzle, PoB involves finding the best λ value. The process of solving the overall benefit problem is completed by each of the nodes. The PoB mechanism comprises mining preparation and mining execution rounds. In the mining preparation round, the participating nodes execute transactions from the latest winning block. During the mining round, the participating nodes solve the benefit number problem, which determines the winning block, whose transactions are executed in the subsequent round. The PoB protocol is summarized in Figure 3.

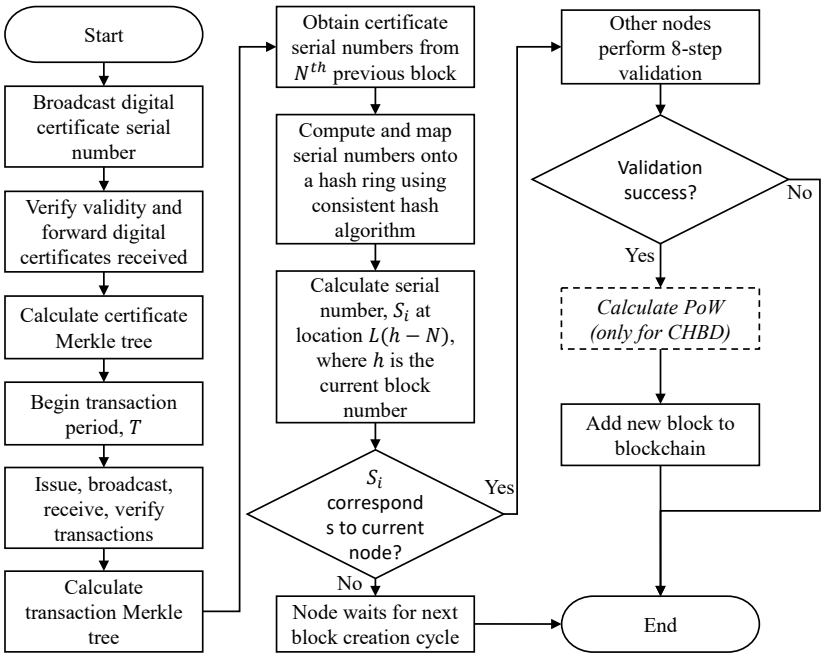

Figure 3. Proof of benefit flowchart.

4.1.2. Proof of Phone

PoP was introduced to solve the dilemma of the continually increasing operational cost of conventional blockchain platforms [57]. PoP attempts to overcome the problem of unconstrained mining competition and cost by limiting mining hardware to only smartphones. The smartphone is widely used for other purposes regardless of its mining capability, and therefore, would be financially infeasible for a single miner or a group of miners to acquire enough devices to dominate consensus. In addition, the possession of hundreds or thousands of smartphones by a single individual is of no significant benefit, albeit for normal usage or mining. This is referred to by the designers as a High entry cost and Low Operation cost (HELO) concept. The implementation of PoP requires specialized hardware, which the designers refer to as the authenticated mining unit (AMU), whose main purpose is to impose a constraint on competition and cost. It consists of four submodules which are as follows:

- Block control unit (BCU): Verify integrity of blocks and transactions.
- Micro mining accelerator (MMA): Perform hashing operations.
- Data transceiver: Communicate with peers.

- Key authentication unit (KAU): Contains device-specific, system identifiers in a one-time programmable memory for authentication purposes (enforces one AMU per smartphone rule).

The KAU within the AMU ensures that an individual or group cannot amass AMUs without paying the cost of entry (i.e., without acquiring new smartphones). The designers also recommend limiting the lifetime of the AMU module to limit the increase of mining devices, based on the assumption that new smartphone products are typically released annually. The overall mining procedure is summarized in Figure 4.

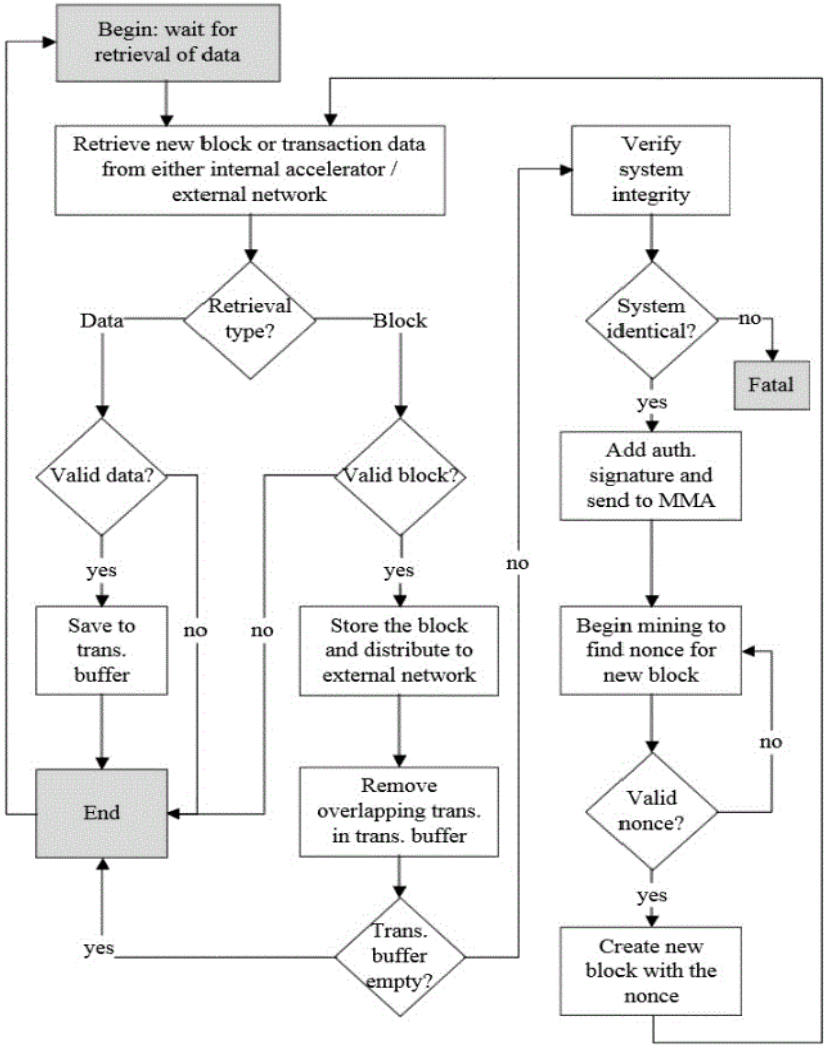

Figure 4. Proof of phone flowchart (adapted from [57]).

4.1.3. Proof of Learning

The PoL consensus protocol is a protocol that allows machine learning problems or general problems to be solved by a participating node to validate a block in a network [58]. A repository of these solutions is saved for usage by other users. The energy that is involved in the validation process in PoL is focused on solving useful tasks, thereby creating an open repository of machine learning models and data sets. PoL is currently used to validate transactions in WekaCoin. A WekaCoin blockchain, B is a list of blocks where each block b_i consists of three major elements: transaction data, a hash pointer to the previous block, and metadata about the machine learning competition used to validate the block. Three actors are involved in the validation process:

- Suppliers: Nodes who host machine learning competitions.
- Trainers: Nodes who train and submit models for machine learning tasks.
- Validators: Nodes who evaluate the machine learning models, form a consensus and propose new blocks.

The glaring advantage of PoL over PoW is that the energy expended is directed toward a meaningful endeavor, rather than just for the sake of solving a hash puzzle. It leads to an open database of machine learning solutions to various problems that can prove useful to the community. However, this also raises the concern of ensuring a continuous supply of machine learning tasks to validate transactions at an acceptable rate. The workflow of PoL is summarized in Figure 5.

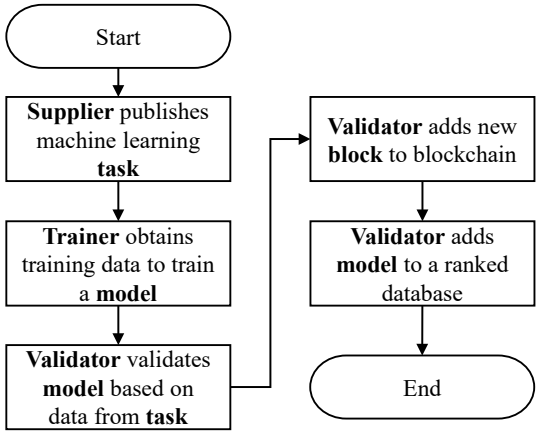

Figure 5. Proof of Learning flowchart.

4.1.4. Proof of Sincerity

PoSn relies on how *sincere* a participating node has been in a blockchain network to select mining nodes [59]. PoSn introduces a new parameter known as sincerity as a replacement for difficulty in PoW. Sincerity is used to scale mining rewards such that miners with smaller computing power are still able to participate and get rewarded, even in the presence of miners with ASICs or powerful GPUs. Sincerity essentially behaves like difficulty, as it determines the number of leading zeroes in the hash code that needs to be discovered. However, rather than being adjusted to enforce a specific block rate, participants can select their own sincerity level based on their computing capabilities. Rewards are then calculated based on the ratio of the participants' sincerity to the successful miner's sincerity level.

This simple modification of the difficulty notion from PoW provides great benefit to participants involved in PoSn. Miners with weaker computing devices can choose a lower sincerity level to ensure that they are still rewarded and increase their chances of solving the hash puzzle alongside miners with powerful mining rigs who can choose a higher sincerity level for bigger rewards. This results in a more balanced reward system, thus eliminating the imbalance of the wealth often caused by mining domination by a small number of huge resource owners. Indirectly, this encourages a larger number of individuals to participate in the validation process, which in turn enforces the security of the overall blockchain.

4.1.5. Proof of Accuracy

PoA attempts to leverage the strengths of multiple consensus algorithms, such as PoW and PoC [60]. The ideas presented by the designers were mainly theoretical in nature without concrete implementations. In PoA, the task of deciding the block leader is not only based on computing a problem with a certain computational complexity threshold (such as in PoW), but must also involve proof that the node has possession of data required to compute the problem to a certain accuracy threshold. This input data must be incomplete

and rigorously defined such that the effort to collect them is entirely stochastic in nature. This ensures that it is difficult to solve the problem with a certain accuracy within a certain amount of time.

To facilitate the stochastic nature of data collection, the required input data must be randomly distributed and stored in various locations (e.g., memory locations or network addresses). Participants then compete for access to these locations to gather the data needed to solve the mining problem. To be selected as the block proposer, nodes need to prove that they have accessed these locations, have the required data in their possession and solve the mining problem to a certain accuracy threshold. By imposing the need to compete for access to the data locations, even participants with weaker computing capabilities have a chance to be selected as block proposers. The designers suggested the use of optimal algorithms as the mining problem.

A simple example provided by the designers involves generating a random number, R, for which a hash value $H(R) = h$ is computed. R is then decomposed into k parts using a (k, n) secret sharing scheme [61] and distributed to random network addresses. Some of these addresses do not contain any information. Participants then need to collect all k parts to recover the value of R, which can then be used to recover h. They also need to prove that they have visited all the IP addresses that contain the k parts. This simple example can be extended to using random memory locations (similar to PoC) and also include decoy parts to further increase the difficulty of collecting the required data. The entire process is shown in Figure 6.

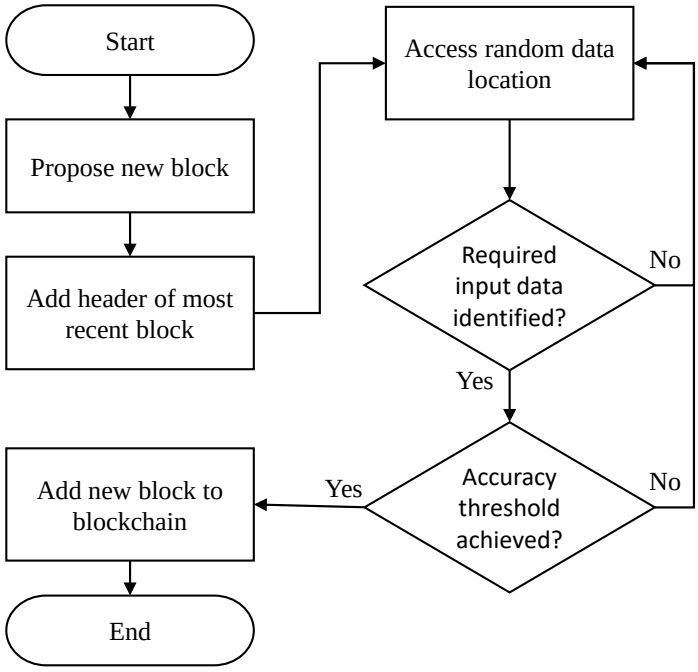

Figure 6. Proof of Accuracy flowchart.

4.1.6. Proof of Adjourn

Proof of Adjourn (PoAj) is a consensus protocol that adjourns participating nodes from all activities by introducing an Adjourn Period (AP) [62]. The adjourn period ensures that an attacker or malicious mining pool would not be able to construct their own chain of falsified blocks because nodes that broadcast more than one block within the adjourn period receive a penalty.

Block Verification time (eBVT) and Block Sorting Time (BST) are what makes up a total AP. Any broadcast block is denoted as the initial block (IB). The IB is only confirmed after the AP has completed successfully. The AP involves checking the legitimacy of n number of IBs before adding the verified block to the blockchain. During this entire period, no mining

is performed. Upon completion of the AP, the mining process resumes. The first miner to solve the mathematical puzzle (hash function) in PoAj and broadcast the solution to the network enters the AP along with other nodes that acknowledge the broadcast solution.

In the case that an IB fails to meet the condition checks of Phase 1 of the AP, its proposer is imposed a penalty and the network's mining process resumes. Participating nodes that broadcast more than one block within this timeframe do not have these blocks included in the blockchain, and the nodes are restricted from mining activities for a certain period. Unlike PoW, PoAj does not adopt the longest-chain rule to verify the genuineness of the chain. Rather, it imposes an AP to regulate block verification, which is completed at Phase 1 of the AP. In addition, transactions in PoAj are confirmed after just one confirmation, unlike the six-confirmation waiting period in PoW, making PoAj a consensus protocol with a much faster transaction confirmation rate compared to many mainstream and alternative consensus protocols The overall PoAj flowchart is as shown in Figure 7.

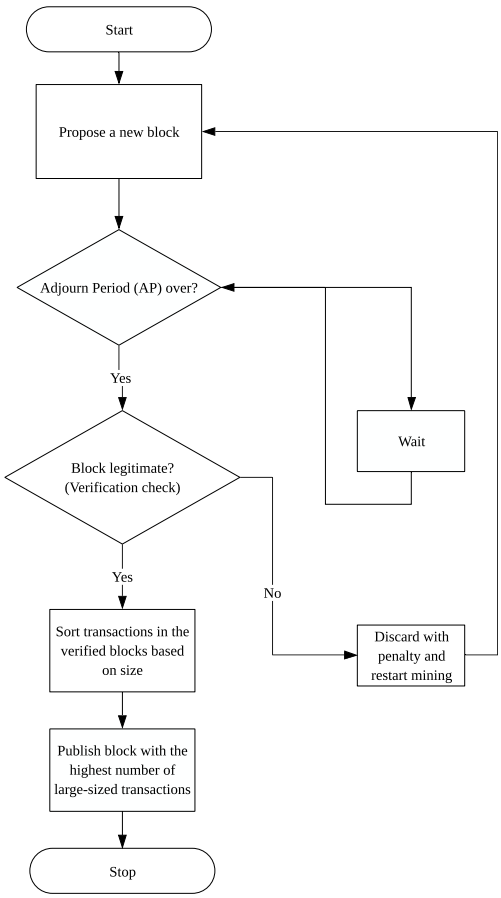

Figure 7. Proof of Adjourn flowchart.

4.1.7. Proof of Search

Similar to PoW, Proof of Search (PoSe) selects block leaders based on the computational effort expended. However, the computational power is used to search for solutions to optimization problems rather than just solving hash puzzles [63]. Nodes known as clients submit requests to search for solutions to these optimization problems. These requests are known as jobs. A job is essentially the data representation of these requests, and consists of all the relevant information required to perform the search including the evaluator, a deterministic computer program that calculates the fitness of each solution.

In a PoSe-based blockchain, any node can become a client by submitting a job to the system. The job submission includes the client's ID and the evaluator for the given problem. An example provided by the PoSe designers is where a client implements an evaluator

for the traveling salesman problem. The evaluator receives the order of visited cities as an input and returns the total length traveled as an output. The client then compensates for the node that discovered the best approximate solution to its problem. In comparison to PoW, the results show that PoSe has a lower likelihood of forks occurring and lower variations in block production time as compared to PoW. The overall PoSe flowchart is as shown in Figure 8.

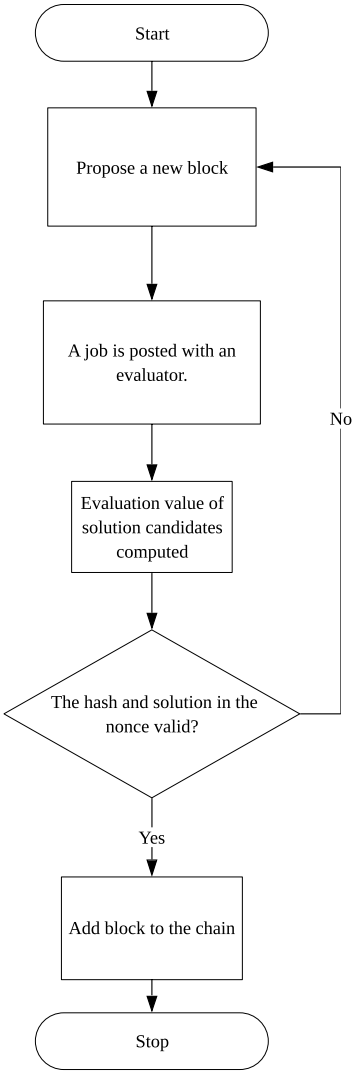

Figure 8. Proof of Search flowchart.

4.1.8. Proof of Evolution

Proof of Evolution (PoE) shares many similarities with PoSe, as it also performs a search for solutions to problems by running genetic algorithms (GAs) [64]. Thus, computational power is also channeled toward a beneficial cause. In PoE, a GA is made up of Individuals (possible solutions to a given problem), Chromosomes (encoded representations of Individuals), Population (a set of Individuals), Genes (the smallest components of each Chromosome), and Fitness (a function to evaluate how well Individuals solve a problem). PoE maintains some of the properties of PoW and PoSe. Solving hard puzzles similar to PoW is still present, and this provides a probabilistic guarantee that a large number of nodes has been evaluated. A client first submits a Job along with a Searcher, the GA that will be executed during the mining process. Interested miners can attempt to solve the problem by running the Searcher to generate solutions.

This protocol includes a component that keeps track of the number of solution candidates for each Job, which is kept up to date by the Searcher. Solutions in the population can also be updated as part of a collaborative effort between miners. PoE uses mini-blocks similar to PoSe. There is also an Evaluator, an algorithm defined by the client that assigns scores to each possible solution of the GA while a validator node checks the transactions and Jobs similar to PoSe. PoE encourages miners to share their current best-found solutions, leading to cooperation among participating nodes. This collaborative effort leads to better solutions to the problems that the GAs are trying to solve. The node that finds the best solution is rewarded by the client. This discourages the client from re-posting the same Job for which the client already has a solution. Mining in PoE is similar to PoW whereby a Job is selected from a list of existing Jobs, and then miners attempt to find solutions by executing GAs repeatedly for multiple iterations. Verification is performed by checking hash values and the solution validity in the nonce, which is then passed to the publisher if the block is valid. The overall PoE flowchart is as shown in Figure 9.

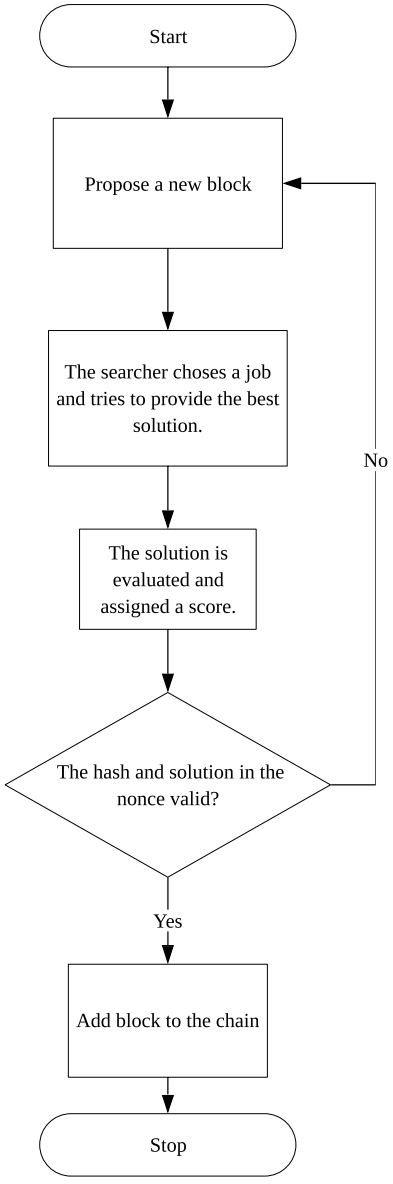

Figure 9. Proof of Evolution flowchart.

4.1.9. Proof of Experience

PoEx is a consensus protocol that uses prior PoW done by a node over time (which would otherwise be wasted if the node is unsuccessful in publishing blocks) to reduce its mining difficulty [65]. It uses the same unspent transaction output (UTXO) model, digital signature (based on elliptic curve), and mining mechanism as Bitcoin.

The difficulty of finding a valid nonce in PoEx differs from Bitcoin because there is a global difficulty level (global target) that dictates the mining difficulty for all nodes and a local difficulty level (local target) that is unique to each node. When a node mines for the very first time, its local target is equal to the global target. However, a miner can reduce its local target by using previously accumulated Proof of Work done and positively contributing to the network.

A miner starts with the global target when it mines for the first time. The more "experienced" the miner becomes, its local target reduces and the difficulty is lowered. So each miner, in effect, creates its own local target with every block it builds with regard to the previous PoW done. Whenever a miner broadcasts a block, its local target is recalculated and verified by other participating nodes. Unlike PoW, PoEx blocks also enclose a pool of nonces, a local target, and the unique identifier of the miner that computed the block. Unpublished blocks also have an ordered set of nonces that miners can use to recalculate the PoW of the preceding blocks. Therefore, the local target can be computed as a function of the newly obtained hashes from these unpublished blocks. The remaining components of a PoW consensus protocol are not altered. These changes are essential for simplifying the computation of the local target.

Compared to PoW, PoEx increased the size of the nonce to 64 bits so as to increase the number of possibilities and stop miners from the likelihood of being constrained by the number of possible nonces. The overall PoEx flowchart is as shown in Figure 10.

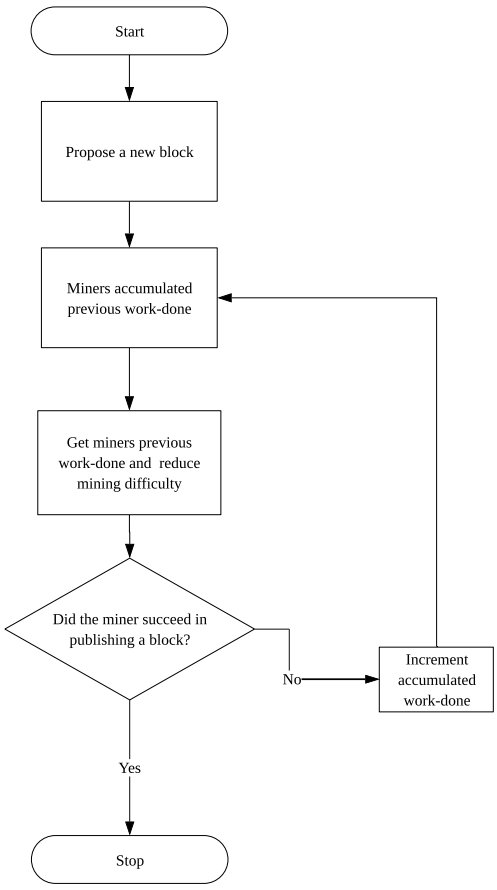

Figure 10. Proof of Experience flowchart.

4.2. Consensus Protocols Based on Wealth or Resources

4.2.1. Proof of Participation and Fees

PoPF was proposed for a new blockchain-based distributed ledger known as JCLedger, which facilitates the exchange of cloud resources [66]. It is essentially a modification of the PoW protocol to improve computational cost and efficiency without degrading security. In PoPF, a block leader is selected for every block based on a ranking that is determined by the node's participation (by participating in the mining process) and fees (what a participating node pays as fees). Block leaders in PoPF are referred to as accountant nodes. The top n% ranked users are selected as the accountant for each block. Each participating node competes for the right to produce the next block by solving a hash puzzle, similar to PoW. However, unlike PoW, there is a difference in difficulty for each participating node. This mining difficulty is adjusted based on the ranking list such that it is easier for higher-ranked participants, compared to lower-ranked ones, to solve the puzzle.

The PoPF algorithm needs a certain number of users (if the number of accountant nodes is set to n, then PoPF needs at least n users) as a condition to run. In the beginning, PoPF cannot be run due to this requirement. As such, the original PoW protocol is used initially until the condition is satisfied. With every new block, participating nodes checks if the PoPF operating conditions have been met, upon which all nodes switch to PoPF. The entire flow of the PoPF protocol is shown in Figure 11.

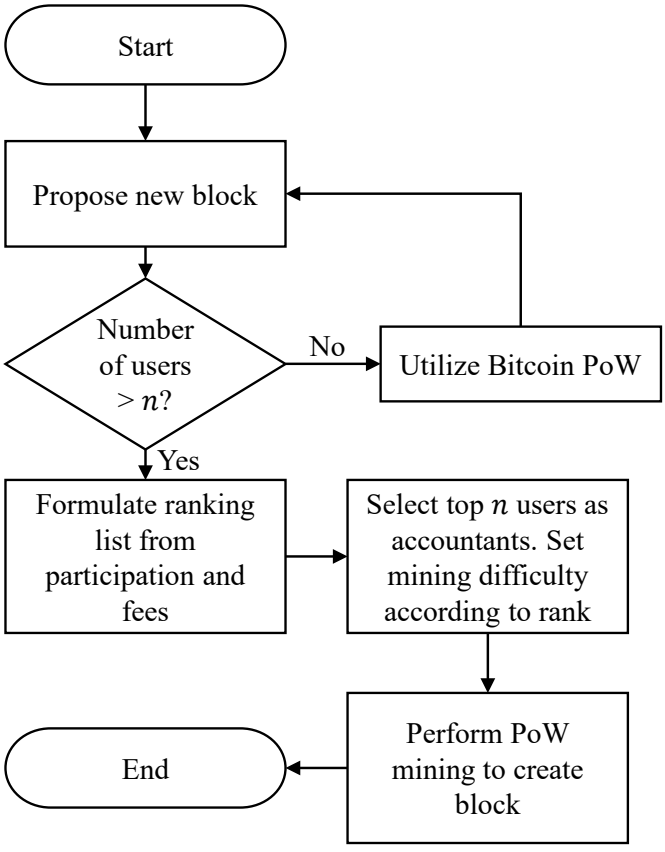

Figure 11. Proof of Participation and Fees flowchart.

The ranking of a node, x, is calculated based on its fees paid, $F(x)$, and the number of times it has been selected as an accountant, $M(x)$, as follows:

$$R(x) = \frac{F(x)}{M(x)+1} \tag{1}$$

which is then used to determine its mining difficulty, $D(x)$. The more times a node is selected as an accountant, the lower its rank. This makes the selection process more balanced without favoring nodes with powerful computing capabilities. Nodes within the same ranking area should have similar mining difficulties. The size of the ranking area and the number of accountant candidates can be determined based on the number of transactions in the network. For example, the top 10% of the accountants have lower difficulty, compared to the subsequent 10% of accountants.

4.3. Consensus Protocols Based on Past Behavior or Reputation

4.3.1. Proof of Familiarity

PoF is an off-chain collaborative consortium consensus protocol, where a partial information storage technique is adopted [67]. This leads to smaller block sizes as compared to PoW and other protocols. It was implemented for a collaborative medical decision-making scheme with four major entities, which are considered nodes in a consortium blockchain network:

- Patient, P;
- Recovered patient P_R;
- Doctor, D;
- Insurance company, I_c.

Decisions are made based on an individual's level of expertise, experience and success rate. It was established that D, I_c, and P_R who gain more experience over a longer period of time make better decisions on issues in the domain. Additionally, D, I_c, and P_R that previously made accurate decisions in a domain have a higher chance of also making more accurate decisions in future collaborations. If D, I_c and P_R provide a medical decision or opinion, the best decision maker is decided on based on both experience and accuracy in decision making. To numerically represent individual qualitative achievements, the individual familiarity index (IFI) is calculated. IFI has a range of $[0.0, 1.0]$. More importantly, IFI is ranked and updated, according to the performance history of D, I_c and P_R stored in the local database and connected blockchain database. IFI index of PoF is dependent on the following factors:

1. The doctor's judgment: To rate a doctor's decision, factors such as job experience time (JET) and treatment success rate (TSR) are considered. The IFI of a doctor IFI_D is represented as $IFI_D = \{JET, TSR\}$ (job experience time and treatment success rate).
2. The perspective of a recovered patient: To evaluate the perspective of a recovered patient, factors such as treatment experience time (TET), current condition (CC), and experience of disease (ED) of recovered patients are considered. The IFI of a recovered patient IFI_{P_R} is represented as $IFI_{P_R} = \{TET, CC, ED\}$ (treatment experience time, current condition and experience of disease).
3. Insurance company's perception: The perception of an insurance company is a significant aspect of collaborative medical decision making. IFI of an insurance company IFI_{I_c} is calculated from the settlement time (ST) and cover amount (CA). IFI_{I_c} is represented as $IFI_{I_c} = \{ST, CA\}$ (settlement time and amount covered by the insurance).

To make a final decision in the collaborative medical decision scheme, PoF calculates the final collaborative medical decision, C_d. This takes into consideration IFI_{I_c}, IFI_{P_R}, and IFI_D of the three entities in the domain, where the best option is selected from each of the entities, D, I_c, and P_R. C_d is decided for P after considering the winning outcomes of the three entities. The entire process of PoF is summarized in Figure 12, where PII denotes personally identifiable information.

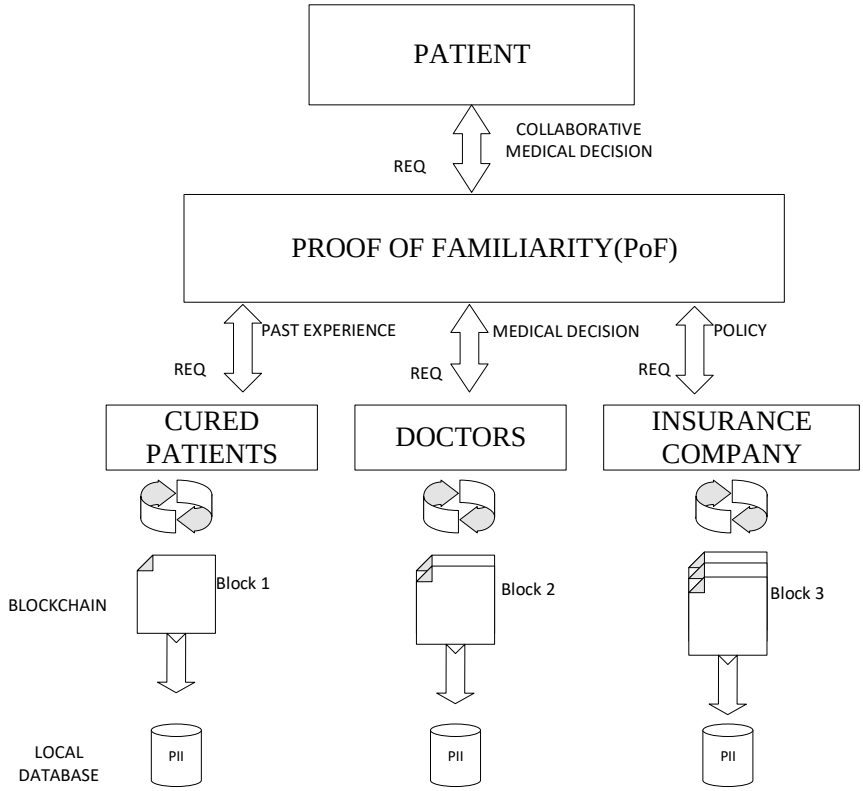

Figure 12. Proof of Familiarity flowchart.

4.3.2. Proof of Reputation

In PoR, trust and reputation are the incentives to enforce desirable behavior in protocol participants [68]. Participants who publish blocks in a timely fashion when selected are rewarded with trustworthiness, which then leads to reputation for these participating nodes. Similar to other protocols where participants or miners are incentivized with network coins for block creation or good conduct, PoR rewards participants with trust, which is non-transferable.

Trust is a measure of reputation. Participating nodes with consistently high values of trustworthiness can provide better and more stable services than those without. In contrast to PoW where nodes that solve the hash puzzle within the shortest amount of time are selected to publish a block, PoR selects participating nodes with the highest reputation or trust value in the group. By publishing a block, the selected node automatically increases the overall rank of its reputation. The reputation of sincere validators are also increased.

Alteration of a blockchain on PoW would require computing power amounting to 51% of the total computing power. Although this seems infeasible for the time being, the existence of large mining pools implies that it could be a viable threat in the future. To avoid this problem, the public keys of all participants in PoR are stored locally. Every participating node has a local copy of all the public keys, making it difficult for any malicious node to attempt to falsify identities without been detected instantly. The PoR protocol has three steps:

1. Broadcasting transactions: A service requestor records the rate of the service via feedback at the end of each interaction. This message is broadcast along with its signature to other nodes who then verify and store them in memory.
2. Building blocks: Nodes receive transactions until a certain threshold. Upon hitting this threshold, the node stops receiving transactions and ranks each service provider based on this set of transactions. If the current node happens to be the highest-ranked service provider, it constructs and publishes a block, signed with its private key.

3. Verifying blocks: The block is appended to the blockchain after verifying that the sender is truly the most reputable node. This is performed by all nodes receiving the block, who also verify transaction signatures using the signer's public key. If verification is successful, the block is included in the blockchain.

PoR is a consensus protocol for permissioned blockchains with a few traits suitable for consortium blockchains. It can be a component suitable for transactional applications that require making decisions based on the reputation of the nodes. The overall flow of the PoR protocol is shown in Figure 13.

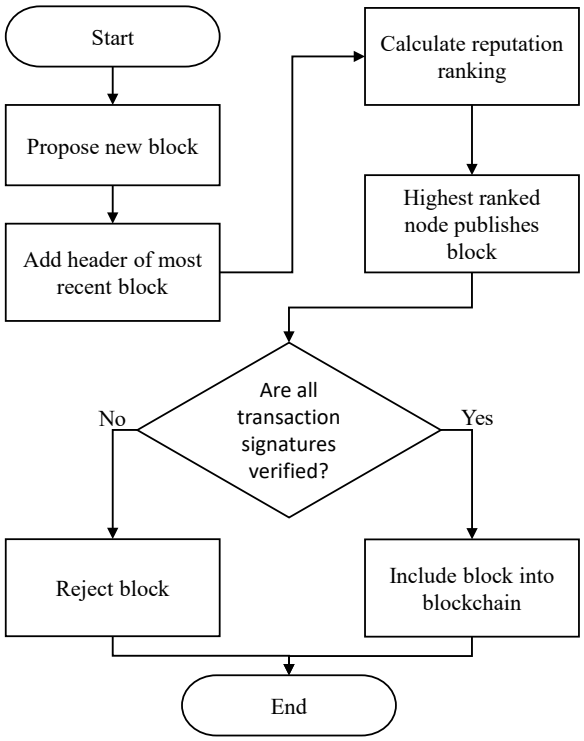

Figure 13. Proof of Reputation flowchart.

4.3.3. Proof of Reputation X

PoRX integrates a reputation component to other PoX protocols, such as PoW [69]. Nodes with good conduct and behavior history are selected based on their accumulated reputation resource. These resources, earned through good participatory conduct, are stored in the reputation module that helps the system reduce puzzle-solving difficulties for the underlying PoX protocols, and eliminates the risk of centralization from the use of ASIC and mining pools. In PoRX, nodes with higher reputation have lower mining difficulty as compared to nodes with a lower reputation. After a successful mining round, a node is rewarded with more reputation along with mining fees. However, if a node fails to produce a block within a stipulated time, it faces a reduction in reputation, which is reflected in the reputation module. This reputation also decays over time.

Proof of X-repute (PoX-R) is another reputation-based consensus protocol that combines PoX with a reputation layer [70]. Here, we use PoX-R to refer to this protocol, rather than POXR, to clearly differentiate between the two. Both PoX-R and PoRx were actually developed by the same team of designers and as such, have similar properties. PoX-R also implements a mainstream protocol, as does PoW, with a reputation layer that rewards good behavior and punishes nodes that depict negative behavior. Each participating node is assigned a reputation value that stores reputation scores generated from its behavior. Nodes with positive contributions and good behavior are rewarded with increased reputation scores, while those that act maliciously or are unreliable have their reputation scores reduced. Similar to PoRX, nodes with higher reputation values are rewarded with reduced

difficulty in the mining process of the chosen PoX. Participating nodes that contribute positively to the system end up publishing more blocks than nodes that do not. PoX-R improves the overall efficiency of the PoX consensus protocol, resists stronger attacks, and provides users with lower computing power a greater opportunity to participate in the mining process. The experimental results show that the reputation-based consensus protocol has advantages in terms of security and that its resistance against attacks is improved as compared to mainstream protocols. The overall PoX-R flowchart is as shown in Figure 14.

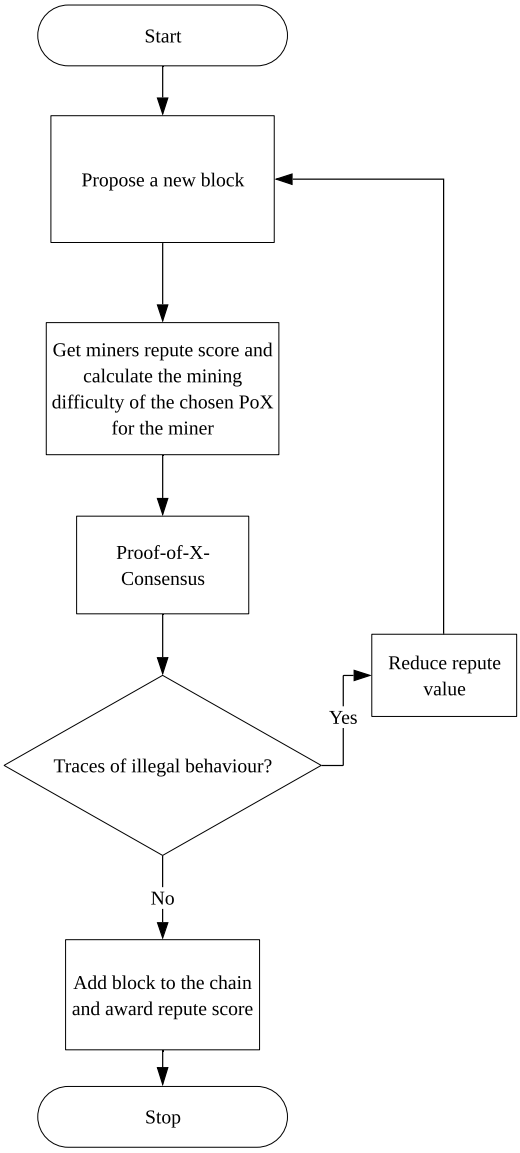

Figure 14. Proof of X-repute flowchart.

4.4. Consensus Protocol Based on Representation

4.4.1. Proof of Vote

PoV is a consensus protocol proposed for consortium blockchains [71]. Consortium blockchains are maintained by alliances consisting of nodes representing different organizations or institutions rather than individual users. To enforce decentralization, PoV separates voting and execution rights. Voting is performed by nodes known as Commissioners, whereas nodes known as Butlers produce blocks. Block producers are selected at random from the list of Butlers, thus eradicating power wastage. As depicted in Figure 15, the four main entities of PoV are as follows:

- Commissioners: Commissioners are chosen from the consortium members, represented by a working node. A Commissioner has the power to recommend, vote and evaluate the Butlers in addition to the obligation of verifying and forwarding both transactions. All Commissioners are considered to be of equal status. Every block generated in the blockchain network is sent to and verified by all Commissioners. A block is marked as valid and be added to the blockchain if it receives at least 51% of the votes.
- Butlers: Blocks are produced by Butler nodes, which are limited in number. Butlers are analogous to miners in PoW but rather than competing to be a block producer based on computational capability, they take turns to be appointed randomly. Butlers are in charge of gathering transaction data, packing them into blocks and signing them. They are then rewarded for their efforts, taken from an alliance fund that is supplemented by commissioners. Butlers are elected by commissioners from the list of Butler candidates. After the tenure cycle is over, Butlers can accept re-election. It is possible for a node to be both a Commissioner and a Butler at the same time.
- Butler candidates: A Butler is elected from Butler candidates based on votes by Commissioners who vote to elect the candidates. In the advent of a loss in the election, they can stay online and wait for the next election. Butler candidates are scored based on their performance during their tenure as a Butler. This score is taken into consideration when voting for new Butlers. Three mandatory steps are required to apply to be a Butler candidate:
  1. Register a user account and submit an application.
  2. Submit a recommendation cryptographically signed by at least one Commissioner.
  3. Submit deposit, which is used to enforce good behavior.
- Ordinary users: Ordinary users can join or exit the network anytime without being authorized, and their behavior can be arbitrary. Ordinary users can only be part of block distribution and message forwarding that are not part of block generation unless they apply to become Butlers. The entire consensus protocol is visible to ordinary users.

The designers of PoV show that their consensus protocol guarantees security, transaction finality, and prevents blockchain forking, all while minimizing power consumption.

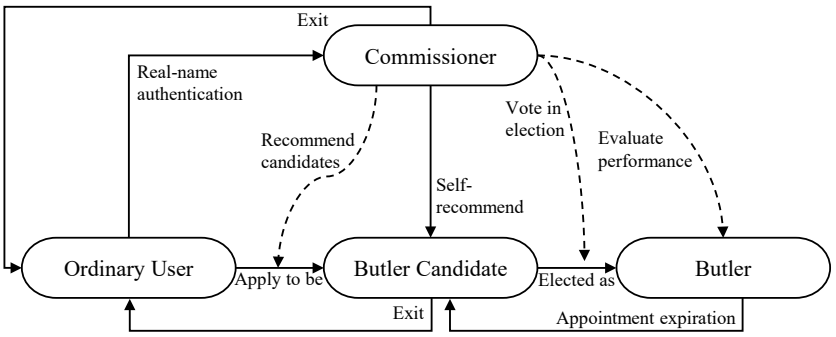

Figure 15. Relationship between roles in Proof of Vote (adapted from [71]).

4.4.2. CHB and CHBD

Rather than using a random public–private key, the CHB consensus protocol uses unique public–private key pairs issued by a certification authority (CA) [72]. Nodes do not need to use computational power to compete to create blocks but are selected randomly in the form of gambling based on the pseudorandom property of the consistent hash algorithm. The only computational power required is for validating blocks, data storage and message passing. The entire process of the CHB protocol can be summarized as follows:

1. Nodes broadcast their digital certificate and verify the validity of all digital certificates broadcast by their peers. These certificates are then hashed and included in a digital certificate Merkle tree.
2. Nodes can then issue and broadcast transactions to their peers. All nodes collect transactions, verify them, and forward them. These transactions are also hashed and included in a transaction Merkle tree.
3. At the end of the transaction period, one of the nodes are selected at random based on the consistent hash algorithm and digital certificate serial numbers from one of the previous blocks in the blockchain. The chosen node receives tokens and is granted the privilege of creating the new block.
4. Other nodes validate the new block and include it into their copy of the blockchain. The validation process also includes a check to ensure that the same node cannot be selected consecutively as the block leader.

Note that in the proposed protocol, the first N blocks only consist of digital certificate serial sets and must be generated using another consensus protocol, such as PoW. Due to the use of a CA, digital certificates and a random selection process based on the consistent hash algorithm, malicious nodes who attempt to perform a double-spending attack would need to expend an enormous amount of computational power (as compared to regular nodes) to rebuild the chain. However, this difficulty level is fixed and is still lower than in PoW.

This is addressed in a modified variant of the CHB protocol, known as the CHBD protocol (the D denotes difficulty), where a PoW needs to be computed for the block. This can either be computed solely by the selected node or assisted by a mining pool consisting of all other nodes. Because PoW is implemented in a non-competitive setting, the overall energy consumption is lower than PoW. The difficulty of the entire mining process (consistent hash and regular hashing) can be adjusted to match the difficulty of PoW. The designers mentioned that omitting certain validation steps (such as the check for consecutive block leaders) can lead to better performance but sacrifices resistance against the 51% attack. The overall CHB/CHBD flowchart is as shown in Figure 16.

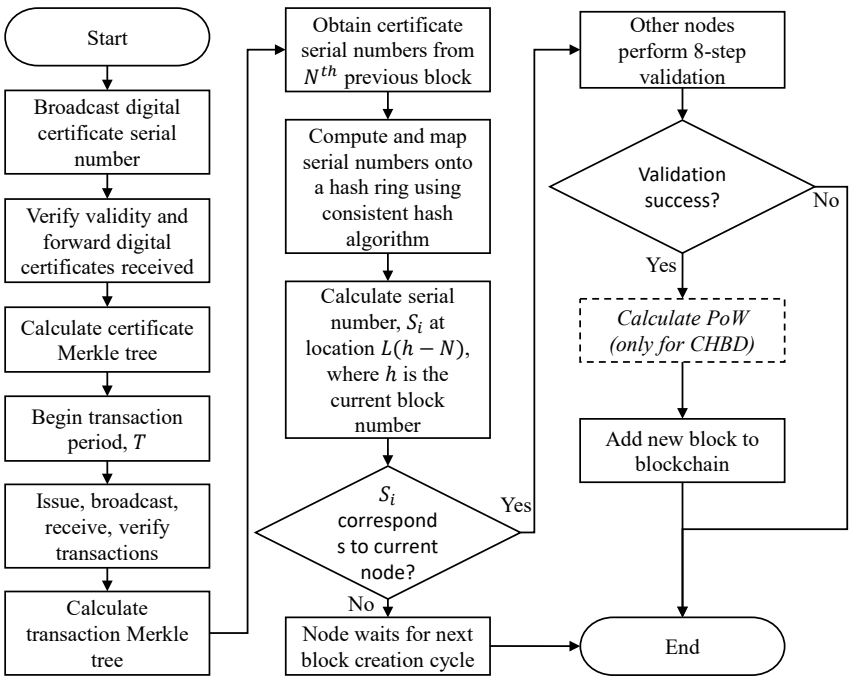

Figure 16. CHB/CHBD flowchart.

5. Evaluation of Alternative Protocols

In this section, we evaluate the alternative consensus protocols based on the five key metrics we have identified during the review process. In our evaluation, we benchmark the consensus protocols against PoW. These metrics are first introduced in Section 5.1 before the in-depth comparison is performed in Section 5.2. Table 3 provides a summary of the comparisons made in this section. Note that the information in Table 3 is based on the experimental results, findings or estimates obtained from the protocols' respective publications. We, again, wish to remind the reader that these protocols should not be adopted without further experimental analysis or simulation and that this comparison is made solely for academic purposes.

Table 3. Comparison of alternative consensus protocols benchmarked against PoW.

Protocols	Energy Consumption	Scalability	Finality	Tolerated Adversary	Throughput (Tps)
PoF	Low	High	Absolute	≤75%	Very High
PoR	Low	High	Probabilistic	Instant Detection	High
PoRX	Low	PoX-dependent	Probabilistic	>50%	PoX-dependent
PoX-R	PoX-dependent	PoX-dependent	PoX-dependent	≤50%	PoX-dependent
PoB	High	Moderate	Probabilistic	≤50%	High
PoP	High	Moderate	Probabilistic	≤50%	Low
PoL	Moderate	Low	Probabilistic	≤50%	Low
PoSn	High	Moderate	Probabilistic	≤50%	Moderate
PoA	Low	Moderate	Probabilistic	-	High
PoPF	Moderate	High	Probabilistic	≤50%	Moderate
PoV	Low	High	Probabilistic	≤50%	Moderate
CHB *	Low	Low/Moderate	Probabilistic	>50%/≤50%	Low/High
CHBD *	Low	Low/Moderate	Probabilistic	>50%/≤50%	Low/Moderate
PoAj	High	Moderate	Probabilistic	≤50%	High
PoE	High	Moderate	Probabilistic	≤50%	Moderate
PoEx	High	Moderate	Probabilistic	≤50%	Moderate
PoSe	High	Moderate	Probabilistic	≤50%	Moderate

* with/without full validation steps.

5.1. Evaluation Metrics

Throughput reflects upon the number of blocks that can be verified and deployed to the blockchain per second. This can be quantified as the number of processed transactions per unit of time, denoted as transactions per second (Tps) [24]. The efficiency of a blockchain-based system is largely dependent on the type of consensus protocol in use, which determines how nodes communicate with one another to validate and append data to blocks, and to ensure consistency with the replicated copies of the ledger. For example, Bitcoin has a throughput of approximately 7 Tps [73]. Other factors that affect the throughput of a protocol is the design, size of data and scope of the blockchain.

Scalability refers to the ability of a computing process to be used in a wide range of capabilities. In the context of blockchain, it refers to the ability of a network to sustain a higher number of occurrences, users, or data [67]. Blockchains scale best with lightweight metadata, provenance, transaction and audit information. On the other hand, it is difficult to scale with the addition of large data types such as images or full genomic data sets. A highly scalable consensus protocol is one where the overall performance does not decline or increase with the addition of nodes.

Security denotes how well a consensus protocol resists various attacks, such as the distributed denial of service (DDoS) [74], P + epsilon [75], Sybil [9], long range attack [76] and the 51% attack. In our comparison, we only take into consideration the 51% attack, as it is a general attack applicable to a wide range of protocols. The 51% attack occurs when an attacker or a group of attackers is in possession of 51% of the resources used to determine block leadership, such as computational power or wealth. An attacker can then create an isolated chain that exceeds the length of the honest chain. This is then presented to the

network to be established as the genuine chain. This results in a fork in the blockchain that allows attackers to double spend [27]. In our comparison, we use the term tolerated adversary, which defines the percentage of nodes permitted to be malicious in a network or the number of resources that is allowed to be controlled by these malicious nodes while maintaining the network.

Energy consumption is one of the key issues of blockchain that limits its widespread application. It is estimated that Bitcoin alone has an annualized energy consumption of 132.89 terawatt-hours (as of 29 June 2021), which is comparable to Argentina and exceeds countries such as Norway and the United Arab Emirates [77]. Another cryptocurrency, Peercoin, also has large energy consumption, approximately 30% that of Bitcoin [78].

Finality is vital for practical applications of blockchain to ensure that blocks cannot be revoked once they are deployed. For example, a vendor receiving payment via cryptocurrency for an item is confident enough to release it to the buyer if they are sure that the payment cannot be reversed or changed by the buyer. When a user transacts, they want to be double sure that once the transactions go through, they cannot be reversed or changed arbitrarily. There are two basic types of finality: absolute/immediate finality and probabilistic finality. For absolute/immediate finality, a transaction is finalized immediately after it is included in a blockchain, whereas probabilistic finality implies that a transaction is not finalized immediately, but rather the probability that the transaction could be reversed decreases as more blocks are added to the chain (buried deeper in the blockchain).

5.2. Protocol Comparison

Throughput: We consider a consensus protocol to have high throughput if it greatly outperforms PoW, whereas a consensus protocol has low throughput if it performs similarly to PoW. PoF is considered to have high throughput, as it can process approximately 2 million transactions per second (2 million TPs) which is considered high, compared to the conventional protocols, such as PoW and PoS. Its high throughput can be attributed to its partial chain storage, which allows it to operate on small block sizes with less data and off-chain storage. The throughput of PoR varies based on its block size, which ranges between 200 and 1000 transactions per block for a network size of 100 to 500 nodes. The throughput is proportionate to the number of network participants (20, 30, 50, 70, 85 Tps for 100, 200, 300, 400, 500 nodes, respectively) as the size of a block increases. This is because larger-sized blocks can support higher transaction volume but instead, take more time to produce. The designers show experimentally that the throughput of PoR is dependent on the number of participating nodes active in the network. For instance, in a network with a few thousand blocks and participating nodes, hundreds of transactions can be processed per second (> 100 Tps). This is much higher compared to the throughput of PoW-powered solutions, such as Bitcoin. Another pair of consensus protocols relying on reputation, PoRX and PoX-R, includes a reputation module into existing PoX protocols. Mining difficulty is then scaled based on reputation. Its throughput is expected to be similar to its underlying PoX protocol.

On the other hand, similar to PoW, alternative protocols, such as PoB, PoP, PoL, PoSe and PoE, require nodes to compute puzzles (benefit number calculation from charging/discharging demands, machine learning or GA), which leads to lower throughput. However, rather than being wasted, computing power is used to come up with solutions for potential real-world problems. PoSn replaces difficulty in PoW with sincerity. The sincerity mechanism allows nodes with varying computing capabilities to essentially determine their difficulty level for solving the hashing puzzle. This will, theoretically, lead to an improvement in terms of throughput as compared to PoW.

PoA is a combination of PoW and PoC, thus it does not require a large number of computing resources. As such, its throughput is higher than PoW. PoPF involves ranking all participating nodes and selecting n nodes to be accountants who compete for the right to produce blocks. Accountants of higher rank have lower mining difficulty, compared to those of lower ranks. The ranking process ensures decentralization without the need to

constantly ramp up the difficulty. Thus, PoPF has a throughput that is theoretically larger than PoW, limited only by the processing speed of the individual participating. PoEx uses accumulated PoW to reduce the mining difficulty of nodes which then leads to higher throughput because nodes will lower miner difficulties who can, in theory, discover the target hash values in a shorter amount of time. The PoV consensus protocol has lower latency transaction validation, which leads to higher throughput as compared to PoW. When the full set of validations is used for the CHB and CHBD protocols, there is a chance that the stringent validation process may prevent a new block from being successfully created. In that case, they have a lower throughput as compared to PoW. However, the designers recommend omitting certain validation steps when the network is large enough and the number of honest nodes exceeds 50%. The throughput is then expected to be faster than and similar to PoW for CHB and CHBD, respectively. On the other hand, throughput in PoAj is higher compared to PoW because transactions are confirmed in PoAj after a single confirmation compared to the six confirmation waiting time in PoW.

Scalability: We consider a protocol to be highly scalable if it outperforms PoW, whereas low scalability implies that the protocol underperforms, compared to PoW. PoF is highly scalable as compared to PoS and PoW due to its off-the-chain data storage and smaller block size, hence more users can participate without the overall efficiency of the network being deteriorated. PoR was shown to be more scalable than PoW because the increase in participants leads to a shorter time in reaching a consensus. Its block production time decreases as the number of participants or users increase.

On the other hand, PoB, PoL, PoSe, PoEx, PoE, PoAj, and PoSn have similar architectures to PoW, whereby nodes compete to be block leaders by computing functions. Thus, they are expected to have similar scalability as PoW. Similarly, PoRX and PoX-R are expected to have similar scalability as their underlying PoX protocol. However, PoB and PoSn are expected to be more scalable than PoL because PoL must ensure that there are enough machine learning problems to solve in order to verify more blocks. PoA also involves nodes competing to solve a computational problem. In addition, PoA requires gathering input data that are stored in a stochastic manner, which prevents centralization by amassing computing power. As such, its scalability should be similar to PoW but lower than PoS.

PoPF is another consensus protocol that relies on solving the hash puzzle, but its block size and interval can be dynamically adjusted to scale accordingly. PoV also scales well, as the number of users does not affect the consensus process, which involves a specified number of commissioners and butlers. The CHB and CHBD protocols perform poorly, as the number of nodes increases due to their stringent validation process, which may prevent new blocks from being created. By omitting some of the overly strict validation steps, they scale similar to PoW.

Security: PoF is secure against the 51% attack because it tolerates a 0.7% diversity rate, which means that 75% of the participating nodes must agree before a chain can be taken over by an adversary. On the other hand, PoR operates in a permissioned blockchain network where all the participating nodes have their participation documented based on their public keys. Thus, any attempt by any malicious user to either fabricate identities or manipulate the network is detected instantly because all other network participants have a copy of all the public keys.

PoRX claims resistance to the 51% attack as malicious nodes will experience a reduction in reputation the more consecutive blocks that they publish. CHB and CHBD protocols also have resistance against the 51% attack due to a validation step that prevents nodes from being selected consecutively as block leaders. However, it is important to note that although CHB is resistant to a 51% attack, the relative computing power required to circumvent CHB is lower than in PoW. As the rest of the protocols are modifications or variants of PoW, they can only tolerate adversarial nodes of 50% or less.

Energy Consumption: Consensus protocols, such as PoW, have high energy consumption, as they require computational power for block leader selection. Thus, consensus

protocols, such as PoP, PoSe, PoEx, PoE, PoAj, that are based directly on PoW are considered to have high energy consumption. There are some protocols (PoE, PoSe and PoL) that channel this energy to solve useful, real-world problems. In addition, some of these protocols mitigate this problem by restricting the number of miners (PoA, PoPF) or by assigning varying levels of mining difficulty (PoSn, PoPF, PoRX, PoX-R). Some consensus protocols, such as PoB and PoL, do not perform hashing but rely on miners competing to solve different types of problems, such as maximizing a particular value (benefit) or creating machine learning models. Consensus protocols that do not require massive computational power to decide on block leaders include PoF, PoR, PoV and CHB. Although CHBD requires the calculation of PoW, it is not used in a competitive manner. The block leader is already selected at random, and PoW is computed for additional security.

Finality: In PoF, transactions are finalized immediately after being included in a block, which is then added into the blockchain. The rest of the consensus protocols are more conventional, whereby the transaction finality increases with the number of blocks appended to the blockchain.

6. Discussion

6.1. Critical Analysis

Participating nodes in blockchain networks that adopt CPE must provide proof of computational effort to be elected as a block publisher. Usually, nodes that come up with the fastest correct solution are elected to publish a block. Unfortunately, one of the main disadvantages of CPE is that mining consumes a large amount of electricity, which has an adverse effect on the environment. As an example, based on the 2017/2018 University of Cambridge's Bitcoin Electricity Consumption Index, Bitcoin consumes about 119.87 terawatt-hours per year, which is far more than what countries such as the United Arab Emirates and the Netherlands consume annually. In addition, this enormous electricity consumption has an adverse effect on the environment, and it is a reverse of the global desire and drive for clean energy. Instead, some CPEs, such as PoL, PoSe, PoE, and PoEx, channel the computational effort exerted toward solving useful problems. Although energy is still consumed for these protocols, it does not entirely go to waste. The reason why CPEs are popular in many blockchain networks is that attackers would require a large (often unattainable) computational effort to perform an attack. However, this also means that mining in CPE benefits those who can afford to amass a large amount of computing power. Protocols such as PoA attempt to overcome this problem by involving some stochastic data access to make it fairer for nodes with different computing capabilities, while PoSn allows miners with different computing capabilities to set their own mining difficulty levels in exchange for lower (but more consistent) rewards. The PoP protocol opts for a different direction to address the mining imbalance problem by adopting the high entry and low operation (HELO) cost approach but requires specialized hardware (AMU) to be installed on phones, which reduces its practicality. Instead, researchers can look into software-based alternatives to the AMU, which can drastically improve the feasibility of PoP or similar protocols.

Nodes in blockchain networks that adopt CPW must show evidence of staked wealth/resources or perform some form of payment before they can be selected as a block publisher. Evidence of wealth staked or burned within the network is what determines who publishes a block. The advantage of CPWs is that an attacker requires an enormous amount of wealth to launch an attack. An attacker will have to own 51% of the entire network's value, which is difficult to achieve in practice. Although protocols in the CPW category do not require large computational power like the CPE, the disadvantage of protocols in this category is that they benefit wealthier participants, as they are able to stake more cryptocurrency. Thus, these participants receive the most incentives and are able to grow their wealth even further.

In blockchain networks based on CPPB, participating nodes need to show evidence of good conduct and behavior earned over a certain period. Participating nodes with a

good record of behavior are incentivized with a non-transferable trust/reputation score. For protocols such as PoR, nodes with a high reputation/trust score have a higher chance of being selected as a block publisher than a node with a lower behavior score or reputation [69,70]. Identities of nodes are maintained using public keys, which can be used to easily detect falsification. Nodes (inclusive of attackers) have to contribute positively toward the blockchain for a long period of time to grow its reputation/trust score since the incentive in this category of protocols is non-transferable. Another behavior-based protocol, PoF, rewards nodes that make more accurate decisions by increasing their likelihood of being selected as a block proposer. However, the application of both PoR and PoF is limited to permissioned blockchains. For general-purpose or public blockchains, protocols such as PoRx and PoX-R combine the reputation module of the protocol with an underlying PoX protocol, such as PoW or PoS. Participating nodes with good conduct and behavior history are selected based on their accumulated reputation stored in the reputation module. PoS combined with the reputation module of PoR allow nodes with a higher reputation rank and moderate coin stake to compete to be selected as block publishers. Additionally, PoR combined with PoW helps the system to reduce the puzzle-solving difficulties of the underlying PoW protocol and also eliminates the risk of centralization from the use of ASIC and mining pools. In PoRX, nodes with higher reputations have lower mining difficulty as compared to nodes with lower reputations. Both of these protocols inherit both the advantages and disadvantages of underlying protocols. Both PoX-R and PoRx were developed by the same team of designers and as such, have similar properties. PoX-R improves the overall efficiency of the underlying PoX consensus protocols, resists stronger attacks and provides users with lower computing power and stacked coin, a greater opportunity to participate in the mining process. Apart from paying some fees, PoPF also rewards nodes that contribute actively by increasing their likelihood of being selected as block proposers.

For blockchain networks that rely on CPR, nodes must be elected or selected to validate and represent the interest of other participating nodes. These nodes are expected to act in the overall interest of the other participating nodes whose interest they are representing. Thus, nodes can participate in a more fair manner because amassing wealth or computational power provides no advantage. Protocols such as PoV still have miners, but these miners are elected by administrative nodes and are appointed to mine in a random manner to ensure fairness. For an intruder to attack protocols from this category, they must first be elected as a representative and gather a reasonable number of other elected representatives. Block validation in protocols such as CHB and CHBD are faster because they maintain a small number of validators. To ensure fairness, block proposers in these protocols are also selected in a random manner. However, protocols in this category tend more toward centralization because the decision-making mechanism in these protocols lies in the hands of a few representatives who act on behalf of the entire network. Another CPR, PoV, can only be implemented in a permissioned blockchain. All points in this subsection are summarized in Table 4.

6.2. Open Problems and Future Research Work

Based on our review, we found that a number of alternative consensus protocols were proposed for specific application areas, such as medicine or electric vehicles [56,67]. However, some of their design philosophies could be applied to develop more generalized consensus protocols or be used in other areas. For example, the concept of familiarity (PoF) could be applied in other decision-making applications that require transparency and collaboration from multiple parties. The task of optimizing a particular target value (PoB) when selecting block leaders could be applied in other areas, such as autonomous vehicle routing. Some consensus protocols, such as PoP, PoA and CHBD, propose interesting alternatives to existing protocols but are mainly theoretical in nature [57,60,72]. There is an opportunity for future researchers to study the actual implementation of these protocols to examine their feasibility and potential for practical use.

Table 4. Analysis of alternative consensus categories.

Protocol Category	Protocol	Description	Advantages	Disadvantages
Consensus Protocol based on Effort or Work (CPE)	PoB *, PoP, PoL *, PoSn, PoA, PoPF, PoAj, PoSe *, PoE *, PoEx *	Computational effort required to publish blocks	Large computational effort required to attack protocol, computational power spent for useful purposes *	High energy consumption, benefits nodes that amass computing power
Consensus Protocol based on Wealth or Resources (CPW)	PoPF	Staked wealth or payment required to publish blocks	Enormous wealth required to attack protocol, no energy wastage	Benefits wealthy participants
Consensus Protocol based on Past Behavior or Reputation (CPPB)	PoF **, PoR **, PoRx ***, PoX-R ***, PoPF	Non-transferable incentive based on node behaviour affects the selection of block publisher	Attacker must gather trust/reputation incentive which will be lost due to malicious actions, incentive is non-transferable, inherit advantages from underlying protocols ***	Only for permissioned blockchains **, inherits disadvantages from underlying protocols ***
Consensus Protocol based on Representation (CPR)	PoV **, CHB, CHBD	Block publishers are selected based on voting/election mechanism	An attacker must first be elected and have control of other elected representatives to be successful, faster validation due to small number of validators	Tends toward centralization, only for permissioned blockchains **

PoSn, PoPF and PoRX are modifications of PoW or PoX protocols that assign varying mining difficulty levels based on certain criteria. The dynamic nature of the mining difficulty alleviates the problem of centralization, as amassing more computational power does not lead to huge gains [59,66,69]. Researchers can leverage this feature to enhance existing protocols or to develop new ones. Finally, quite a number of consensus protocols with good trade-offs between the consensus protocol metrics (energy consumption, scalability, finality, tolerated adversary and throughput) have been designed specifically for permissioned blockchains [67,68,71]. As such, developing consensus protocols for public blockchains that can achieve a balanced trade-off of these requirements is still an open problem. In short, we note the following areas of research that future researchers can look into:

- Adopting design philosophies that channel puzzle-solving toward useful purposes.
- Incorporating non-transferable incentives (such as reputation or familiarity) that can dynamically control mining difficulty.
- Redesigning permissioned blockchain protocols with desirable properties to be applicable for public blockchains.
- Redesigning or improving alternative protocols to be applicable for real-world use.
- Conducting an experimental evaluation of alternative protocols on the same machine using the same simulation framework, such as BlockSim.

7. Conclusions

In this paper, we provided an overview of blockchain consensus protocols, emphasizing lesser-known *alternative* consensus protocols. Prior reviews in the blockchain space focused primarily on conventional blockchain consensus protocols, such as PoW and PoS, leaving out many of the alternative protocols that have been proposed in recent years. Although these alternative protocols have yet to be adopted for real-world use, they have unique design features that can contribute toward the design or improvement of existing consensus protocols. After a brief discussion of popular conventional protocols for the benefit of newer readers, we delved into 15 different alternative protocols proposed in the past 3 years that fall under four broad categories: consensus protocols based on effort/work, wealth/resources, past behavior/reputation and representation. We provided an in-depth evaluation of these alternative protocols with respect to throughput, scalability, security, energy consumption and finality. Apart from the individual protocols themselves,

we also analyzed the advantages and disadvantages of their broad categories. We found that CPE requires a large amount of computational power to ensure the security of the network, which not only has an adverse effect on the environment, but might also not be sustainable in the long run. However, the negatives can be offset by using the computational power spent on meaningful endeavors, such as finding solutions to computationally hard problems. Although CPW protocols do not require huge computational power, they tend toward centralization, as the network would be controlled by the wealthy to whom the incentives are also rewarded. This disadvantage can be overcome by ensuring that the mechanism for selecting block leaders also takes into account other non-wealth-related properties, such as node reputation in CPPB. CPPB operates on a non-transferable incentive where nodes are required to show evidence of good behavior or conduct over time. When implemented on top of other protocols, it can alleviate some of the disadvantages of these protocols, albeit not entirely. CPR, on the other hand, requires nodes to be selected or elected to represent the interest of other participating nodes. Although these protocols also tend toward centralization and are mainly for permissioned blockchains, they have faster validation procedures and require only a few nodes to maintain the network. In short, we have shown that alternative protocols have unique design features that can be used to develop or improve mainstream protocols in the future. Some of these protocols, such as those based on reputation, should be further investigated, as they may lead to better alternatives than the current state of the art.

Author Contributions: Conceptualization, D.P.O., J.S.T. and M.A.; methodology, D.P.O. and J.S.T.; investigation, D.P.O., J.S.T.; resources, J.S.T. and N.J.; data curation, D.P.O.; writing—original draft preparation, D.P.O., J.S.T., M.A.; writing—review and editing, D.P.O., J.S.T., M.A. and N.J.; visualization, D.P.O., J.S.T.; supervision, J.S.T.; project administration, J.S.T. and N.J.; funding acquisition, N.J. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Malaysian Fundamental Research Grant Scheme under Grant No. 203.PKOMP.6711801 and the Uniten BOLD Publication Fund 2021. The first author, D.P.O. is funded under the Tertiary Education Trust Fund (TETFUND) Nigeria.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

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

References

1. Yu, J.; Kozhaya, D.; Decouchant, J.; Esteves-Verissimo, P. RepuCoin: Your Reputation Is Your Power. *IEEE Trans. Comput.* **2019**, *68*, 1225–1237. [CrossRef]
2. Zou, J.; Ye, B.; Qu, L.; Wang, Y.; Orgun, M.A.; Li, L. A Proof-of-Trust Consensus Protocol for Enhancing Accountability in Crowdsourcing Services. *IEEE Trans. Serv. Comput.* **2019**, *12*, 429–445. [CrossRef]
3. Alzahrani, N.; Bulusu, N. Towards True Decentralization: A Blockchain Consensus Protocol Based on Game Theory and Randomness. In *Lecture Notes in Computer Science*; Springer International Publishing: Cham, Switzerland, 2018; pp. 465–485. [CrossRef]
4. Liu, B.; Liu, M.; Jiang, X.; Zhao, F.; Wang, R. A Blockchain-Based Scheme for Secure Sharing of X-Ray Medical Images. In *Security with Intelligent Computing and Big-Data Services*; Springer International Publishing: Cham, Switzerland, 2019; pp. 29–42. [CrossRef]
5. Domenico, M.D.; Baronchelli, A. The fragility of decentralised trustless socio-technical systems. *EPJ Data Sci.* **2019**, *8*. [CrossRef]
6. Yavuz, E.; Koc, A.K.; Cabuk, U.C.; Dalkilic, G. Towards secure e-voting using ethereum blockchain. In Proceedings of the 2018 6th International Symposium on Digital Forensic and Security (ISDFS), Antalya, Turkey, 22–25 March 2018. [CrossRef]
7. Giungato, P.; Rana, R.; Tarabella, A.; Tricase, C. Current Trends in Sustainability of Bitcoins and Related Blockchain Technology. *Sustainability* **2017**, *9*, 2214. [CrossRef]

8. Chen, Z.; Chen, S.; Xu, H.; Hu, B. A Security Authentication Scheme of 5G Ultra-Dense Network Based on Block Chain. *IEEE Access* **2018**, *6*, 55372–55379. [CrossRef]
9. Zhang, S.; Lee, J.H. Double-Spending With a Sybil Attack in the Bitcoin Decentralized Network. *IEEE Trans. Ind. Inform.* **2019**, *15*, 5715–5722. [CrossRef]
10. Shen, C.; Pena-Mora, F. Blockchain for Cities—A Systematic Literature Review. *IEEE Access* **2018**, *6*, 76787–76819. [CrossRef]
11. Nguyen, G.T.; Kim, K. A Survey about Consensus Algorithms Used in Blockchain. *J. Inf. Process. Syst.* **2018**, *14*, 101–128. [CrossRef]
12. Sharkey, S.; Tewari, H. Alt-PoW: An Alternative Proof-of-Work Mechanism. In Proceedings of the 2019 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON), Newark, CA, USA, 4–9 April 2019. [CrossRef]
13. Puthal, D.; Mohanty, S.P. Proof of Authentication: IoT-Friendly Blockchains. *IEEE Potentials* **2019**, *38*, 26–29. [CrossRef]
14. Lu, Y. Blockchain: A Survey on Functions, Applications and Open Issues. *J. Ind. Integr. Manag.* **2018**, *3*, 1850015. [CrossRef]
15. Vukolić, M. The Quest for Scalable Blockchain Fabric: Proof-of-Work vs. BFT Replication. In *Open Problems in Network Security*; Springer International Publishing: Cham, Switzerland, 2016; pp. 112–125. [CrossRef]
16. Cachin, C.; Vukolic, M. Blockchain Consensus Protocols in the Wild (Keynote Talk). In *Leibniz International Proceedings in Informatics (LIPIcs), Proceedings of the 31st International Symposium on Distributed Computing (DISC 2017), Vienna, Austria, 16–20 October 2017*; Richa, A.W., Ed.; Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik: Dagstuhl, Germany, 2017; Volume 91, pp. 1:1–1:16. [CrossRef]
17. Bano, S.; Sonnino, A.; Al-Bassam, M.; Azouvi, S.; McCorry, P.; Meiklejohn, S.; Danezis, G. SoK. In Proceedings of the 1st ACM Conference on Advances in Financial Technologies—AFT'19, Zurich, Switzerland, 21–23 October 2019 ; ACM Press: New York, USA, 2019. [CrossRef]
18. Wang, W.; Hoang, D.T.; Hu, P.; Xiong, Z.; Niyato, D.; Wang, P.; Wen, Y.; Kim, D.I. A Survey on Consensus Mechanisms and Mining Strategy Management in Blockchain Networks. *IEEE Access* **2019**, *7*, 22328–22370. [CrossRef]
19. Xiao, Y.; Zhang, N.; Lou, W.; Hou, Y.T. A Survey of Distributed Consensus Protocols for Blockchain Networks. Available online: http://arxiv.org/abs/1904.04098v3 (accessed on 30 June 2021).
20. Alsunaidi, S.J.; Alhaidari, F.A. A Survey of Consensus Algorithms for Blockchain Technology. In Proceedings of the 2019 International Conference on Computer and Information Sciences (ICCIS), Sakaka, Saudi Arabia, 3–4 April 2019. [CrossRef]
21. Ismail, L.; Materwala, H. A Review of Blockchain Architecture and Consensus Protocols: Use Cases, Challenges, and Solutions. *Symmetry* **2019**, *11*, 1198. [CrossRef]
22. Lepore, C.; Ceria, M.; Visconti, A.; Rao, U.P.; Shah, K.A.; Zanolini, L. A Survey on Blockchain Consensus with a Performance Comparison of PoW, PoS and Pure PoS. *Mathematics* **2020**, *8*, 1782. [CrossRef]
23. Berentsen, A. Aleksander Berentsen Recommends "Bitcoin: A Peer-to-Peer Electronic Cash System" by Satoshi Nakamoto. In *21st Century Economics*; Springer International Publishing: Cham, Switzerland, 2019; pp. 7–8. [CrossRef]
24. Ouattara, H.F.; Ahmat, D.; Ouédraogo, F.T.; Bissyandé, T.F.; Sié, O. Blockchain Consensus Protocols. In *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*; Springer International Publishing: Cham, Switzerland, 2018; pp. 304–314. [CrossRef]
25. Kingslin, S.; Zahra, R. An Effective Randomization Framework to POW Consensus Algorithm of Blockchain (RPoW). *Int. J. Eng. Adv. Technol.* **2019**, *8*, 1793–1797. [CrossRef]
26. Chaudhry, N.; Yousaf, M.M. Consensus Algorithms in Blockchain: Comparative Analysis, Challenges and Opportunities. In Proceedings of the 2018 12th International Conference on Open Source Systems and Technologies (ICOSST), Lahore, Pakistan, 19–21 December 2018. [CrossRef]
27. Sayeed, S.; Marco-Gisbert, H. Assessing Blockchain Consensus and Security Mechanisms against the 51% Attack. *Appl. Sci.* **2019**, *9*, 1788. [CrossRef]
28. *Komodo White Paper*; Technical Report; Komodo Platform. 2018. Available online: https://cryptorating.eu/whitepapers/Komodo/2018-02-14-Komodo-White-Paper-Full.pdf (accessed on 9 February 2021).
29. Leonardos, S.; Reijsbergen, D.; Piliouras, G. Weighted Voting on the Blockchain: Improving Consensus in Proof of Stake Protocols. In Proceedings of the 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Seoul, Korea, 14–17 May 2019. [CrossRef]
30. Chalaemwongwan, N.; Kurutach, W. State of the art and challenges facing consensus protocols on blockchain. In Proceedings of the 2018 International Conference on Information Networking (ICOIN), Chiang Mai, Thailand, 10–12 January 2018. [CrossRef]
31. Ogawa, T.; Kima, H.; Miyaho, N. Proposal of Proof-of-Lucky-Id(PoL) to Solve the Problems of PoW and PoS. In Proceedings of the 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Halifax, NS, Canada, 30 July–3 August 2018. [CrossRef]
32. Kiayias, A.; Russell, A.; David, B.; Oliynykov, R. Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol. Cryptology ePrint Archive, Report 2016/889, 2016. Available online: https://eprint.iacr.org/2016/889 (accessed on 30 June 2021).
33. Luo, Y.; Chen, Y.; Chen, Q.; Liang, Q. A New Election Algorithm for DPos Consensus Mechanism in Blockchain. In Proceedings of the 2018 7th International Conference on Digital Home (ICDH), Guilin, China, 30 November–1 December 2018. [CrossRef]
34. Do, T.; Nguyen, T.; Pham, H. Delegated Proof of Reputation. In Proceedings of the 2019 International Electronics Communication Conference on (IECC), Okinawa, Japan, 7–9 July 2019; ACM Press: New York, NY, USA, 2019. [CrossRef]

35. Barinov, I.; Baranov, V.; Khahulin, P. *POA Network Whitepaper*; Technical Report; 2018. Available online: https://github.com/poanetwork/wiki/wiki/POA-Network-Whitepaper (accessed on 30 May 2021).
36. *NEM Technical Reference*; Technical Report; NEM Foundation, 2018. Available online: https://nemplatform.com/wp-content/uploads/2020/05/NEM_techRef.pdf (accessed on 14 June 2021).
37. Castro, M.; Liskov, B. Practical byzantine fault tolerance and proactive recovery. *ACM Trans. Comput. Syst.* **2002**, *20*, 398–461. [CrossRef]
38. Cho, H. ASIC-Resistance of Multi-Hash Proof-of-Work Mechanisms for Blockchain Consensus Protocols. *IEEE Access* **2018**, *6*, 66210–66222. [CrossRef]
39. Lamport, L.; Shostak, R.; Pease, M. The Byzantine Generals Problem. *ACM Trans. Program. Lang. Syst.* **1982**, *4*, 382–401. [CrossRef]
40. Kwon, J. *Tendermint: Consensus without Mining*; Technical Report; Cornell University: Ithaca, NY, USA, 2014.
41. *State Machine Replication in the Libra Blockchain*; Technical Report; The LibraBFT Team, 2020. Available online: https://developers.diem.com/main/docs/state-machine-replication-paper (accessed on 19 April 2021).
42. Baird, L. *The Swirlds Hashgraph Consensus Algorithm: Fair, Fast, Byzantine Fault Tolerance*; Technical Report; Swirlds, 2016. Available online: https://www.swirlds.com/downloads/SWIRLDS-TR-2016-01.pdf (accessed on 5 March 2021).
43. Androulaki, E.; Manevich, Y.; Muralidharan, S.; Murthy, C.; Nguyen, B.; Sethi, M.; Singh, G.; Smith, K.; Sorniotti, A.; Stathakopoulou, C.; et al. Hyperledger fabric. In Proceedings of the Thirteenth EuroSys Conference on—EuroSys'18, Porto, Portugal, 23–26 April 2018; ACM Press, New York, NY, USA, 2018. [CrossRef]
44. Chase, B.; MacBrough, E. Analysis of the XRP Ledger Consensus Protocol. Available online: http://arxiv.org/abs/1802.07242v1 (accessed on 13 March 2021)
45. *The Ripple Consensus Algorithm*; Technical Report; Ripple. Available online: https://ripple.com/files/ripple_consensus_whitepaper.pdf (accessed on 30 January 2021).
46. *NEO Whitepaper*; Technical Report; NEO Foundation Available online: https://docs.neo.org/docs/en-us/basic/whitepaper.html (accessed on 1 April 2021).
47. Lokhava, M.; Losa, G.; Mazières, D.; Hoare, G.; Barry, N.; Gafni, E.; Jove, J.; Malinowsky, R.; McCaleb, J. Fast and secure global payments with Stellar. In Proceedings of the 27th ACM Symposium on Operating Systems Principles—SOSP'19, Ontario, Canada, 27–30 October 2019; ACM Press: New York, NY, USA, 2019. [CrossRef]
48. Bistarelli, S.; Pannacci, C.; Santini, F. CapBAC in Hyperledger Sawtooth. In *Distributed Applications and Interoperable Systems*; Springer International Publishing: Cham, Switzerland, 2019; pp. 152–169. [CrossRef]
49. Xiao, Y.; Zhang, N.; Li, J.; Lou, W.; Hou, Y.T. *Distributed Consensus Protocols and Algorithms*; John Wiley & Sons, Trenton, NJ, USA, 2019. [CrossRef]
50. Karantias, K.; Kiayias, A.; Zindros, D. Proof-of-Burn. Cryptology ePrint Archive, Report 2019/1096 (to be Presented at Financial Cryptography and Data Security 2020). 2019. Available online: https://eprint.iacr.org/2019/1096 (accessed on 30 June 2021).
51. P4Titan. *Slimcoin: A Peer-to-Peer Crypto-Currency with Proof-of-Burn*; Technical Report; Slimcoin.org: 2014 Available online: https://slimcoin.info/whitepaperSLM.pdf (accessed on 6 May 2021).
52. Gennaro, R.; Robshaw, M. (Eds.) *Advances in Cryptology—CRYPTO 2015*; Springer: Berlin/Heidelberg, Germany, 2015. [CrossRef]
53. Ren, L.; Devadas, S. Proof of Space from Stacked Expanders. In *Theory of Cryptography*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 262–285. [CrossRef]
54. Park, S.; Kwon, A.; Fuchsbauer, G.; Gaži, P.; Alwen, J.; Pietrzak, K. SpaceMint: A Cryptocurrency Based on Proofs of Space. In *Financial Cryptography and Data Security*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 480–499. [CrossRef]
55. Abusalah, H.; Alwen, J.; Cohen, B.; Khilko, D.; Pietrzak, K.; Reyzin, L. Beyond Hellman's Time-Memory Trade-Offs with Applications to Proofs of Space. In *Advances in Cryptology—ASIACRYPT 2017*; Springer International Publishing: Cham, Switzerland, 2017; pp. 357–379. [CrossRef]
56. Liu, C.; Chai, K.K.; Zhang, X.; Chen, Y. Proof-of-Benefit: A Blockchain-Enabled EV Charging Scheme. In Proceedings of the 2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring), Kuala Lumpur, Malaysia, 28 April–1 May 2019. [CrossRef]
57. Kim, J.M.; Lee, J.W.; Lee, K.; Huh, J. Proof of Phone: A Low-cost Blockchain Platform. In Proceedings of the 2019 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 11–13 January 2019. [CrossRef]
58. Bravo-Marquez, F.; Reeves, S.; Ugarte, M. Proof-of-Learning: A Blockchain Consensus Mechanism Based on Machine Learning Competitions. In Proceedings of the 2019 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON), Newark, CA, USA, 4–9 April 2019. [CrossRef]
59. Zaman, M.U.; Shen, T.; Min, M. Proof of Sincerity: A New Lightweight Consensus Approach for Mobile Blockchains. In Proceedings of the 2019 16th IEEE Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 11–14 January 2019. [CrossRef]
60. Kudin, A.M.; Kovalenko, B.A.; Shvidchenko, I.V. Blockchain Technology: Issues of Analysis and Synthesis. *Cybern. Syst. Anal.* **2019**, *55*, 488–495. [CrossRef]
61. Shamir, A. How to share a secret. *Commun. ACM* **1979**, *22*, 612–613. [CrossRef]
62. Sayeed, S.; Marco-Gisbert, H. Proof of Adjourn (PoAj): A Novel Approach to Mitigate Blockchain Attacks. *Appl. Sci.* **2020**, *10*, 6607. [CrossRef]

63. Shibata, N. Proof-of-Search: Combining Blockchain Consensus Formation With Solving Optimization Problems. *IEEE Access* **2019**, *7*, 172994–173006. [CrossRef]
64. Bizzaro, F.; Conti, M.; Pini, M.S. Proof of Evolution: Leveraging blockchain mining for a cooperative execution of Genetic Algorithms. In Proceedings of the 2020 IEEE International Conference on Blockchain (Blockchain), Rhodes, Greece, 2–6 November 2020. [CrossRef]
65. Masseport, S.; Darties, B.; Giroudeau, R.; Lartigau, J. Proof of Experience: Empowering Proof of Work protocol with miner previous work. In Proceedings of the 2020 2nd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS), Paris, France, 28–30 September 2020. [CrossRef]
66. Fu, X.; Wang, H.; Shi, P.; Mi, H. PoPF: A Consensus Algorithm for JCLedger. In Proceedings of the 2018 IEEE Symposium on Service-Oriented System Engineering (SOSE), Bamberg, Germany, 26–29 March 2018. [CrossRef]
67. Yang, J.; Onik, M.; Lee, N.Y.; Ahmed, M.; Kim, C.S. Proof-of-Familiarity: A Privacy-Preserved Blockchain Scheme for Collaborative Medical Decision-Making. *Appl. Sci.* **2019**, *9*, 1370. [CrossRef]
68. Gai, F.; Wang, B.; Deng, W.; Peng, W. Proof of Reputation: A Reputation-Based Consensus Protocol for Peer-to-Peer Network. In *Database Systems for Advanced Applications*; Springer International Publishing: Cham, Switzerland, 2018; pp. 666–681. [CrossRef]
69. Wang, E.K.; Liang, Z.; Chen, C.M.; Kumari, S.; Khan, M.K. PoRX: A reputation incentive scheme for blockchain consensus of IIoT. *Future Gener. Comput. Syst.* **2020**, *102*, 140–151. [CrossRef]
70. Wang, E.K.; Sun, R.; Chen, C.M.; Liang, Z.; Kumari, S.; Khan, M.K. Proof of X-repute blockchain consensus protocol for IoT systems. *Comput. Secur.* **2020**, *95*, 101871. [CrossRef]
71. Li, K.; Li, H.; Hou, H.; Li, K.; Chen, Y. Proof of Vote: A High-Performance Consensus Protocol Based on Vote Mechanism & Consortium Blockchain. In Proceedings of the 2017 IEEE 19th International Conference on High Performance Computing and Communications: IEEE 15th International Conference on Smart City: IEEE 3rd International Conference on Data Science and Systems, Bangkok, Thailand, 18–20 December 2017. [CrossRef]
72. Yu, L.; Zhao, X.-F.; Jin, Y.; Cai, H.-Y.; Wei, B.; Hu, B. Low powered blockchain consensus protocols based on consistent hash. *Front. Inf. Technol. Electron. Eng.* **2019**, *20*, 1361–1377. [CrossRef]
73. Feng, L.; Zhang, H.; Tsai, W.T.; Sun, S. System architecture for high-performance permissioned blockchains. *Front. Comput. Sci.* **2019**, *13*, 1151–1165. [CrossRef]
74. Stephen, R.; Alex, A. A Review on BlockChain Security. *IOP Conf. Ser. Mater. Sci. Eng.* **2018**, *396*, 012030. [CrossRef]
75. The P + epsilon Attack. Available online: https://blog.ethereum.org/2015/01/28/p-epsilon-attack/ (accessed on 1 January 2020).
76. Deirmentzoglou, E.; Papakyriakopoulos, G.; Patsakis, C. A Survey on Long-Range Attacks for Proof of Stake Protocols. *IEEE Access* **2019**, *7*, 28712–28725. [CrossRef]
77. Bitcoin Energy Consumption Index. Available online: https://digiconomist.net/bitcoin-energy-consumption (accessed on 16 January 2020).
78. Xue, T.; Yuan, Y.; Ahmed, Z.; Moniz, K.; Cao, G.; Wang, C. Proof of Contribution: A Modification of Proof of Work to Increase Mining Efficiency. In Proceedings of the 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), Tokyo, Japan, 23–27 July 2018. [CrossRef]