electronics

Article

A Comparative Study of Consensus Mechanisms in Blockchain for IoT Networks

Zachary Auhl [1,*], Naveen Chilamkurti [1], Rabei Alhadad [1] and Will Heyne [2]

[1] Cybersecurity Innovation Node, La Trobe University, Melbourne, VIC 3086, Australia
[2] BAE Systems, Adelaide, SA 5000, Australia
* Correspondence: z.auhl@latrobe.edu.au

Abstract: The consensus mechanism is a core component of Blockchain technology, allowing thousands of nodes to agree on a single and consistent view of the Blockchain. A carefully selected consensus mechanism can provide attributes such as fault tolerance and immutability to an application. The Internet of Things (IoT) is a use case that can take advantage of these unique Blockchain properties. IoT devices are commonly implemented in sensitive domains such as health, smart cities, and supply chains. Resilience and data integrity are important for these domains, as failures and malicious data tampering could be detrimental to the systems that rely on these IoT devices. Additionally, Blockchains are well suited for decentralised networks and networks with high churn rates. A difficulty involved with applying Blockchain technology to the IoT is the lack of computational resources. This means that traditional consensus mechanisms like Proof of Work (PoW) are unsuitable. In this paper, we will compare several popular consensus mechanisms using a set of criteria, with the aim of understanding which consensus mechanisms are suitable for deployment in the IoT, and what trade-offs are required. We show that there are opportunities for both PoW and PoS to be implemented in the IoT, with purpose-made IoT consensus mechanisms like PoSCS and Microchain. Our analysis shows that Microchain and PoSCS have characteristics that are well suited for IoT consensus.

Keywords: consensus; IoT; Blockchain

check for updates

Citation: Auhl, Z.; Chilamkurti, N.; Alhadad, R.; Heyne, W. A Comparative Study of Consensus Mechanisms in Blockchain for IoT Networks. *Electronics* **2022**, *11*, 2694. https://doi.org/10.3390/electronics11172694

Academic Editor: Asma Khatoon

Received: 18 July 2022
Accepted: 22 August 2022
Published: 27 August 2022

1. Introduction

Blockchains are cryptographically linked distributed ledgers that are known for storing the transaction history of the Bitcoin network. Bitcoin adopted the Blockchain as it has two important properties: tamper-evidence and the triple-entry ledger. As each block in the Blockchain is cryptographically linked to the previous block, attempts to tamper with the Blockchain will invalidate the blocks cryptographic link. This means malicious parties cannot arbitrarily alter the history of the Blockchain. Triple-entry accounting refers to the distributed characteristics of a Blockchain. Instead of relying on two parties to provide evidence of their activities, transactions on the Blockchain are transmitted to the whole network, allowing anyone to validate all the transactions on the Blockchain. Often paired with a Blockchain, is a consensus mechanism. Consensus mechanisms allow Blockchains to converge on network-wide agreement on the state of the Blockchain, meaning that nodes all agree on the same history of the ledger. Consensus on the Bitcoin's Blockchain relies on two mechanisms, Proof of Work (PoW), and the Longest Chain Rule (LcR). PoW is a cryptographic puzzle that miners on the Bitcoin network attempt to solve. This provides miners with a financial incentive to support the network, and prevents Sybil attacks. The LcR is a fork resolution tool, that manages competing histories of the Blockchain, and converges the network back onto a single state in cases where the Blockchain forks. Recently, there has been interest in applying Blockchain technology to the Internet of Things (IoT). Specifically, consensus mechanisms have been modified to be less resource intensive, and more suitable for deployment in the IoT, with consensus mechanisms such as the Credit-Based PoW (CBPoW) and Proof of Supply Chain Share (PoSCS). This paper will examine the suitability of permissionless Blockchains for the IoT and the trade-offs required, especially for resource limited IoT devices.

1.1. Outline

We begin the paper by discussing the criteria. We use these criteria to compare the consensus mechanisms discussed in the paper. Next, we briefly discuss the fundamental properties of Blockchains in the background section. The second half of the paper is focused on analysing consensus mechanisms. We start by covering consensus mechanisms commonly found in blockchains, such as PoW and PoS; then, we discuss four newer consensus mechanisms from the literature, specifically designed for the IoT. Finally, we compare the discussed consensus mechanisms using our proposed criteria. We pay close attention to properties that positively and negatively impact the critical characteristics of IoT devices. Finally, we conclude with consensus recommendations for the IoT and outline future work.

1.2. Contributions

In this paper, we provide the following contributions:

- Analysis of PoW and PoS consensus mechanisms and their usability in the IoT.
- Analysis of four novel consensus mechanisms from the literature, specifically designed for the IoT.
- A comparison between the mentioned consensus mechanisms, with clear criteria to show their suitability for the IoT.

2. Criteria for Blockchain Consensus

The IoT environment is incredibly diverse, involving a wide range of hardware solutions and software solutions paired with a stringent set of requirements, involving power consumption, storage, and computational capabilities. IoT devices work in dynamic environments, where sensors generate data constantly, coming online and offline depending on their power requirements, and expected to work in ad hoc networks.

Due to this flexibility, IoT devices have seen widespread usage in applications such as smart cities [1] supply chains [2–5] and healthcare [6]. The consensus mechanism is a critical part of most Blockchain deployments, but the choice becomes even more important when working with Blockchain deployments targeted towards the IoT. All Blockchains come with trade-offs; there is no such thing as a ’perfect’ Blockchain. Some are more resource-intensive, some are faster, and others are more centralised. To compare the Blockchains discussed throughout this paper, we will define a set of requirements, to better understand their usability and impact in IoT environments.

1. Processor Usage: How are IoT devices going to agree on the content and order of the Blockchain? Historically, Blockchains made use of PoW to decide this. However, PoW is well-known for being computationally expensive, and environmentally destructive, and has seen waning interest in newer consensus mechanisms. Extending the battery life of IoT devices, and maintaining an acceptable processor utilisation, will be an important factor when selecting a consensus mechanism.

2. Security: Blockchain implementations may provide stronger security guarantees, when compared to traditional IoT networks with central points of failure (coordinators, controllers, cloud networks etc.). Many Blockchains suffer from potential attacks when the number of malicious nodes reaches 51% and 33% depending on the consensus mechanism. While there is no longer a single point of failure for a malicious actor to target, Blockchain specific attacks can still compromise the IoT network.

3. Decentralisation: A choice for most Blockchains which is not binary and operates on a sliding scale. Increasing the network’s decentralisation further diversifies Blockchain storage and decision-making, but usually impacts the speed and scalability of the net-

work. Decreasing decentralisation has an inverse effect, which reduces the diversity in the consensus process, and prioritises scalability and speed.

4. Storage: A factor that needs to be considered for security and the decentralisation aspects of a Blockchain. If all nodes on the network store a full copy of the Blockchain, they can independently verify transactions, and help new nodes bootstrap their Blockchains. IoT devices generally do not have the capacity to store hundreds of gigabytes worth of Blockchain data, so a compromise that still allows for security, and potentially decentralisation needs to be made.

5. Transactions Per Second (TPS): Another trade-off occurs between decentralisation and speed. The more nodes that participate in consensus, the higher the latency to make a decision, which results in generally lower speed, but higher decentralisation. A lower number of nodes participating in consensus could lead to increased transaction throughput and lower block times, which is generally desirable for IoT devices.

3. Background

Consensus mechanisms were traditionally deployed to maintain critical control systems, such as those aboard commercial airlines. Aboard an airline, the consensus mechanism is used to coordinate multiple control systems, keeps the system operational even in partial failure, and keeps hundreds of passengers on an aircraft safe [7].

Consensus mechanisms, such as Paxos, have also been widely adopted by Google and Amazon in their distributed systems. Google's projects such as Spanner [8], which provide a distributed, replicated database system, and Mesa [9], which provides distributed data warehousing.

A consensus mechanism defines a set of rules or protocols a group of systems needs to abide by, in order to make a decision. Let us use Bitcoin as an example. Part of Bitcoin's consensus mechanism lets users running full nodes agree on Bitcoin's Blockchain history. Each nonfaulty participant running a full node, and enforcing Bitcoin's consensus rules, will check transactions for issues like: a user spending Bitcoin they do not own, a user trying to print Bitcoin out of thin air, or a miner creating a block and rewarding themselves with thousands of Bitcoin [10].

Rather than a central entity enforcing these rules, every full node on the Bitcoin network is enforcing these rules. One of the most famous examples of distributed agreement was published by Lamport et al. titled the "Byzantine Generals Problem" [11]. This paper includes a thought experiment involving a set of generals planning an attack on an enemy city. If a treacherous commander or lieutenant attempts to deceive their peers by sending incorrect orders, this could lead to disaster for the army. If parts of the army attack, while other parts of the army retreat, the battle cannot be won. The paper proves that in order to deal with b Byzantine nodes, there must be at least 3b+1 nodes on the network [11]. Byzantine actors are capable of colluding to deceiving other users in our system, which breaks consensus by simple majority. To accommodate for Byzantine actors and to maintain safety and liveness guarantees, $b < 1/3$ of the total nodes on the network, so that consensus cannot be split by a 50/50 vote on our network by colluding nodes [12].

Types of Blockchains

Blockchains can be generalised into two categories: permissionless, and permissioned. Permissionless, like Bitcoin and Ethereum, are public Blockchains that anyone can participate in. Users are free to create transactions on these networks, interact with smart contracts, and are free to propose blocks if they participate in mining. Users on permissionless Blockchains are generally pseudonymous, which means users wallets are associated with certain public addresses, but not necessarily linked to a person's name.

Permissioned Blockchains are more restrictive on how users can interact with the Blockchain, and generally have different use cases. Examples of permissioned Blockchains include R3 [13] and J.P. Morgan's Quorum [14], which are both targeted at the financial industry. These Blockchains may allow the public to join, but are generally invite only, and have a stricter set of rules.

Private Blockchains can be thought of as an extension to permissioned Blockchains. The difference is permissioned Blockchains may be publicly accessible, as long as a user meets some sort of criteria. Private Blockchains are generally not accessible to the public, and are almost always invite only [15]. In Section 4, we will discuss PoW Blockchains, and their usability in the IoT.

4. Proof of Work

PoW is a consensus mechanism used by Bitcoin, and has been forked, modified and copied by many other cryptocurrencies [16]. At its core, most PoW implementations involve solving a cryptographic puzzle with certain parameters, and the first machine to solve this problem is rewarded. Figure 1 shows a diagram of this process. More specifically, miners are searching for a nonce (a random number), that can be hashed together with the block header, to produce a block hash, with a specific number of starting zeros [17]. The first miner on the network to produce a hash with these specific requirements, is given the block reward as payment for their service. Bitcoin's consensus mechanism involves the interaction between two important components, PoW, and the Longest Chain Rule (LcR), otherwise known as Nakamoto Consensus [10]. PoW provides two important features: a mechanism that provides a financial incentive for mining, and a way of preventing Sybil attacks. Bitcoin's second component, LcR, exists due to a trade-off Bitcoin made to maintain consensus guarantees. Bitcoin had to compromise either safety or liveness to guarantee consensus under Byzantine actors, and in an asynchronous network environment, with Bitcoin choosing safety [18]. Bitcoin is unable to provide strong safety, which means Bitcoin cannot guarantee that every node on the network will have identical copies of the Blockchain [19]. With the ability for forks to occur, Bitcoin requires the LcR as a fork resolution tool. In the case the network is split, the LcR states that the fork with the most aggregated computational work, is the correct Blockchain [20].

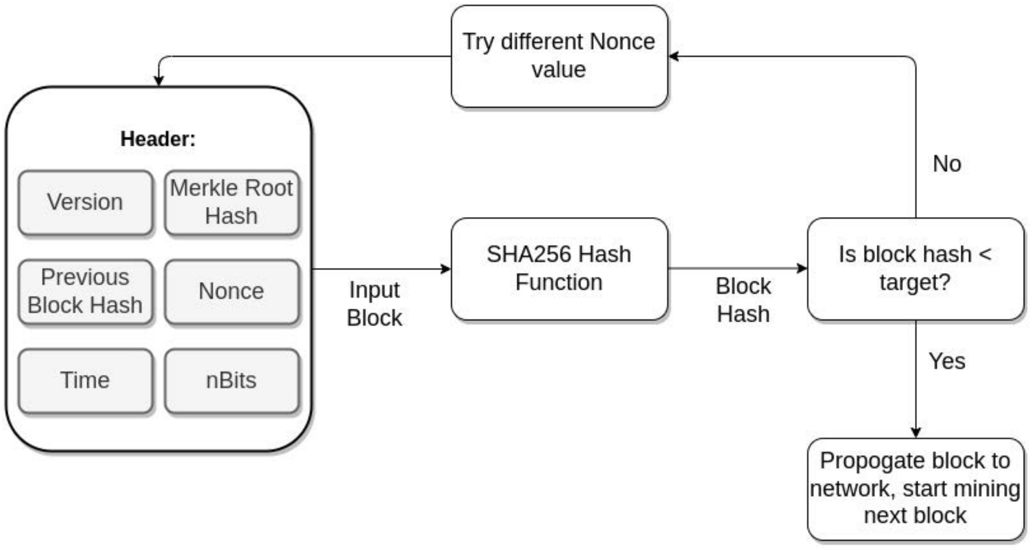


Figure 1. The process of mining on PoW Blockchains.

Proof of work is still widely used by several cryprocurrencies such as Ethereum, Litecoin, and Monero with slight differences. Ethereum uses a modified version of the SHA3 hashing algorithm called Keccak-256 [21], Litecoin uses the Scrypt hash function [22], and Monero uses an evolution of the CryptoNote hash function called CryptoNight [23].

With these two rules we have a system that rewards miners, stops Sybil attacks, and can reconcile forks in a trustless and decentralised manner. With hundreds of billions of dollars at stake, Bitcoin is yet to be hacked catastrophically, and, in the words of Andreas

Antonopoulos, Bitcoin has become the “sewer rat” of Blockchains (https://aantonop.com/bubble-boy-and-the-sewer-rat/ (accessed on 20 August 2022)).

4.1. Credit-Based PoW (CBPoW)

Huang et al. proposes a credit-based PoW system that’s suitable to run on IoT devices [24]. The authors created a consensus mechanism that dynamically adjusts a device’s PoW difficulty depending on their adherence to the consensus rules. A node’s total score can be calculated by taking the sum of their positive score along with their negative score as shown in Figure 2. A positive score can be increased by following the consensus mechanism, while their negative score grows by disobeying consensus. The paper focuses on two specific attacks that could lower a client’s score: lazy tips and double spending. Lazy tips is an issue that specifically effects Directed Acyclic Graphs (DAGs), where a malicious actor will avoid confirming recent transactions by building on top of old, preexisting transactions.

This can be detrimental to the network, as honest nodes may not have their new transactions approved. Huang et al. also penalises users for attempting to spend their tokens twice. If nodes are found to be acting maliciously, a penalty function that takes the sum of their malicious transactions, over a certain period of time, multiplied by a punishment coefficient, is used to penalise their credit amount. As nodes are required to confirm two previous transactions before submitting their own, a low PoW credit will make adding a transaction time intensive, and computationally expensive. CBPoW uses a tiered node network, where lite nodes are responsible for collecting data and broadcasting transactions, and full nodes are responsible for maintaining the tangle.

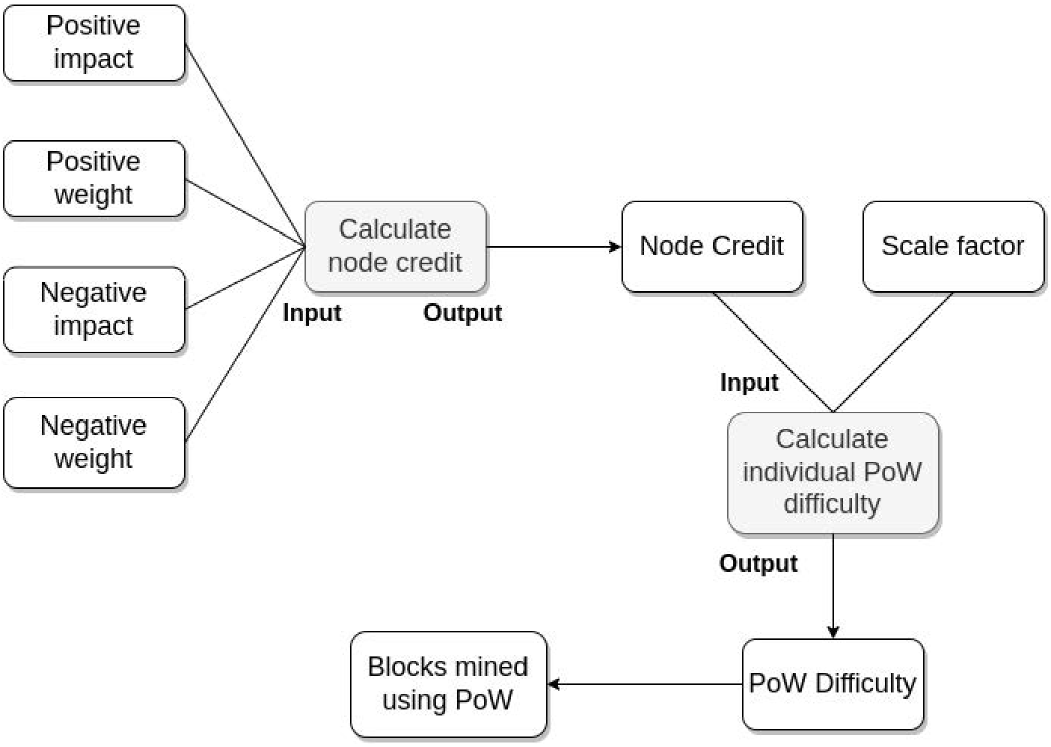


Figure 2. CBPoW Consensus.

4.2. Proof of Elapsed Work and Luck (PoEWAL)

PoEWAL is a consensus mechanism with similar traits to Bitcoin, but has been modified to be mineable on resource constrained devices [25]. PoEWAL still requires devices to solve a cryptographic puzzle, however, rather than devices searching for the matching nonce, miners just need to mine for a short period of time. This heavily reduces the power and computational load on IoT devices. Once the mining time in a round elapses, miners will compare their hash values used to solve the computational puzzle. The node, whose hash value has the highest number of consecutive zeros, has the right to produce a block for the round. In the case that two miners propose hashes with the same number of consecutive zeros, Huang et al. proposes a fork resolution tool called Proof of Luck. Proof of Luck compares the two hashes values with equal consecutive zeros, then selects the node whose hash value was the lowest to propose a block as presented in Figure 3. PoEWAL is able to enforce these highly synced time limits on their consensus mechanism as the authors assume that their IoT devices will have synced clocks. PoEWAL also implements a dynamic difficulty level depending on the number of collisions. If collisions happen at a regular frequency, the difficulty level will be raised in an attempt to lower the number of collisions.

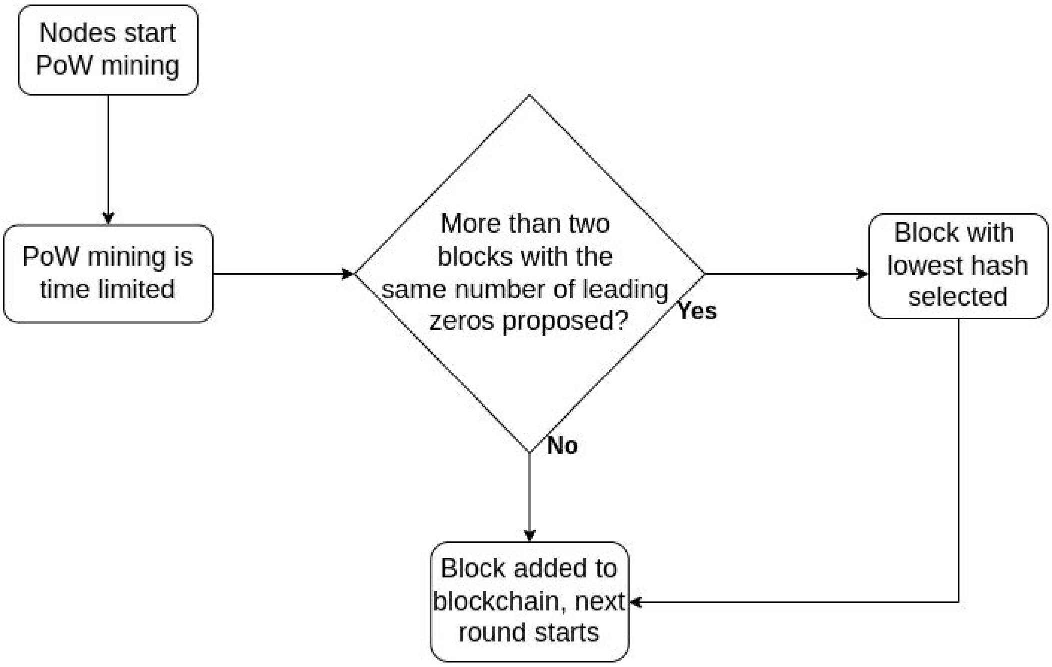

Nodes start PoW mining
PoW mining is time limited
More than two blocks with the same number of leading zeros proposed?
Yes
No
Block with lowest hash selected
Block added to blockchain, next round starts

Figure 3. PoEWAL Consensus.

5. Proof of Stake

Proof of Stake (PoS) was presented in a paper written by Sunny King and Scott Nadal in 2012 [26]. King et al. proposed that the age of a cryptocurrencies coin, known as the coin-age, could be used to develop an alternative consensus mechanism to PoW. The authors propose a system where PoW mints the initial supply of coins on the network, and then slowly diminishes the mining rewards to lower the reliance on PoW. Sunny King went on to create Peercoin (PPC) a fork of Bitcoin in 2013. Peercoin implemented an initial PoW coin distribution. The proposed consensus mechanism also used coin-age to stop wealthy users from hoarding staking rewards, and checkpoints to deny changes to the Blockchain after a certain point [24]. Rather than finding a nonce, a node is selected to mine the next block using a pseudorandom lottery. The larger the node's stake of coins in proportion to the rest of the network, the higher the chance of being selected to mine a block [16]. Similarly, to PoW, the header is hashed, but rather than spending large amounts of electricity, constantly hashing different nonces, PoS does one calculation. If the coin age > blockhash/target, the node can create a new valid block. Figure 4 shows a similar mechanism, but displays the more sophisticated PoS kernel as a replacement to the coin age. If not, the node waits til the next round to check if it meets the criteria to produce a block [27]. PoS proved popular due to its minimal hardware requirements and reduced energy usage compared to PoW. Cryptocurrencies such as Algorand, Cardano, and PIVX are examples of cryptocurrencies that have adopted PoS as their consensus mechanism. Algorand [28] co-found by cryptographer Silvio Micali, Cardano, one of the largest smart contract platforms by market cap [29], and PIVX, that allows users to run 'masternodes', nodes which provide extra security and functionality for the network [6].

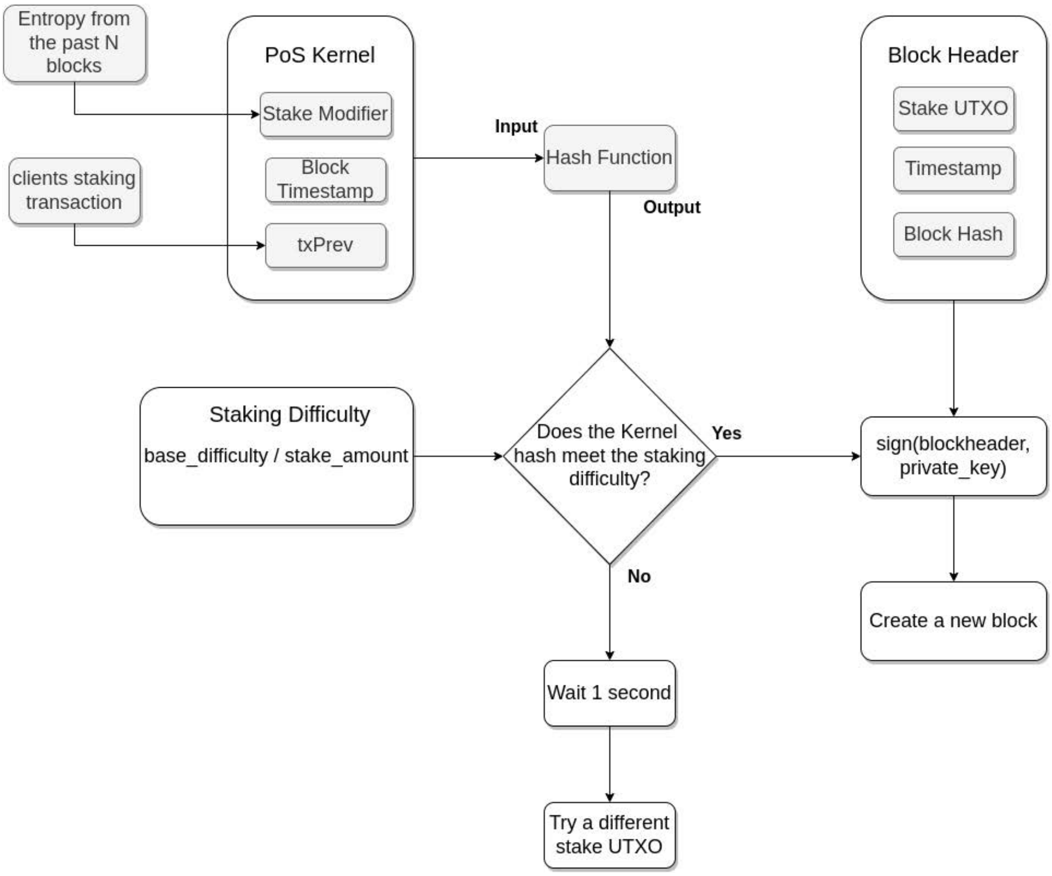

Entropy from the past N blocks
clients staking transaction
PoS Kernel
Stake Modifier
Block Timestamp
txPrev
Input
Hash Function
Output
Block Header
Stake UTXO
Timestamp
Block Hash
Staking Difficulty
base_difficulty / stake_amount
Does the Kernel hash meet the staking difficulty?
Yes
No
sign(blockheader, private_key)
Create a new block
Wait 1 second
Try a different stake UTXO

Figure 4. PoS Consensus.

5.1. Byzantine Agreement Protocol (BAP)

Algorand is a cryptocurrency co-found by Silvio Micali, and uses a Verifiable Random Function (VRF) to power its consensus mechanism, the Byzantine Agreement Protocol [28]. Nodes on the network can choose to participate in consensus by computing an evaluation function. The Decentralised Random Beacon (DRB) allows nodes to agree on a VRF and to collaboratively create one new output of the VRF every round. A VRF in this context means a commitment to a deterministic, pseudorandom value. In particular, the VRF outputs are unbiased due to their pseudorandom qualities [30].

On the Algorand network, a user has their own unique secret key, and a 'magic seed' known to nodes on the Algorand network. The evaluation function returns an output string (which is used to select committee members) and a proof to verify the output. Next the output string is checked to see if it falls between a range [0, user stake] where user stake is the proportion of the coins a user has staked compared to the total coins staked. If the output string falls in this range, the user is selected to join the committee for the current round [28]. The VRF also acts as a lottery to select leaders to propose blocks to the committee.

If most of the committee is honest, and a node proposes a valid block, a block can be certified and added to the Blockchain. This process is shown in Figure 5.

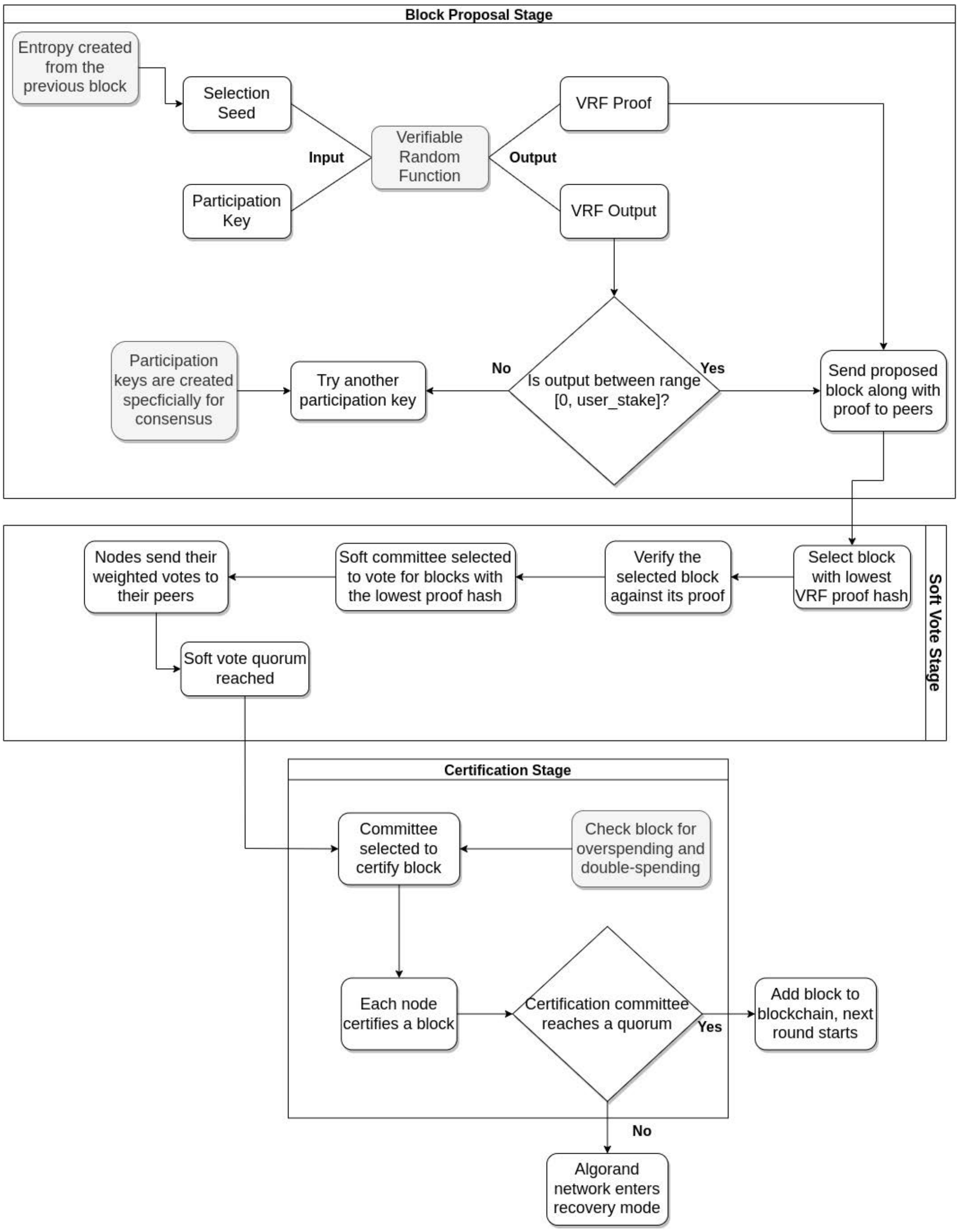

Block Proposal Stage
Entropy created from the previous block
Selection Seed
Participation Key
Input
Verifiable Random Function
Output
VRF Proof
VRF Output
Participation keys are created specficially for consensus
Try another participation key
No
Is output between range [0, user_stake]?
Yes
Send proposed block along with proof to peers
Soft Vote Stage
Select block with lowest VRF proof hash
Verify the selected block against its proof
Soft committee selected to vote for blocks with the lowest proof hash
Nodes send their weighted votes to their peers
Soft vote quorum reached
Certification Stage
Committee selected to certify block
Check block for overspending and double-spending
Each node certifies a block
Certification committee reaches a quorum
Yes
Add block to blockchain, next round starts
No
Algorand network enters recovery mode

Figure 5. BAP Consensus.

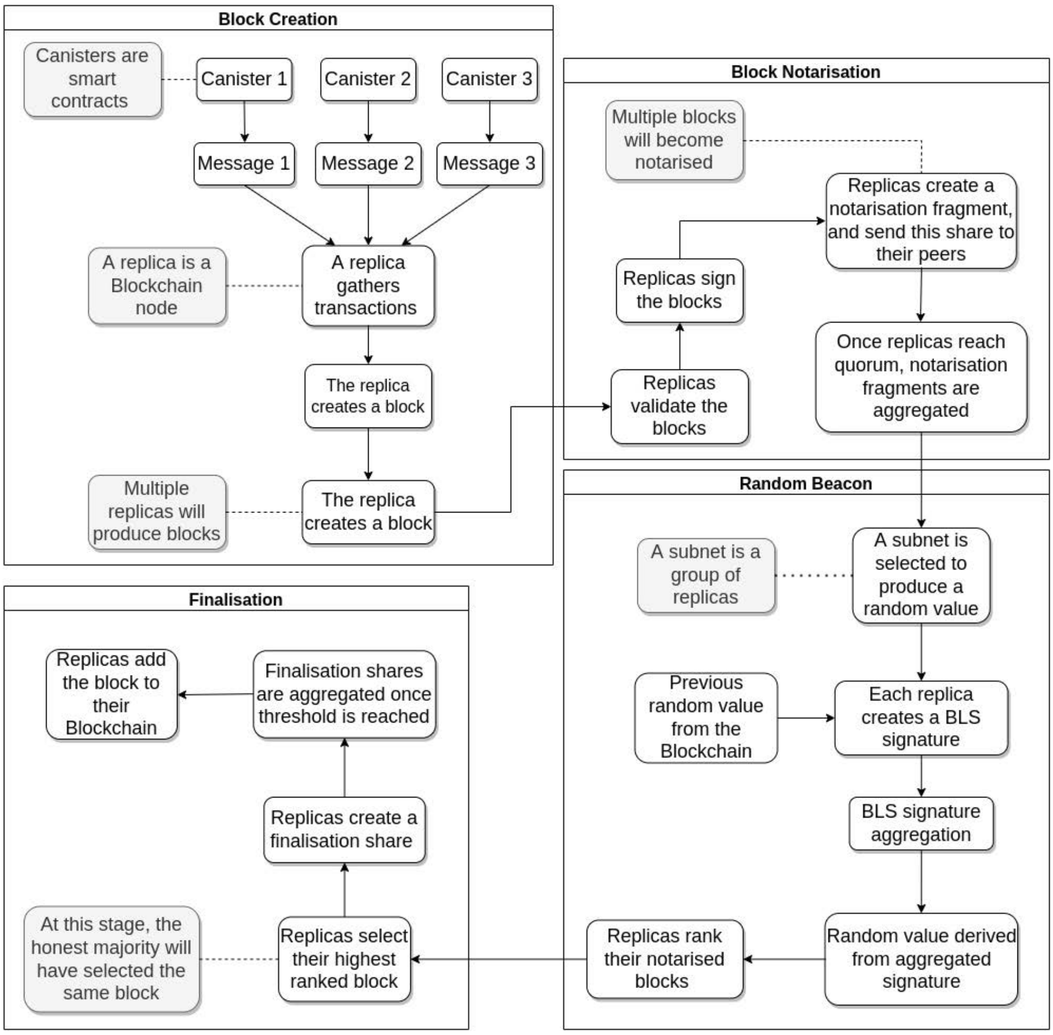
5.2. Dfinity

Dfinity Launched on the 18 December 2020 and is positioning itself as the "Internet Computer". Dfinity is creating a Blockchain that can host various online services, such as those provided by Amazon AWS and Google Cloud. The Dfinity consensus mechanism is split into 4 segments [31], summarised here, and depicted in detail in Figure 6.

1. Identities and Registry: Used to register clients to the network, each client has a permanent pseudonymous identity. This is a form of Sybil protection to defend against malicious users flooding the network with fake identities.
2. Random Beacon: built on top of a Verifiable Random Function (VRF) that allows registered clients on the network to generate and agree upon random numbers. Dfinity uses an optimised implementation of the Boneh–Lynn–Shacham (BLS) signature scheme, which they have used to solve the last actor problem. This problem involves

the last actor in the protocol knowing the random value for the next round, and having the ability to abort the protocol

3. Blockchain and Fork Resolution: This segment implements the Probabilistic Slot Protocol (PSP) that is used to rank the clients for a particular round according to the output from the Random Beacon. This rank is used to assign a weight to block proposers, with a higher rank resulting in a better chance of being selected to create a block. PSP offers instantaneous ranking and a deterministic block time.
4. Notarisation and near-instant finality: Block notarisation is Dfinity's technique to provide near-instant finality, that is, network-wide and irreversible agreement on a new block. Dfinity takes advantage of the BLS threshold signatures [32], Random Beacon, and the client ranking system to achieve this.



Block Creation
Canisters are smart contracts
Canister 1
Canister 2
Canister 3
Message 1
Message 2
Message 3
A replica is a Blockchain node
A replica gathers transactions
The replica creates a block
Multiple replicas will produce blocks
The replica creates a block
Block Notarisation
Multiple blocks will become notarised
Replicas create a notarisation fragment, and send this share to their peers
Replicas sign the blocks
Once replicas reach quorum, notarisation fragments are aggregated
Replicas validate the blocks
Random Beacon
A subnet is a group of replicas
A subnet is selected to produce a random value
Previous random value from the Blockchain
Each replica creates a BLS signature
BLS signature aggregation
Replicas rank their notarised blocks
Random value derived from aggregated signature
Finalisation
Replicas add the block to their Blockchain
Finalisation shares are aggregated once threshold is reached
Replicas create a finalisation share
At this stage, the honest majority will have selected the same block
Replicas select their highest ranked block

Figure 6. Dinifity Consensus.

5.3. Ethereum PoS Consensus

Ethereum is a Blockchain that was originally conceived by Vitalik Buterin in 2013 [33], and extended upon in 2014 by Gavin Wood to define Ethereum's smart contract functionality [34]. Ethereum launched in 2015, with the code name 'Frontier'. At the time of writing, Ethereum is the second largest cryptocurrency in terms of market cap. Ethereum is a Blockchain known for decentralised applications, commonly called 'DApps'. The Ethereum DApp ecosystem is diverse, with a range of widely used DApps in areas such as finance [35,36], gaming [37] and prediction markets [38]. Currently, Ethereum is using PoW as its consensus mechanism, but plans to move to PoS in an upgrade commonly known as 'Eth2'. The Eth2 upgrade is underway, and being rolled out progressively. The roll out is planned to occur in 3 stages, with each stage implementing several changes. Stage 1, launches the Beacon chain, which will introduce PoS, stage 2 will focus on merging the Ethereum PoW chain, and the Beacon chain, and step 3 will focus on scalability with the implementation of sharding [39]. Consensus on Eth2 is going to involve two components, the Greedy Heaviest Observed Subtree (GHOST), which will act as a fork resolution tool, and Casper the Friendly Finality Gadget (CFFG), which will finalise the decisions that GHOST makes [40]. The GHOST protocol compromises on its safety, which means that it is possible to switch between different forks, with different chain heights. However, as GHOST has liveness guarantees, blocks can continue to be added to the Ethereum Blockchain even when the chain is under attack.

The CFFG protocol finalises the blocks that are added to the chain, and as CFFG favours safety over liveness, the protocol's decisions are final. CFFG has similar properties to the Practical Byzantine Fault Tolerance (PBFT) consensus mechanism, in that both protocols use justification rounds and finalisation rounds to come to consensus [41]. CFFG also employs a method to batch justification and finalisation messages, which increases Ethereum's potential scalability. While the Ethereum network is functioning normally, GHOST will provide a fork resolution process, then CFFG will finalise the decision and add the block to Ethereum's Blockchain. However, in the event the network is under attack, or there is an issue that causes many nodes to go offline, GHOST will continue to function, and blocks will still be added to the Blockchain, but will not be finalised. Once the attack subsides, CFFG will start working again, and will finalise the blocks that GHOST has proposed, and add them to the Blockchain if they are valid. CFFG and GHOST cover each other's weaknesses, and allow a consensus mechanism with both safety, and liveness guarantees [42]. A partial snapshot of Ethereum's PoS implementaion is shown in Figure 7.

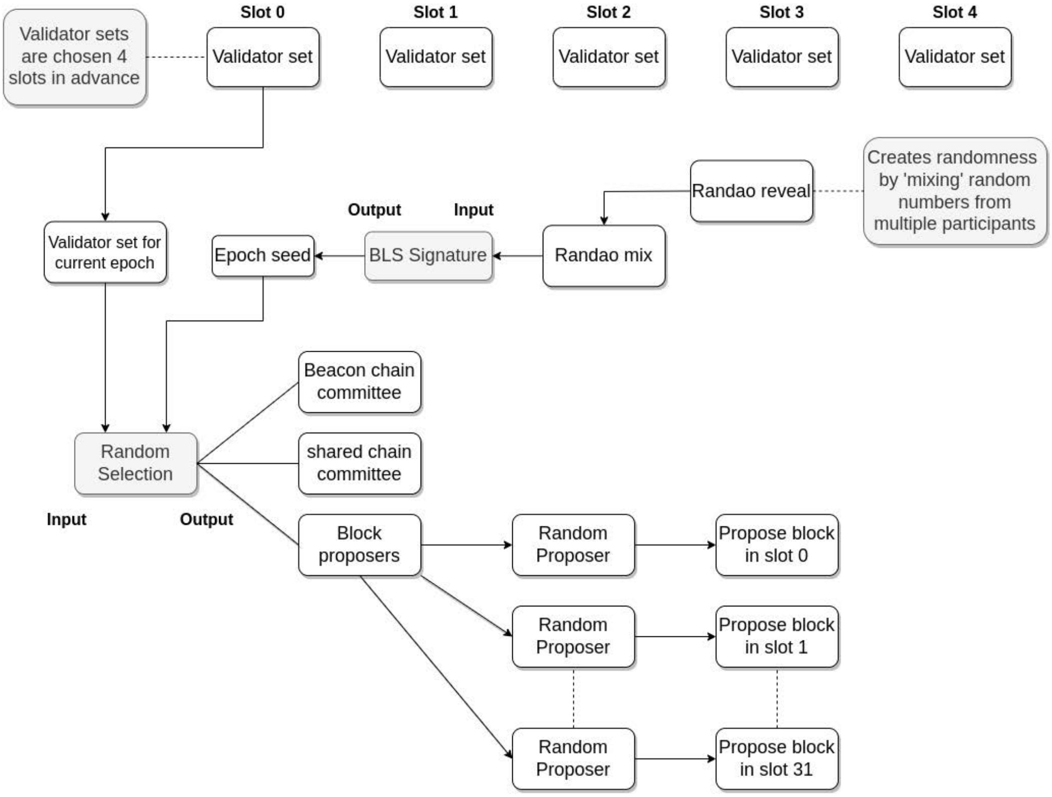


Figure 7. Ethereum PoS Consensus.

5.4. Microchain

Microchain proposes a lightweight consensus mechanism aimed at the IoT [43]. Microchain's consensus mechanism has similar properties to PoS, where a number of validators are selected to join a committee, and from the committee a node is selected to produce a block. The purpose of the committee is to select a pseudorandom subset of the network, to avoid biased or malicious block producers. Microchain also uses a committee called a 'Dynasty', where eligible validators are selected to join the committee. Microchain's consensus mechanism is broken down into two major components: Proof of Credit (PoC), and Voting based Chain Finality (VCF). PoC is a PoS mechanism that uses a credit weight to increase the chance a particular node has of producing a block as depicted in Figure 8. Given the distribution of credits in a particular Dynasty, nodes that have a higher credit weight have a larger chance of being selected to produce a block.

The VCF is a fork resolution tool, and is also responsible for extending the chain by adding new blocks, and protecting the Blockchain from malicious or accidental reorganising by adding checkpoints. The consensus mechanism proposed by Xu et al. leverages a VRF to power its slot selection (that is, the process of picking nodes to join a Dynasty). Microchain makes an assumption that networks are synchronous, and is able to provide two guarantees: persistence and liveness. Persistence guarantees that all the users agree on the same history of the Blockchain, and if one honest node finds a transaction to be finalised, all honest nodes see the transaction as final. Liveness guarantees that a valid transaction submitted by an honest node will eventually be added to a new block.

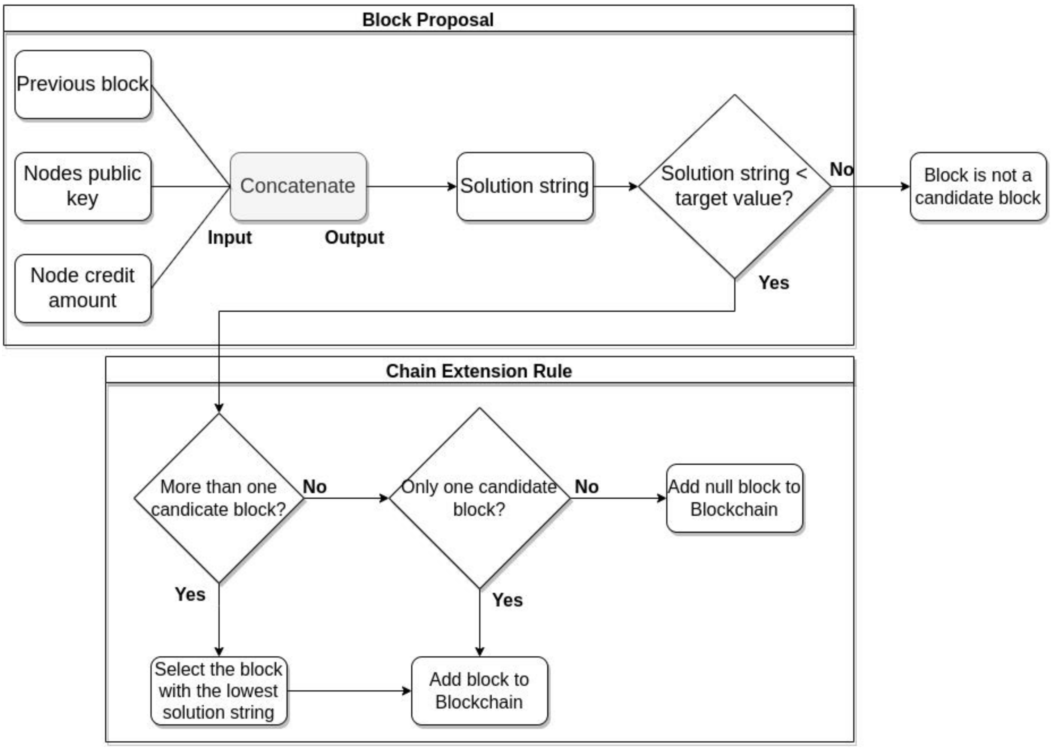

Block Proposal
Previous block
Nodes public key
Node credit amount
Concatenate
Input
Output
Solution string
Solution string < target value?
No
Block is not a candidate block
Yes
Chain Extension Rule
More than one candicate block?
No
Only one candidate block?
No
Add null block to Blockchain
Yes
Yes
Select the block with the lowest solution string
Add block to Blockchain

Figure 8. Microchain Consensus.

5.5. Proof of Supply Chain Share (PoSCS)

PoSCS is a consensus mechanism proposed by Tsang et al. targeted towards the Perishable Food Supply Chain (PFSC) [44]. The project uses a framework that incorporates an IoT network to manage monitoring and communication, a Blockchain to manage the data of food through the life cycle of the supply chain, and a database to archive supply chain information. The authors quickly point out that PoW is not suitable for the IoT due to the computationally expensive mining process. The authors propose a consensus mechanism like PoS, but replace the need for a currency, with a reputation system. Each node participating in consensus has four components that determine its reputation: Influence Factor (INF), Interest Factor (INT), Devotion Factor (DEV), and Satisfaction Factor (SAT). These factors can then be weighted using three strategies: the interest-first strategy, moderate-strategy, and devotion-first strategy. These weights prevent the consensus mechanism from favouring participants who attempt to maximise a single factor. Lastly, the shipment volume, considers the ingoing and outgoing volume a particular party is moving on the supply chain network. This process is summarised in Figure 9. These factors and weights are used to pseudorandomly select a block producer, who will be required to forge a block.

Block forgers are also required to do a small amount of PoW mining, which allows the block creation time to be controlled. Rather than all the nodes participating in PoSCSs consensus mechanism, only the block forger is required to mine. The architecture of the system uses a hybrid approach, combining Blockchain and the cloud. The Blockchain is used to record the data about a particular object in the supply chain in tandem with a traditional database. Once the object has completed its journey through the supply chain, the object is removed from the storage of the IoT devices, and remains archived in the cloud.

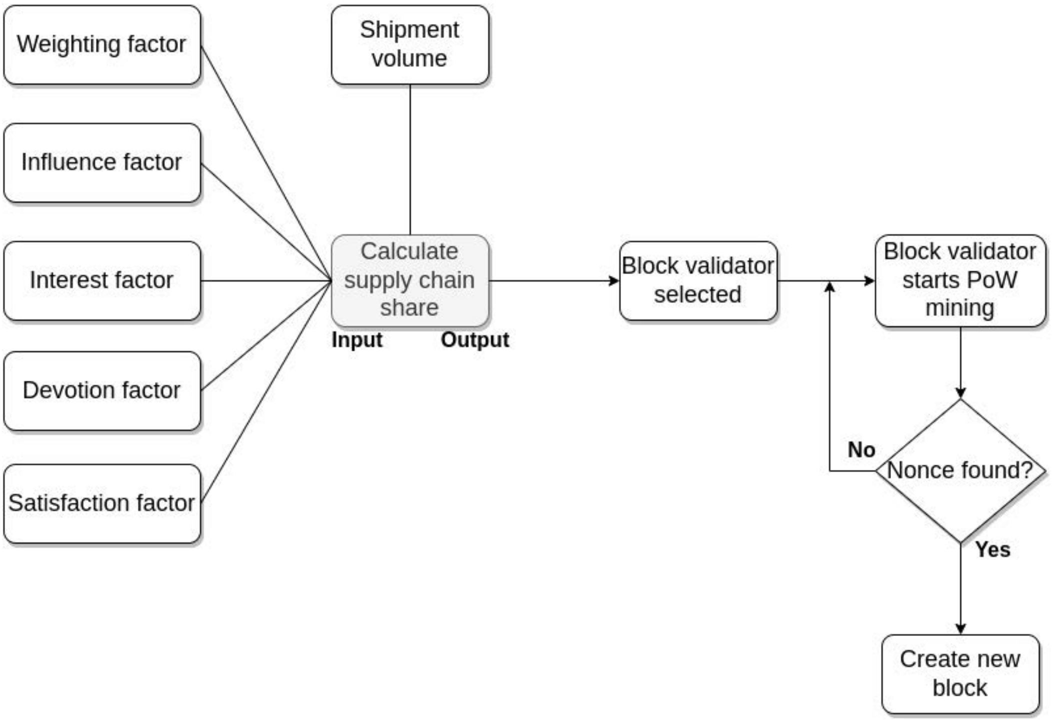


Figure 9. PoSCS Consensus.

5.6. Tendermint

A flexible consensus mechanism in the Byzantine Fault Tolerance (BFT) family that can be configured to work in either public, private, or permissioned networks [16]. Tendermint can be configured as a public consensus mechanism with PoS, or as a permissioned/private Blockchain with predetermined validator nodes. Tendermint's consensus mechanism uses a voting mechanism that has three steps: proposal, prevote and precommit. The proposal message is used by a proposer to suggest a particular value or state, while the prevote and precommit messages allow other nodes to vote on the proposal [45]. Tendermint uses a locking mechanism to guarantee consensus if the number of malicious nodes on the network does not surpass one-third of the total participants [12]. This locking mechanism uses the term 'polka', which checks that two-thirds of the prevotes are for a single block. If a validator tries to publish a block without a polka, it is considered malicious behaviour as shown in the 'Commit' phase in Figure 10. Cosmos is an example of a Blockchain that is leveraging the Tendermint consensus mechanism. Cosmos is a multichain Blockchain, which allows many independent Blockchains, called zones, to run in parallel, with the ability to communicate through a central Blockchain, called the hub. The native token on the Cosmos Blockchain is called Atom.

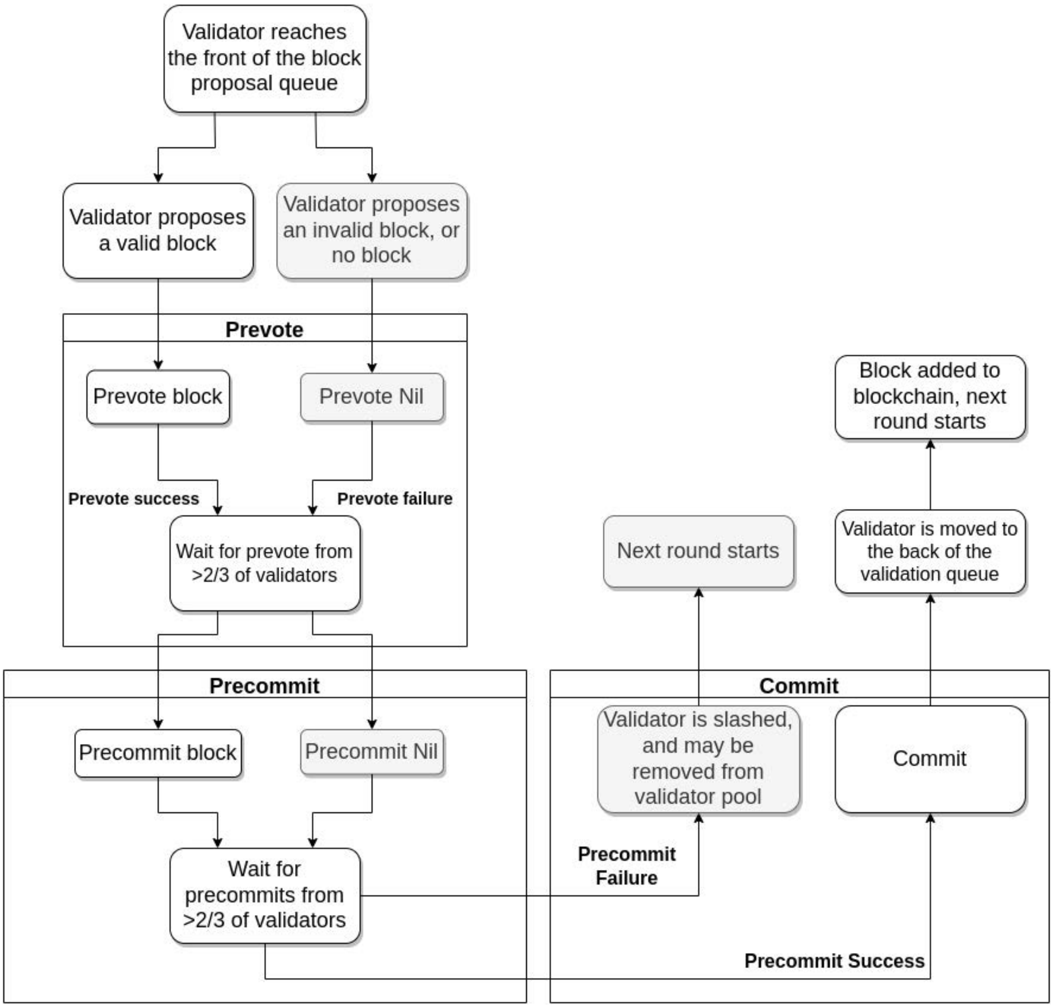

Validator reaches the front of the block proposal queue
Validator proposes a valid block
Validator proposes an invalid block, or no block
Prevote
Prevote block
Prevote Nil
Prevote success
Prevote failure
Wait for prevote from >2/3 of validators
Precommit
Precommit block
Precommit Nil
Wait for precommits from >2/3 of validators
Commit
Validator is slashed, and may be removed from validator pool
Commit
Precommit Failure
Precommit Success
Next round starts
Validator is moved to the back of the validation queue
Block added to blockchain, next round starts

Figure 10. Tendermint Consensus.

6. Alternative Consensus Mechanisms

The consensus mechanisms PoW and PoS are well-known and widely used in Blockchains and cryptocurrencies. In this section, well cover 3 consensus mechanisms that deviate from purely PoW and PoS. PoC creates consensus using hard drive capacity, PoI heavily integrates a reputation system into its consensus mechanism. Section 6 concludes by covering hybrid consensus mechanisms.

6.1. Proof of Capacity

A consensus mechanism that focuses on hard drive capacity, rather than mining with graphics cards or ASICs (Application-specific integrated circuit). Proof of Capacity saw its first use in the cryptocurrency BurstCoin. Mining on BurstCoin has two phases, plotting, and mining. Plotting involves hashing a list of nonce values, and then storing them on a hard drive. BustCoin uses the Shabal hashing algorithm, which is harder to hash than Bitcoins SHA256. Rather than discarding the hashes like in Bitcoin, they are bundled together into scoops (a pair of hashes), and stored on the nodes hard drive. In the mining phase, miners calculate a scoop number, and they use that scoop number to create a deadline value [46]. A node's deadline value will vary depending on the hashes that have been calculated, and will represent a time limit in seconds. A node that is able to calculate a deadline with the lowest time, is given the right to produce a block. An outline of this porocess can be found in Figure 11. BurstCoin has since rebranded to Signum, and has changed to a hybrid consensus mechanism called PoC+. PoC+ still requires a commitment of hard drive space, with miners now having the option to stake their Signa coins, which will increase their chance of mining a block.

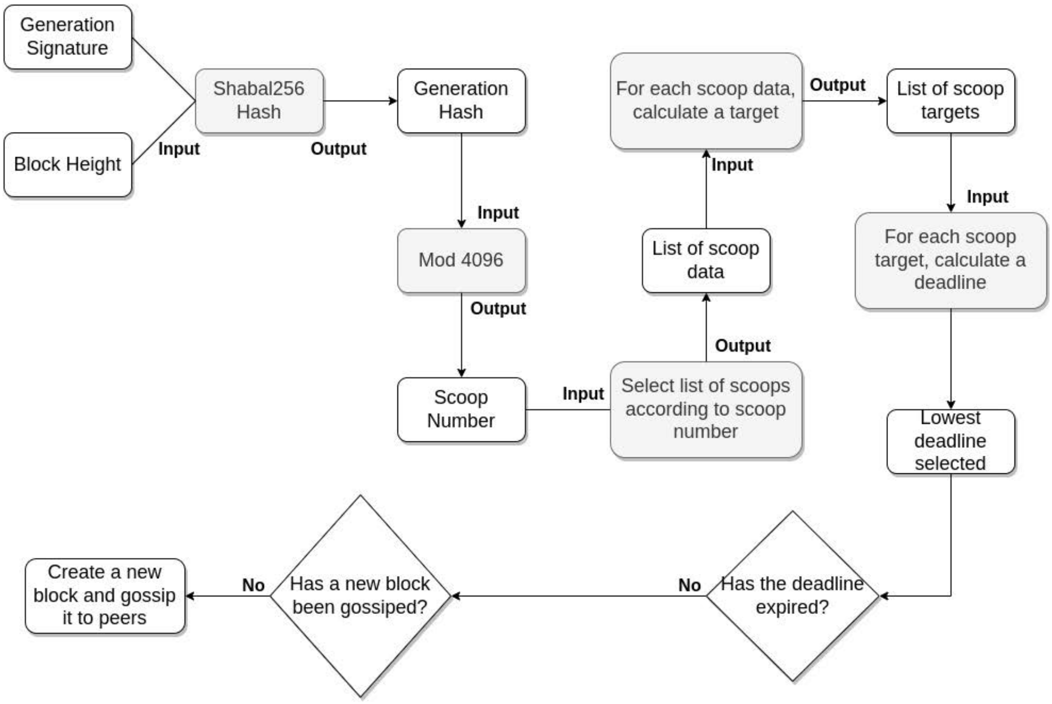

Generation Signature
Block Height
Input
Shabal256 Hash
Output
Generation Hash
Input
Mod 4096
Output
Scoop Number
Input
Select list of scoops according to scoop number
Output
List of scoop data
Input
For each scoop data, calculate a target
Output
List of scoop targets
Input
For each scoop target, calculate a deadline
Lowest deadline selected
Has the deadline expired?
No
Has a new block been gossiped?
No
Create a new block and gossip it to peers

Figure 11. PoC Consensus.

6.2. *Proof of Importance*

Proof of Importance (PoI) is a consensus mechanism originally proposed by the New Economy Movement (NEM). PoI shares similarities to PoS, where nodes are required to lock up a certain about of coins. However, rather than just keeping a node running like in the case of PoS, PoI has some extra requirements to encourage network usage and calculate a wallet's importance [47] as scene in Figure 12. To be selected for the importance calculation, NEM wallets must have a minimum of 10,000 coins vested for a certain period. An importance score can also be increased by using the NEM network and sending transactions. Safeguards have been put in place against loop attacks, which involves sending coins between accounts controlled by a single actor, to boost their importance [48]. NEM has added a mechanism that heavily weights the importance of an account sending NEM, and minimal weight to an account that sends many coins, but receives most or all of their NEM back. Even if an account were to attempt the loop attack, they gain a minor increase in their importance score (<10%) but gain very little monetarily, as the extra money that receive from their higher importance, is lost in transaction fees attempting to boost their importance [48]. The name of NEM's native Blockchain token is XEM.

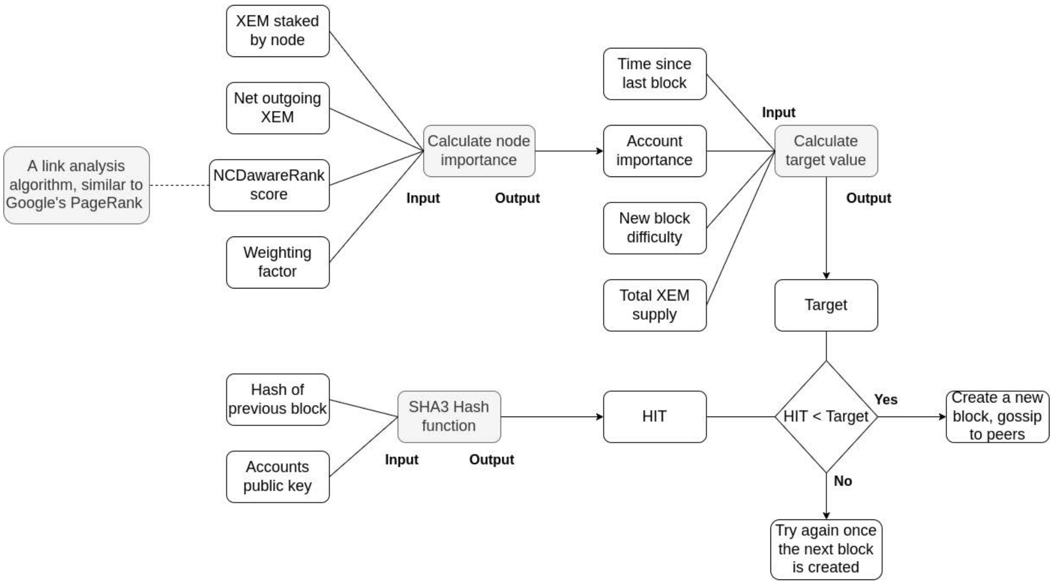


Figure 12. PoI Consensus.

6.3. Hybrid Consensus (PoW/PoS)

A number of cryptocurrencies have taken alternative approaches to consensus, by combining elements from PoW and PoS. Decred is a cryptocurrency that saw the flaws in PoW (double-spend problem) and the issues in PoS (nothing at stake) and decided to create a hybrid consensus mechanism to mitigate these problems as shown in Figure 13. Miners on the Decred network are still used to produce blocks, but are unable to add blocks directly to the Blockchain. Instead, miners propose their blocks to a network of PoS nodes who purchase tickets as their stake [49]. If a PoS node is pseudorandomly selected from this pool of tickets, they will be required to validate the block and add it to Decred's Blockchain, as shown in figure Figure 13. These improvements stop miners from creating private chains and adds a checkpoint system that stops large parts of the Blockchain from being reorganised in the event of an attack. The cryptocurrency Horizen also uses a Hybrid consensus mechanism. Horizen leverages a network of PoW miners, to solve a cryptographic puzzle. Horizen full nodes are still given a reward for running honestly but are not part of the consensus process. Horizen's 'Secure Nodes' provide a more secure version of standard full nodes which are found in other cryptocurrencies.

Horizen requires their Secure Nodes to use TLS encryption, hold a small number of tokens, and maintain a full copy of the Blockchain. Secure Node users are compensated with part of the block reward, providing a financial incentive to support the network [50]. In 2018, Horizen was a victim of a 51% attack (https://www.coindesk.com/markets/2018/06/08/blockchains-once-feared-51-attack-is-now-becoming-regular/ (accessed on 20 August 2022)), and decided to make a modification to their consensus mechanism to make future attacks more difficult. Horizen added a delay function, that penalises miners for keeping their private Blockchain hidden from the network. Malicious miners will be required to continue mining their Blockchain for a certain number of blocks, according to the delay function, rather than having honest nodes instantly adopt their modified Blockchain once it is made public [51]. This makes 51% attacks require more time to execute, and require more electricity, when comparing attacks to Horizen's original implementation of the LcR.

The consensus mechanisms discussed in Sections 4–6 have been conveniently summarised into 3 tables. Table 1 discusses common properties of consensus mechanisms, such as block time, Transactions Per Second (TPS), and adversary tolerance. Table 2 specficially discusses the consensus mechanisms designed for IoT devices (PoSCS, Microchain, PoE-WAL, and CBPoW) in more detail. Table 3 compares the discussed consensus mechnaisms against our criteria that was defined in Section 2, and allocates each consensus mechanisms a rating.

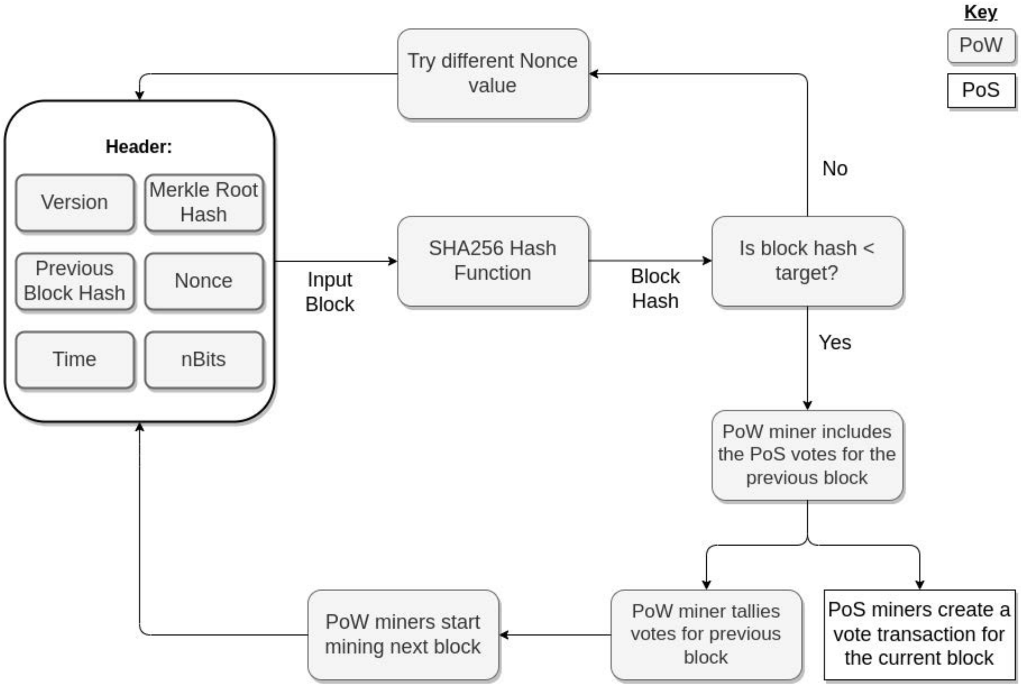

Key
PoW
PoS
Try different Nonce value
Header:
Version
Merkle Root Hash
Previous Block Hash
Nonce
Time
nBits
Input Block
SHA256 Hash Function
Block Hash
Is block hash < target?
No
Yes
PoW miner includes the PoS votes for the previous block
PoW miner tallies votes for previous block
PoS miners create a vote transaction for the current block
PoW miners start mining next block

Figure 13. Decred's hybrid PoW/PoS consensus mechanism.

Table 1. Overview of all the consensus mechanisms mentioned in Sections 4–6.

Consensus	Blockchain	Block Time	TPS	Adversary Tolerance	L2 Network	Reference
PoW	Bitcoin	10 min	7	<51%	Lightening Network	[52,53]
	Litecoin	2.5 min	56			[54]
	Monero	2 min	Variable		None	[55]
	Ethereum	12–14 s	15		Side Chains, Rollups	[56]
	Horizen	2.5 min	N/A		Side Chains	[50]
	CBPoW	Variable	500+		None	[24]
	PoEWAL	Variable	25		None	[25]
PoS	Ethereum (PoS)	12 s	TBD	<51%	TBD	[40]
	Algorand	4.5 s	1000	<33%	Off-chain Contracts	[57]
	Dfinity	Variable	Variable	<33%	None	[31]
	Cosmos	6 s	1000+	<33%		[58]
	PIVX	60 s	173	<51%		[55,59]
	Microchain	9 s	230+	<33%		[43]
	PoSCS	Variable	Variable	<51%		[44]
PoW + PoS	Decred	5 min	14	<51%	Lightening Network	[49]
PoC	BurstCoin	4 min	80+	<50%	None	[60]
PoI	NEM	1 min	4000	<51%		[48]

A Blockchains layer 1 network is the Blockchains primary network, where transactions are created on-chain. Transactions on layer 2 are created off-chain, and are often compressed and posted on a Blockchains layer 1 network to increase scalability. Note: Transactions Per Second (TPS)—Only considers a Blockchains layer 1 network.

Table 2. Overview of the IoT specific consensus mechanisms mentioned in Sections 4 and 5.

Consensus	Similar to	Decentralised	Features	Apps	Drawbacks	Reference
PoSCS	PoS	No	Reputation System	Supply Chains	Cloud Reliance	[44]
Microchain	PoS	Partially	Crypto Sortition	IoT Blockchain	Synchronous Networks	[43]
PoEWAL	PoW	Partially	Time-limited PoW	IoT Dapps	Synced Clocks	[25]
CBPoW	PoW		Credit System	Industrial IoT	DAG Coordinator	[24]

Table 3. Consensus mechanism suitability for IoT devices, measured against our criteria defined in Section 2.

Consensus	Processor Usage	Security	Decentralisation	Storage	TPS	Suitable?	Reference
PoW	High	High	High	High	Low	No	[16]
PoS	Medium	High	Medium	High	Variable	Partially	[16]
PoW + PoS	High	High	High	High	Low	No	[59]
PoC	Low	High	High	High	Low	No	[60]
PoI	Low	High	High	High	High	Partially	[48]
PoSCS	Low	High	Low	Low	Variable	Partially	[44]
CBPoW	Low	High	Medium	Low	Medium	Yes	[24]
PoEWAL	Low	High	High	High	low	Partially	[25]
Microchain	Medium	High	Medium	High	Medium	Yes	[43]

Storage refers to the internal memory needed to store the Blockchain on IoT devices. TPS Refers to the transactions per second of the consensus mechanism. less than 100 TPS is low, 100–1000 TPS is considered medium, and 1000+ is considered high TPS.

7. Analysis

Before starting the analysis, we will discuss the three trade-offs Blockchains commonly make. The Blockchain trilemma commonly effects the design choices of consensus mechanisms, and also has consequences related to IoT devices. After, we will discuss each consensus mechanism, and talk about their suitability for the IoT.

7.1. Blockchain Trilemma

Blockchains have three important properties: security, decentralisation, and scalability. Many Blockchains are only able to pick two of these properties, while having to compromise on the third [61]. The Blockchain trilemma was originally coined by Ethereum's creator, Vitalik Buterin. Buterin explains that, using these three properties, that simple (meaning with no advanced techniques, such as sharding [62]) Blockchains can broadly be placed into three categories: traditional Blockchains, high Transaction Per Second (TPS) Blockchains, and multichain Blockchains.

Bitcoin and Ethereum (pre-PoS Ethereum) are examples of traditional Blockchains. Bitcoin and Ethereum highly value decentralisation, and security, at the expense of scalability [63]. Blockchains that prioritise speed, and security, generally have a limited amount of nodes participating in consensus. Blockchains with Delegated Proof of Stake (DPoS) such as EoS [64] and TRON [65] are examples of Blockchains that prioritise performance. These Blockchains are able to process more transactions per second, but are more prone to centralisation due to a smaller number of nodes participating in consensus [66]. Blockchains such as COSMOS [58] and Avalanche [67] are two examples of multi-chain Blockchains. From the trilemma triangle, these sorts of Blockchains generally prioritise scalability and decentralisation. Buterin suggests that multichain Blockchains may not be able to provide certain security guarantees, when implementing more advanced techniques, such as sharding [61].

7.2. Proof of Work Suitability

PoW can be quickly discarded from the list of suitable consensus mechmanisms for IoT devices. PoW is extemely energy intensive [65], processor intensive, and requires specalised mining hardware. PoW is not suitable for the IoT.

7.3. Proof of Stake Suitability

PoS was given a suitability score of partial, and could potentially be implemented for the IoT. PoS has more desirable qualities than PoW for deployment in the IoT, as the energy and processor intensive nature required for consensus has been removed. However, PoS still has challenges for IoT usage:

1. Cryptography such as VRFs and the BLS signature scheme can be processor intensive, and may become problematic when working with resource constrained devices, especially as the network grows. Ethereum (PoS) Figure 7 and Dfinity Figure 6 are well-known to use these cryptographic functions in their consensus implementations.
2. PoS Blockchains are based on monetary concepts that involve a currency, which may not be suitable for some IoT applications. Arbitrary tokens could replace a monetary system, but this still comes with economic issues (who generates the tokens, how are tokens distributed, is the token amount capped etc.).
3. The transaction throughput of the network may not be adequate depending on the IoT use case, the PoS TPS varies dramatically from just over 100 TPS, to well over 1000 TPS as shown in Table 1. Selecting a particular PoS implementation will become important, as different PoS implementations have varying performance capabilities.

Due to these issues, PoS may be suitable for the IoT, but only under particular circumstances, such as where a more decentralisaed network is required.

7.4. Hybrid Consensus (PoW/PoS) Suitability

PoW/PoS hybrid consensus mechanisms are an alternative solution when attempting to solve the 51% attack problem and nothing at stake problem using a novel two consensus solution. Unfortunately, having PoW as a prerequisite brings all the same problems faced by PoW and its IoT suitability. Theoretically, the PoW porition of this hybrid consensus mechanism could be done offsite by powerful ASICS. The PoS portion could then be delegated to the IoT devices. We believe a configuration like the one mentioned above would be excessively complicated for consensus in the IoT and do not recommend Hybrid PoW/PoS for the IoT.

7.5. Proof of Capacity Suitability

PoC is another novel consensus mechanism that uses the concept of hard drive capacity to create consensus. Local storage on IoT devices is limited and would not be available for use in consensus. PoC is not suitable for use in the IoT.

7.6. Proof of Importance Suitability

This consensus mechanism takes concepts from PoS, and combines them with an importance mechanism. The higher a node's importance score, combined with the nodes total amount of staked coins, calculates their chance of being selected to mint a block. PoI satisfies many criteria in Table 1, which make it well suited for the IoT. Adoption of PoI consensus in Blockchains is limited, as NEM is the only Blockchain to implement the PoI consensus mechanism as seen on CoinMarketCap [68]. PoI is referenced in the literature, in surveys such as [16,69] giving a general description of the flow of the consensus mechanism. However, the authors are not aware of any papers or references that validate these claims and support the figures in Table 1. Until papers are developed that independently verify the characteristics of the PoI, we can only partially recommend the PoI consensus mechanism for the IoT.

7.7. PoSCS Suitability

PoSCS is a consensus mechanism that also uses elements of PoS. PoSCS uses a staking system, but replaces a monetary system with a reputation system based on how a node interacts with the supply chain. PoSCS also relies on the cloud to archive parts of the Blockchain that are not required to be stored on IoT devices, one of the few consensus mechanisms discussed here that addresses this problem. Based on the results of PoSCS [44], the transaction throughput may not be high enough for some IoT use cases, and the reliance on the cloud may not be suitable for some implementations. For these reasons, we think PoSCS is partially suitable, and may be usable for some IoT implementations.

7.8. CBPoW Suitability

One of the two consensus mechanisms that adapts PoW for the IoT. CBPoW dynamically adjusts the PoW mining difficulty for nodes, and actively punishes misbehaving nodes by making their PoW mining difficulty very high, to the point where mining is infesable. CBPoW also replaces a traditional Blockchain with a DAG, which has the ability to prune itself to reduce the sizde of the Blockchain stored locally on the device. This consensus mechanism performes well according to Table 1 with a throughput of 500 TPS. CBPoW has one central point of failure in current implementations (the coordinator, in the case of the IoTA DAG [70]) shown in Table 2. CBPoW has a number of characteristics that are suitable for the IoT including: a DAG Blockchain structure which has the ability to reduce its size, a lightweight PoW consensus mechanism and moderately high transaction throughput. Due to these 3 features that are favorable for IoT devices, we have labeled CBPoW as suitable in Table 3.

7.9. PoEWAL Suitability

PoEWAL consensus mechanism is also based on a modified version of PoW. In PoEWAL, the PoW mining process is time limited. Devices have a short amount to mine a block, significantly reducing power and processor usage. PoEWAL also makes the assumption that devices all have synced clocks, which is probably a reasonable assumption for IoT devices on a wireless sensor network collecting time series data. This requirement may unsuitable for some implementations, as IoT devices use commodity parts and are generally more susceptible to drifting out of sync [71]. PoEWAL has two limitations: low transaction throughput and reliance on sycnhronised clocks. Due to these limitations, we have labeled PoEWAL as only partially suitable on Table 3.

7.10. Microchain Suitability

Microchain has adapted concepts from PoS and made them more suitable for IoT usage. Nodes use credit amounts, rather than a monetary system, and Microchain has a block selection process to accommodate this change in PoS. Microchain also uses VRF cryptography to power its consensus mechanism, which may incur high processor usage on IoT devices as the network grows. Microchain also made some assumptions around network environments (synchronous networks in Table 2), which makes this consensus mechanism unusable for public Blockchains. Microchain is also shown to have reasonable performance, managing 230 TPS in their tests. Microchain has a number of features that make it suitable for IoT devices:

- A PoS implementation not dependent on a monetary system
- A moderately fast transaction trhougput
- Processor usage which remains low on controlled private networks

Due to these features which favour IoT devices, Microchain has been labeled as suitable in Table 3.

8. Conclusions

In this paper, we discussed resource constrained IoT devices, and the limitations of the current consensus mechanisms on the IoT. We started the discussion by defining a criterion to rank the consensus mechanisms against, including speed, security, decentralisation, and others. The discussion continued by individually describing a number of consensus mechanisms and defining their general flow through a series of figures. We described well-known consensus mechanisms such as PoW and PoS, but also discuss 4 IoT consensus mechanisms purpose built for the IoT (CBPoW, Microchain, PoEWAL, and PoSCS). These IoT focused consensus mechanisms make modifications to the already existing PoW and PoS consensus, but remove the need for energy inefficient mining and monetary systems respectfully.

In our analysis, discuss the advantages and disadvantages of each consensus mechanism, and their suitability for the IoT. The results from our analysis show that Microchain and CBPoW are suitable for the IoT. Microchain shows that it has suitable performace for the IoT in private environments, and CBPoW addresses problems around local Blockchain storage on IoT devices. Microchain and CBPoW have been labeled suitable in our analysis. Four other consensus mechanisms where also labeled partially suitable, including: PoEWAL, PoSCS, PoI and PoS. Some of the partially recommended consensus mechanisms addressed issues with monetary systems, computational overhead, and storage requirements. However, other issues where introduced, such as cloud reliance, synchronised clocks and poor and unverified performance claims.

Future Work

The current trend around consensus research in academia, and in industry, has been focused on making mechanisms that are lightweight enough for low powered devices, and reducing the carbon footprint of the mining process. Our future research will look more closely into novel consensus mechanisms. Specifically, those that can meet the unique requirements of an IoT device, and mechanisms that can be deployed to meet business specific goals in both private, and potentially public operating environments.

Author Contributions: Conceptualization, Z.A. and N.C.; investigation, Z.A.; writing—original draft preparation, Z.A.; writing—review and editing, N.C., R.A., W.H.; supervision, N.C., R.A. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the SmartSat C.R.C., whose activities are funded by the Australian Government's C.R.C. Program.

Acknowledgments: The authors would like to thank SmartSat CRC for providing scholarship funding, and BAE Systems Australia, our industry collaborator.

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

References

1. Singh, S.; Sharma, P.K.; Yoon, B.; Shojafar, M.; Cho, G.H.; Ra, I.H. Convergence of blockchain and artificial intelligence in IoT network for the sustainable smart city. *Sustain. Cities Soc.* **2020**, *63*, 102364. [CrossRef]
2. Min, H. Blockchain technology for enhancing supply chain resilience. *Bus. Horizons* **2019**, *62*, 35–45. [CrossRef]
3. Dujak, D.; Sajter, D. Blockchain Applications in Supply Chain. In *SMART Supply Network*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 21–46.
4. Sternberg, H.S.; Hofmann, E.; Roeck, D. The Struggle is Real: Insights from a Supply Chain Blockchain Case. *J. Bus. Logist.* **2021**, *42*, 71–87. [CrossRef]
5. Casado-Vara, R.; Prieto, J.; Prieta, F.D.L.; Corchado, J.M. How blockchain improves the supply chain: Case study alimentary supply chain. *Procedia Comput. Sci.* **2018**, *134*, 393–398. [CrossRef]
6. Werner, R.; Lawrenz, S.; Rausch, A. Blockchain Analysis Tool of a Cryptocurrency. In *PervasiveHealth: Pervasive Computing Technologies for Healthcare*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 80–84.
7. Hopkins, A.L.; Lala, J.H.; Smith, T.B. *The Evolution of Fault Tolerant Computing at the Charles Stark Draper Laboratory, 1955–85*; Springer: Berlin/Heidelberg, Germany, 1987; pp. 121–140.

8. Corbett, J.C.; Dean, J.; Epstein, M.; Fikes, A.; Frost, C.; Furman, J.J.; Ghemawat, S.; Gubarev, A.; Heiser, C.; Hochschild, P.; et al. Spanner: Google's Globally-Distributed Database. *ACM Trans. Comput. Syst. (TOCS)* **2013**, *31*, 1–21. [CrossRef]
9. Gupta, A.; Yang, F.; Govig, J.; Kirsch, A.; Chan, K.; Lai, K.; Wu, S.; Dhoot, S.; Kumar, A.; Agiwal, A.; et al. Mesa: Geo-Replicated, Near Real-Time, Scalable Data Warehousing. *Proc. Vldb Endow.* **2014**, *7*, 1259–1270. [CrossRef]
10. Nakamoto, S. *Bitcoin: A Peer-to-Peer Electronic Cash System*; Portal Unicamp: Campinas, Brazil, 2008.
11. Lamport, L.; Shostak, R.; Pease, M. The Byzantine generals problem. *ACM Trans. Program. Lang. Syst.* **2019**, *4*, 382–401. [CrossRef]
12. Buchman, E. Tendermint: Byzantine Fault Tolerance in the Age of Blockchains. Ph.D. Thesis, The University of Guelph, Guelph, ON, Canada, 2016.
13. Helliar, C.V.; Crawford, L.; Rocca, L.; Teodori, C.; Veneziani, M. Permissionless and permissioned blockchain diffusion. *Int. J. Inf. Manag.* **2020**, *54*, 102136. [CrossRef]
14. Polge, J.; Robert, J.; Traon, Y.L. Permissioned blockchain frameworks in the industry: A comparison. *ICT Express* **2021**, *7*, 229–233. [CrossRef]
15. Yang, R.; Wakefield, R.; Lyu, S.; Jayasuriya, S.; Han, F.; Yi, X.; Yang, X.; Amarasinghe, G.; Chen, S. Public and private blockchain in construction business process and information integration. *Autom. Constr.* **2020**, *118*, 103276. [CrossRef]
16. Salimitari, M.; Chatterjee, M.; Fallah, Y.P. A survey on consensus methods in blockchain for resource-constrained IoT networks. *Internet Things* **2020**, *11*, 100212. [CrossRef]
17. Conti, M.; Sandeep, K.E.; Lal, C.; Ruj, S. A survey on security and privacy issues of bitcoin. *IEEE Commun. Surv. Tutorials* **2018**, *20*, 3416–3452. [CrossRef]
18. Anceaume, E.; Lajoie-Mazenc, T.; Ludinard, R.; Sericola, B. Safety analysis of Bitcoin improvement proposals. In Proceedings of the 2016 IEEE 15th International Symposium on Network Computing and Applications, NCA 2016, Boston, MA, USA, 31 October–2 November 2016; pp. 318–325.
19. Decker, C.; Seidel, J.; Wattenhofer, R. Bitcoin meets strong consistency. In Proceedings of the 17th International Conference on Distributed Computing and Networking, Singapore, 4–7 January 2016.
20. Shi, E. Analysis of deterministic longest-chain protocols. In Proceedings of the 2019 IEEE 32nd Computer Security Foundations Symposium (CSF), Hoboken, NJ, USA, 25–28 June 2019; pp. 122–135.
21. Antonopoulos, A.M.; Wood, G. *Mastering Ethereum: Building Smart Contracts and Dapps*; O'Reilly Media: Sebastopol, CA, USA, 2018.
22. Alwen, J.; Chen, B.; Pietrzak, K.; Reyzin, L.; Tessaro, S. Scrypt is maximally memory-hard. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Springer: Berlin/Heidelberg, Germany, 2017; Volume 10212 LNCS, pp. 33–62.
23. Jamtel, E.L. Swimming in the Monero pools. In Proceedings of the 11th International Conference on IT Security Incident Management and IT Forensics, IMF 2018, Hamburg, Germany, 7–9 May 2018; pp. 110–114.
24. Huang, J.; Kong, L.; Chen, G.; Wu, M.Y.; Liu, X.; Zeng, P. Towards secure industrial iot: Blockchain system with credit-based consensus mechanism. *IEEE Trans. Ind. Inform.* **2019**, *15*, 3680–3689. [CrossRef]
25. Raghav; Andola, N.; Venkatesan, S.; Verma, S. PoEWAL: A lightweight consensus mechanism for blockchain in IoT. *Pervasive Mob. Comput.* **2020**, *69*, 101291. [CrossRef]
26. King, S.; Nadal, S. Ppcoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake. Available online: https://bitcoin.peryaudo.org/vendor/peercoin-paper.pdf (accessed on 20 August 2022).
27. Zhang, S.; Lee, J.H. Analysis of the main consensus protocols of blockchain. *ICT Express* **2020**, *6*, 93–97. [CrossRef]
28. Gilad, Y.; Hemo, R.; Micali, S.; Vlachos, G.; Zeldovich, N. Algorand: Scaling Byzantine Agreements for Cryptocurrencies. In Proceedings of the 26th Symposium on Operating Systems Principles, Shanghai, China, 28–31 October 2017.
29. Badertscher, C.; Gaži, P.; Kiayias, A.; Russell, A.; Zikas, V. Ouroboros genesis: Composable proof-of-stake blockchains with dynamic availability. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, ON, Canada, 15–19 October 2018; pp. 913–930.
30. Galindo, D.; Liu, J.; Ordean, M.; Wong, J.M. Fully Distributed Verifiable Random Functions and their Application to Decentralised Random Beacons. *IACR Cryptol. EPrint Arch.* **2020**, *2020*, 96.
31. Hanke, T.; Movahedi, M.; Williams, D. DFINITY Technology Overview Series, Consensus System. *arXiv* **2018**, arXiv:1805.04548.
32. Boneh, D.; Lynn, B.; Shacham, H. Short Signatures from the Weil Pairing. In *Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2001; Volume 2248, pp. 514–532.
33. Buterin, V. *A Next-Generation Smart Contract and Decentralized Application Platform*; Ethereum Foundation: Bern, Switzerland, 2014.
34. Wood, G. *Ethereum: A Secure Decentralised Generalised Transaction Ledger*; Ethereum Foundation: Bern, Switzerland, 2022.
35. Angeris, G.; Kao, H.T.; Chiang, R.; Noyes, C.; Chitra, T. An analysis of Uniswap markets. *Cryptoecon. Syst.* **2019**, *1*. [CrossRef]
36. Schär, F. Decentralized Finance: On Blockchain- and Smart Contract-Based Financial Markets. *Fed. Reserve Bank St. Louis Rev.* **2021**, *103*, 153–174. [CrossRef]
37. Scholten, O.J.; Gerard, N.; Hughes, J.; Deterding, S.; Drachen, A.; Walker, J.A.; Zendle, D. Ethereum Crypto-Games: Mechanics, Prevalence and Gambling Similarities. In Proceedings of the Annual Symposium on Computer-Human Interaction in Play, Barcelona, Spain, 22–25 October 2019; Association for Computing Machinery: New York, NY, USA, 2019.
38. Peterson, J.; Krug, J.; Zoltu, M.; Williams, A.K.; Alexander, S. Augur: A decentralized oracle and prediction market platform (v2. 0). *arXiv* **2021**, arXiv:1501.01042.

39. Mohanty, D. Ethereum: What Lies Ahead. In *Ethereum for Architects and Developers*; Apress: New York, NY, USA, 2018; pp. 245–258.
40. Buterin, V.; Hernandez, D.; Kamphefner, T.; Pham, K.; Qiao, Z.; Ryan, D.; Sin, J.; Wang, Y.; Zhang, Y.X. Combining GHOST and Casper. *arXiv* **2020**, arXiv:2003.03052.
41. Buterin, V.; Griffith, V. Casper the Friendly Finality Gadget. *arXiv* **2017**, arXiv:1710.09437.
42. Beekhuizen, C. Validated, Staking on Eth2: #2—Two Ghosts in a Trench Coat. Available online: https://blog.ethereum.org/2020/02/12/validated-staking-on-eth2-2-two-ghosts-in-a-trench-coat (accessed on 20 August 2022).
43. Xu, R.; Chen, Y.; Blasch, E.; Chen, G. Microchain: A Hybrid Consensus Mechanism for Lightweight Distributed Ledger for IoT. *arXiv* **2019**, arXiv:1909.10948.
44. Tsang, Y.P.; Choy, K.L.; Wu, C.H.; Ho, G.T.S.; Lam, H.Y. Blockchain-Driven IoT for Food Traceability with an Integrated Consensus Mechanism. *IEEE Access* **2019**, *7*, 129000–129017. [CrossRef]
45. Buchman, E.; Kwon, J.; Tendermint, Z.M. The latest gossip on BFT consensus. *arXiv* **2018**, arXiv:1807.04938.
46. Andrey, A.; Petr, C. Review of Existing Consensus Algorithms Blockchain. In Proceedings of the 2019 IEEE International Conference Quality Management, Transport and Information Security, Information Technologies IT and QM and IS 2019, Sochi, Russia, 23–27 September 2019; pp. 124–127.
47. Wen, Y.; Lu, F.; Liu, Y.; Cong, P.; Huang, X. Blockchain Consensus Mechanisms and Their Applications in IoT: A Literature Survey. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Springer: Berlin/Heidelberg, Germany, 2020; Volume 12454 LNCS, pp. 564–579.
48. Nem Foundation. NEM Technical Reference. Available online: https://www.cryptoground.com/nem-white-paper (accessed on 20 August 2022).
49. Jepson, C. DTB001: Decred Technical Brief. Available online: https://decred.org/dtb001.pdf (accessed on 20 August 2022).
50. Viglione, R.; Versluis, R.; Lippencott, J. Zen White Paper. Available online: https://www.horizen.io/assets/files/Zen-White-Paper.pdf (accessed on 20 August 2022).
51. Garoffolo, A.; Viglione, R. Sidechains: Decoupled Consensus between Chains. *arXiv* **2018**, arXiv:1812.05441.
52. Bhaskar, N.D.; Chuen, D.L.K. Bitcoin Mining Technology. In *Handbook of Digital Currency: Bitcoin, Innovation, Financial Instruments, and Big Data*; Academic Press: Cambridge, MA, USA, 2015; pp. 45–65.
53. Gobel, J.; Krzesinski, A.E. Increased block size and Bitcoin blockchain dynamics. In Proceedings of the 2017 27th International Telecommunication Networks and Applications Conference, ITNAC 2017, Melbourne, Australia, 22–24 November 2017; pp. 1–6.
54. Boshuis, S.; Braam, T.; Marchena, A.P.; Jansen, S. The Effect of Generic Strategies on Software Ecosystem Health: The Case of Cryptocurrency Ecosystems. In Proceedings of the 2018 IEEE/ACM 1st International Workshop on Software Health (SoHeal), Gothenburg, Sweden, 27 May 2018.
55. Lee, J.H. Rise of Anonymous Cryptocurrencies: Brief Introduction. *IEEE Consum. Electron. Mag.* **2019**, *8*, 20–25. [CrossRef]
56. Bach, L.M.; Mihaljevic, B.; Zagar, M. Comparative analysis of blockchain consensus algorithms. In Proceedings of the 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics, Opatija, Croatia, 21–25 May 2018; pp. 1545–1550.
57. Esgin, M.F.; Kuchta, V.; Sakzad, A.; Steinfeld, R.; Zhang, Z.; Sun, S.; Chu, S. Practical Post-quantum Few-Time Verifiable Random Function with Applications to Algorand. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Springer: Berlin/Heidelberg, Germany, 2021; Volume 12675 LNCS, pp. 560–578.
58. Kwon, J.; Buchman, E. Cosmos Whitepaper. Available online: https://v1.cosmos.network/resources/whitepaper (accessed on 20 August 2022).
59. Baudlet, M.; Fall, D.; Taenaka, Y.; Kadobayashi, Y. The Best of Both Worlds: A New Composite Framework Leveraging PoS and PoW for Blockchain Security and Governance. In Proceedings of the 2020 2nd Conference on Blockchain Research and Applications for Innovative Networks and Services, BRAINS 2020, Paris, France, 28–30 September 2020; pp. 17–24.
60. Gauld, S.; von Ancoina, F.; Stadler, R. The Burst Dymaxion. Available online: https://www.allcryptowhitepapers.com/wp-content/uploads/2018/05/Burst-Coin-whitepaper.pdf (accessed on 20 August 2022).
61. Buterin, V. Why Sharding Is Great: Demystifying the Technical Properties. Available online: https://vitalik.ca/general/2021/04/07/sharding.html (accessed on 20 August 2022).
62. Zamani, M.; Movahedi, M.; Raykova, M. Rapidchain: Scaling blockchain via full sharding. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, ON, Canada, 15–19 October 2018; pp. 931–948.
63. Kiayias, A.; Panagiotakos, G. Speed-Security Tradeoffs in Blockchain Protocols. *IACR Cryptol. EPrint Arch.* **2015**, *2015*, 1019.
64. Xu, B.; Luthra, D.; Cole, Z.; Blakely, N. EOS: An Architectural, Performance, and Economic Analysis. Available online: https://blog.bitmex.com/wp-content/uploads/2018/11/eos-test-report.pdf (accessed on 20 August 2022).
65. Li, J.; Li, N.; Peng, J.; Cui, H.; Wu, Z. Energy consumption of cryptocurrency mining: A study of electricity consumption in mining cryptocurrencies. *Energy* **2019**, *168*, 160–168. [CrossRef]
66. Al-Saqaf, W.; Seidler, N. Blockchain technology for social impact: Opportunities and challenges ahead. *J. Cyber Policy* **2017**, *2*, 338–354. [CrossRef]
67. Tanana, D. Avalanche blockchain protocol for distributed computing security. In Proceedings of the 2019 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), Sochi, Russia, 3–6 June 2019.
68. CoinMarketCap. Top POI Tokens by Market Capitalization. Available online: https://coinmarketcap.com/view/poi/ (accessed on 20 August 2022).

69. Zhang, P.; Schmidt, D.C.; White, J.; Dubey, A. Chapter Seven—Consensus mechanisms and information security technologies. *Adv. Comput.* **2019**, *15*, 181–209.
70. Silvano, W.F.; Marcelino, R. Iota Tangle: A cryptocurrency to communicate Internet-of-Things data. *Future Gener. Comput. Syst.* **2020**, *112*, 307–319. [CrossRef]
71. Tirado-Andrés, F.; Rozas, A.; Araujo, A. A Methodology for Choosing Time Synchronization Strategies for Wireless IoT Networks. *Sensors* **2019**, *19*, 3476. [CrossRef] [PubMed]