applied sciences

MDPI

Article

CBCIoT: A Consensus Algorithm for Blockchain-Based IoT Applications

Moin Uddin [1], Muhammad Muzammal [1], Muhammad Khurram Hameed [1], Ibrahim Tariq Javed [1,*], Bandar Alamri [2] and Noel Crespi [3]

[1] Department of Computer Science, Bahria University Islamabad Campus, Islamabad 44000, Pakistan; engrmoin.uett@gmail.com (M.U.); mmuzammal.buic@bahria.edu.pk (M.M.); muh.khurramhameed@gmail.com (M.K.H.)
[2] Lero-The Irish Software Research Centre, University of Limerick, V94 T9PX Limerick, Ireland; Bandar.Alamri@ul.ie
[3] Institut Polytechnique de Paris Telecom SudParis Evry, Courcouronnes FR, 9 Rue Charles Fourier, 91000 Evry, France; noel.crespi@mines-telecom.fr
* Correspondence: Ibrahimtariq.javed@lero.ie

Abstract: Internet of things is widely used in the current era to collect data from sensors and perform specific tasks through processing according to the requirements. The data collected can be sent to a blockchain network to create secure and tamper-resistant records of transactions. The combination of blockchain with IoT has huge potential as it can provide decentralized computation, storage, and exchange for IoT data. However, IoT applications require a low-latency consensus mechanism due to its constraints. In this paper, CBCIoT, a consensus algorithm for blockchain-based IoT applications, is proposed. The primary purpose of this algorithm is to improve scalability in terms of validation and verification rate. The algorithm is developed to be compatible with IoT devices where a slight delay is acceptable. The simulation results show the proposed algorithm's efficiency in terms of block generation time and transactions per second.

Keywords: blockchain; Internet of Things; consensus algorithm; proof of work; proof of stake; stellar consensus protocol

Citation: Uddin, M.; Muzammal, M.; Hameed, M.K.; Javed, I.T.; Alamri, B.; Crespi, N. CBCIoT: A Consensus Algorithm for Blockchain-Based IoT Applications. *Appl. Sci.* **2021**, *11*, 11011. https://doi.org/10.3390/app112211011

Academic Editors: Gianluca Lax and Antonia Russo

Received: 15 October 2021
Accepted: 16 November 2021
Published: 20 November 2021

1. Introduction

Blockchain is a distributed database which provides decentralization and immutability of the transactions or records in a peer-to-peer network. On the other hand, IoT consists of physical devices that are connected to the internet to collect and share data. Combination of blockchain and IoT i.e. blockchain of IoT (BCIoT) will be beneficial to the world in terms of decentralization and immutability but challenges are also present with opportunities. The volume of data collected is currently tremendous and grows due to the growing number of IoT devices. Large number of IoT devices and data platforms open the door to new applications and use cases. However, IoT data security is a major concern that has slowed the technology's adoption as they are a prime target for a range of attacks. Scalability is another issue with today's IoT networks. When the number of devices linked through an IoT network rises, current centralized techniques for authenticating and linking sensor nodes in a network will become a bottleneck. If the server that handles the vast volume of data exchange goes down, the entire network can go down.

Blockchain is considered to be a game-changing technology that has the potential to address IoT security and scalability concerns [1,2]. Blockchain is a distributed ledger technology that can be used to distribute and access data in a secure and decentralized manner [3]. Every transaction may be authenticated to avoid conflicts and build trust among all network participants. However, the integration of IoT and blockchain gives rise to new challenges (Figure 1), such as scalability, big data of IoT devices, security and

privacy that need to be resolved [4]. It is also necessary for blockchain and IoT to implement an encryption algorithm but both ecosystems have different computing capabilities; so, processing time and power [5] could be a great difficulty. Blockchain provides decentralization, stores transaction on every node's ledger and its size is continuously increasing with time. IoT devices have very low storage capacity [6], as this is beyond the capabilities of sensors. IoT is an open nature network and so in BCIoT, security and privacy could be a potential threat. An efficient authentication scheme by message authentication for Internet of Vehicles (EASSAIV) is proposed in [7] to resolve security and privacy concerns.

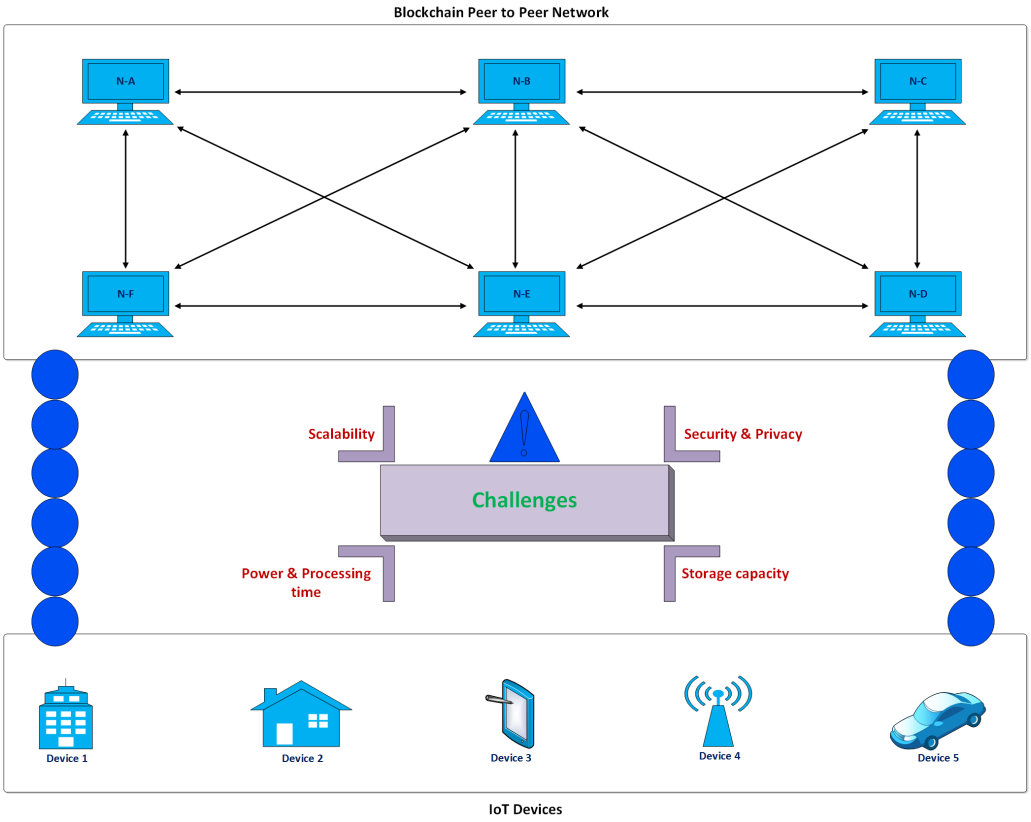

Figure 1. Blockchain IoT challenges.

In blockchain, the consensus is a process of achieving agreement on a single value in a distributed computing environment. The two most popular consensus algorithms are proof of work (PoW) and proof of stake (PoS). However, these algorithms cannot be directly used for IoT scenarios due to their scalability issues [8]. Lightweight consensus algorithms are required for IoT to adopt blockchain because current blockchain platforms such as Bitcoin and Ethereum blockchains are computationally expensive, having high bandwidth overheads and delays for IoT [9]. So, it must be ensured to solve this issue by making a suitable consensus protocol by consulting IoT problems, such as lack of security, different standards of devices, less memory of devices, and a large amount of data. Security and privacy is major concern of the IoT field but if blockchain is involved, then it has to be addressed. A novel blockchain-based approach reported in [10] is used for IoT and suggested that distributed blockchain can solve security concerns for IoT. A decentralized identity management system based on blockchain was presented by [11] which provides security and privacy of the patients for remote healthcare. An idea proposed in [12] is to preserve privacy using blockchain for medical IoT. The storage issue was addressed in [13] for industrial IoT by providing hierarchical blockchain storage structure (ChainSplitter) in which most of the blockchain is stored on the clouds. Blockchain of IoT needs lightweight algorithms to overcome the power and processing time challenges. A lightweight consensus

algorithm proof of block and trade (PoBT) is proposed in [14] for BCIoT, which reduces computation time.

In this study, the scalability problem for BCIoT applications is addressed. A very comprehensive study [15] for blockchain IoT explained most of the BCIoT problems by using a mindmap in which scalability is dependent on throughput, block size and transaction speed. This mindmap also tells that throughput for BCIoT can be achieved by proper use of a consensus algorithm. In [16], a strategical approach was used to build a consensus algorithm. It needs attention before the implementation of blockchain in the IoT field. Performance, limitations, and challenges were discussed based on more than one hundred studies for IoT, and protocols which were carefully reviewed before using them for BCIoT.

In the IoT wireless sensor network (WSN), many sensors continuously collect vast amounts of data from the environment and send them to their central processing unit as shown in Figure 2. Blockchain has to validate these data more efficiently. So, several validations per second should be effective in BCIoT. The primary purpose of this study is to make a scalable consensus protocol for BCIoT.

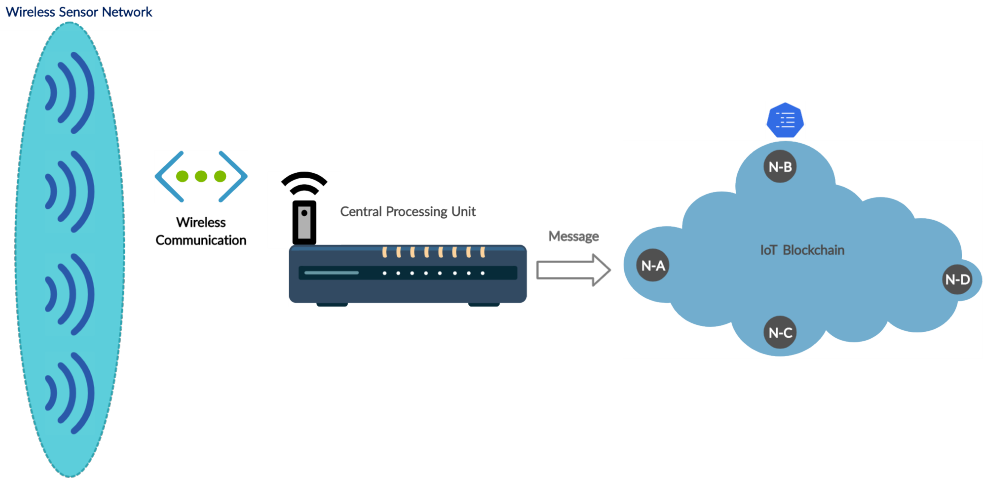

Figure 2. Blockchain for IoT wireless sensor network.

The rest of the paper is structured as follows: the related work is described in Section 2 highlighting the challenges and adopting the existing consensus algorithm for an IoT scenario. The blockchain overview is presented in Section 3 that illustrates the working, consensus, types, and some key terms of blockchain. A novel Consensus algorithm (CB-CIoT) is presented in Section 4 by making suitable changes according to the case study. An analysis is carried out in Section 5 on the proposed algorithm by using flow charts, and results are concluded with limitations. After the analysis and results, conclusions and future work are presented in Section 6.

2. Related Work

This section explains scalability and TPS (Transactions per second) problems, which are improved in cryptocurrencies by consensus algorithms. Different consensus algorithms are the core part of this section; so, a comparison between them is necessary to understand before moving further.

2.1. Scalability Problem

Applying blockchain protocols to IoT can lead us to a new set of problems due to their massive computational loads. Blockchain size grows in IoT with an increase in the number of connected devices, which generates a tremendous amount of data in real-time. This is a major difficulty for IoT blockchain to validate. Many implementations of blockchain in the current cryptocurrencies market are not so scalable according to our required scenario.

In [8], scalability problem was addressed and it was mentioned that this could be the potential barrier for BCIoT because of faster TPS.

2.2. Importance of Consensus

Consensus is very important in the blockchain. Over the period of time, many consensus algorithms have been developed to solve scalability issues. As discussed earlier, the development of blockchain in digital currencies is enormous, so many algorithms are used in this field. Proof of work (PoW) was the first consensus algorithm developed for Bitcoin by Satoshi Nakamoto in 2008 [17]. However, it is believed that this technology was conceived from Hash Cash [18] of Adam Back. He used PoW in his study before the cryptocurrency days. PoW is discussed in this section to understand the consensus in blockchain. A block created in this algorithm takes 10 min due to a large amount of computation power, and if the nodes create the block faster than other nodes, then they get the reward plus a transaction fee. It means that in PoW, the miner requires extra hashing power to get a reward; otherwise, his used resources will be wasted. Selfish mining is another problem [19] with PoW-based blockchain, but practically, it is very hard to get 51% hashing power for a mining pool. A large amount of hashing power is required for PoW to solve a complex problem, make it valuable, and cause an increase in Bitcoin price.

Proof of stake is another consensus algorithm proposed for cryptocurrency in 2011 on Bitcoin forum. It is a hybrid design [20] used to provide network security more than PoW. In this protocol, a stake is required to validate the block, which could be lost on doing a wrong validation. So, 51% attack in this protocol is nearly impossible because it is very hard to get 51% coin out of total value, and there is a threat factor also on doing a wrong validation in terms of stake loss. PoS solves the scalability issue and creates a block faster than PoW. Both PoW and PoS are basic consensus algorithms but they are complete and more decentralized, every new consensus algorithm has to be compared with these in terms of performance.

2.3. Performance Comparison

As blockchain is making progress with every day, new protocols are coming into existence. The scalability issue is also addressed in almost every concurrency. Many new consensus protocols are used in cryptocurrencies. Performance of these consensus algorithms is compared [21] in a delegated way by combining different studies discussed in Table 1. Complete comparison is available in the study, where some protocols are considered as a reference.

Table 1. Performance comparison for different consensus algorithms [21].

Consensus	Decentralization	Accessibility	Scalability	Comp. O.H.	Storage O.H.	Network O.H.	Latency	Throughput	Suitability (IoT)
PoW	High	Public, PL.	High	High	High	Low	High	Low	No
dPBFT	Medium	Private, P.	High	Low	High	High	Medium	High	Partially
Stellar	High	Public, PL.	High	Low	High	Medium	Medium	High	Partially
PoI	High	Public, PL.	High	Low	High	Low	Medium	High	Partially
Ripple	High	Public, PL.	High	Low	High	Medium	Medium	High	Partially
Raft	Medium	Private, P.	High	Low	High	N/A	Low	High	Partially
OmniLedger	High	Public, PL.	High	Medium	Low	Medium	Medium	High	Partially
RapidChain	High	Public, PL.	High	Medium	Low	Low	Medium	High	Partially
DPoS	Medium	Public, PL.	High	Medium	High	N/A	Medium	High	Partially
PoS	High	Private, P or PL	High	Medium	High	Low	Medium	Low	Partially
PoET	Medium	Private, P or PL	High	Low	High	Low	Low	High	Full
PBFT	Medium	Private, P.	Low	Low	High	High	Low	High	Full
Tangle	Medium	Public, PL.	High	Low	Low	Low	Low	High	Full

A cryptocurrency IoTA introduced a very new technology for a distributed ledger called Tangle. It is very highly scalable, with low computing and storage overhead. Moreover, it is very lightweight and specially designed for IoT devices with no transaction fee. Tangle uses DAG (Directed Acyclic Graph) in which each transaction is linked to the previous two transactions approved by it to add on the ledger through PoW. Tangle uses DAG for validation, so that is why it is not considered in the blockchain list and promises to overcome the existing barrier of decentralization for IoT resource-constrained devices. There is no mechanism in Tangle for the selection of the older two nodes to validate in IoT scenario; however, in IoTA cryptocurrency, Tangle runs an algorithm called tip selection for transactions [22]. Another problem is highlighted in [19], if a hacker acquires 33% (one third) hashing power of the total, then it can make it vulnerable and insecure.

As the most trusted platform for computing SGX, Intel proposed a new consensus mechanism proof of elapsed time (PoET) that is primarily based on Byzantine Fault Tolerance (BFT), which focuses on reducing the energy requirement. This protocol is lightweight and perfectly suitable for IoT in public and private ledger domains [23]. The suitability of this protocol can also be checked from the performance comparison table. PBFT (Practical Byzantine Fault Tolerance) is a Byzantine Fault Tolerance protocol, which is highly practical with low algorithm complexity in distributed systems [24] and contains five phases (1) request, (2) pre-prepare, (3) prepare, (4) commit and (5) reply. In PBFT, a client sends a message to the primary node which forwards it to other nodes to reach consensus. The message goes through five phases to complete the round of consensus among nodes and, finally, these nodes reply to the client. Nodes maintain common state to take consistent action in PBFT in each consensus round. PBFT creates and validates the block in DPoS to reduce the time in the consensus round [25]. The Steller consensus protocol (SCP) [26] is an improvement of PBFT which uses the federated byzantine agreement (FBA) protocol to conduct the consensus.

2.4. Continuous Growth in Crypto Market

The Crypto market is developing very fast due to establishing new studies based on new issues in the digital money market. To make digital money more reliable and fast, new lightweight consensuses algorithms emerge to mitigate new threats. A new cryptocurrency called Pi is introduced, which is in the mining phase and about to launch. It uses a very lightweight consensus protocol called Stellar consensus protocol (SCP) [26]. That is why Pi cryptocurrency can be mined easily on cell phones and does not drain the mobile battery, according to their claim. Stellar consensus protocol introduced a new consensus model called federated byzantine agreement (FBA). In a stellar algorithm, every node makes quorum slices by combining nodes on which it trusts. All these quorum slices join together to make a quorum. According to FBA, there should be 67% votes to make a transaction successful, and all the quorums must be joined together by a node to achieve consensus by federated voting. SCP works on two protocols: nomination protocol. All the nodes select transactions for the ledger by a voting process. After that, these nodes prepare and commit transactions by ballot protocol. It is a very scalable protocol for IoT in a public blockchain.

The decentralized control of proof of work and proof of stake is concluded in the previous table can also be seen in Table 2. Still, flexible trust in which users have the right to trust any appropriate combination of a party according to them, was a problem in PoW and PoS protocols. Low latency is a problem in PoW, which is solved in the Ethereum proof of stake (Casper). Digital signatures and hash families parameters are tested with large computing power which is unimaginable to protect against adversaries in Asymptotic Security. Earlier protocols such as PoW and PoS were not asymptotically secure but this problem is solved in the SCP consensus method. Another milestone achieved by SCP is decentralization which was not present in the byzantine agreement. It can be applied to IoT blockchain depending on the blockchain scenarios. If data is sent by IoT devices, after a short interval of time, then there will be a significant difference in block sizes. Another problem with SCP is that its security is highly dependent on the structure of

quorum slices and [27] also proved that PBFT is better than FBA in terms of liveness and security. So, SCP could not be the better option for blockchain IoT scenarios, but in terms of decentralization, it is exceptional with the voting mechanism mentioned in its nomination protocol and ballot protocol. The libra blockchain by Facebook [28] is a decentralized and efficient cryptocurrency for billions of people to exchange. This cryptocurrency is in the development phase and will be launched soon. Libra has developed an open-source implementation prototype to validate the design and needs global efforts for advancement its new ecosystem. To discuss Libra blockchain, the challenges are the same as IoT, such as high validation rate, low CPU utilization power of computational power, and it requires a huge storage due to a large number of accounts.

Table 2. Properties of different consensus mechanisms [26].

Mechanism	Decentralized Control	Flexible Trust	Low Latency	Asymptotic Security
Proof of Work	Yes	No	No	No
Proof of Stake	Yes	No	Maybe	Maybe
Byzantine agreement	No	Yes	Yes	Yes
Tendermint	Yes	No	Yes	Yes
SCP	Yes	Yes	Yes	Yes

The raft consensus algorithm is a very suitable consensus method for IoT in a private blockchain scenario. However, it will cause difficulty in public blockchain because if attackers become successful in compromising one node and change its election time out, there will be greater chances for a malicious node to become the leader. Transactions can be manipulated after that. Security is not a prime concern in this study, but in IoT, we have to become conscious due to vulnerabilities present in IoT networks.

The blockchain technology breakthrough in the cryptocurrencies world is unimaginable. This progression starts from the first generation cryptocurrency Bitcoin and moves towards second and third generation coins (Ethereum and Cardano) by gradually resolving complications. The management of Cardano (3rd Generation) coin [29] has been succeeded in overcoming the previous digital money problems such as scalability and TPS by making the PoS protocol more perceptive. They are struggling towards new issues in digital assets. PoW and PoS can be combined together as in [30]. A novel, two-hope blockchain is proposed in it. Analysis of this study shows that blockchain is secure as long as the majority of the resources are controlled by honest players even in the case of more than 50% computing power and controlling high stakes in that system. A very comprehensive survey on consensus algorithms was presented in [31] in which consensus algorithms were classified into two groups: voting and proof-based consensus algorithms. This study proposed a performance comparison between these two categories. In voting-based consensus algorithms, a researcher considered the roles of a node as in the Raft consensus algorithm [32]. It is very famous for which node can be in leader, candidate, and follower state. A follower can discover a new candidate team and the leader. In the voting process, each node has to go to the candidate state to cast a vote. Before going into the candidate state, each node has a different election time between 150 to 300 ms. After election time out, the follower becomes a candidate. Now, the candidate node has the authority to cast a vote for itself, ask for votes from other nodes (followers), and resets their election time. Each node casts a vote on the candidate and resets its election time out. The candidate becomes a leader if it gets a majority of the votes. The leader received transactions requests from many clients and saved them into his log entry list. The leader sends his logged transactions (r) and last transaction index (pi) to followers to make secure transaction orders for all verifying nodes. After verifying that transactions are the same in the list of all the nodes, the leader will choose an index (pi) and commit all transactions before that index. If the leader does not reply to the messages between election time, the new election starts, and another candidate requests a vote to other nodes. The whole process is fine and suitable in private blockchain. Voting-based algorithms are more secure than proof-based

algorithms but decentralization becomes low in voting-based algorithms except SCP, in which a voting mechanism is also introduced as discussed earlier. Executing nodes in voting-based algorithms are much less than proof-based algorithms which also causes to increase their performance and scalability in terms of transactions per second (TPS). SCP introduces a flexible trust as discussed earlier but, in Raft, trust becomes much less as compared to proof-based algorithms.

In Table 3, voting-based algorithms are compared with proof-based consensus algorithms. Voting-based consensus minimize decentralization, but security threat becomes low. Performance, scalability, and security issues are the basic problems in IoT blockchain due to maximum data handling and vulnerable IoT devices because of the small memory size and absence of standards in IoT. Blockchain protocols have become mature due to repeated use in digital currency. So, the selection of a consensus algorithm becomes easy, and we can also design our desired consensus algorithm according to our IoT scenario requirements by consulting cryptocurrencies protocols and problems related to IoT. The problem presented in this study can be solved by using a suitable consensus algorithm for BCIoT or by making appropriate changes in the existing algorithms.

Table 3. Difference between proof-based and vote-based consensus algorithms [31].

Criterion	Proof-Based Consensus	Vote-Based Consensus
Join nodes freely	Mostly	No
Decentralization	High	Low
executing nodes	Unlimited	Limited
Trust	More Trustful	Less Trustful
Award	More Serious	Less Serious
Security threat	Yes	Mostly No
Examples	PoW, PoS	Raft

In this section, consensus algorithms are discussed, which are used in a public and private blockchain. We have seen that transactions per second (TPS) were also a problem in early cryptocurrencies and improved later by different consensus algorithms. In IoT, there is a need of an efficient and scalable consensus protocol. In this study, a case study is taken in the next section, which require an efficient and scalable consensus algorithm for IoT devices by tolerating delay in communication.

3. Blockchain Overview

Blockchain is a decentralized and distributed public ledger used to record immutable transactions across a peer-to-peer network, i.e., without the need of a third party to monitor mediate transactions, and cryptographic mechanisms are used to secure transactions or blocks. This section presents an overview of the blockchain technology, including the key terms and its use-cases.

3.1. Blockchain Key Terms

To understand blockchain, it is necessary to be aware of its key terms. There are many terms used in blockchain and needed to be understand, but some important terms are discussed here. These terms are used in this study frequently, and it will not be easy to move further without understanding them.

1. Blocks: Blocks contain transaction data, which include sender address, receivers address, transaction amount, transaction fee, last block reference number (hash), and time stamp. There are multiple transactions stored in a block that a validator or miner must verify. It nearly takes ten minutes for Bitcoin to produce the new block.
2. Nodes: In a blockchain, each computer of a peer-to-peer network is called a node. These nodes float the transactions over a network for verification and store them in

their public ledger. The main objective of the nodes is to send the transaction from sender to receiver in a secure way.

3. Distributed Public Ledger: It records all the transactions so that no one can change it. If someone tries to make a change via any node, then it will not be accepted during consensus between nodes.
4. Consensus: It is a general agreement between all nodes in the blockchain. When a block is produced, all the nodes must be agreed if any of the node is changed or manipulated, the other nodes will not accept this change. The consensus mechanism creates a trust between all the nodes and a step towards blockchain immutability.
5. Flooding: Transactions reach every node by a process called flooding. When a node receives data, it sends them to all other nodes because mining will be started when a node receives all the data on the network.
6. Miner/Validator: A validator or miner is a computer that validates the transactions by calculating the hash, stores it in its ledger, and broadcasts it to the network. In bitcoin, many computers try to validate the block and the computer which solves it first gets the transaction fees and block reward.
7. Nonce: It is very easy to calculate the simple SHA-256 hash for certain data, but it becomes much more complex when the computer produces a block hash that meets certain requirements. In simple words, a nonce is a number to achieve a certain difficulty level added to the block for which the validator takes too much time and calculates several hashes until the desired result (hash) has been obtained.

3.2. Types of Blockchain

There are two main types of blockchain i.e. Public and Private, and a combination of public and private blockchain is also used which is called Consortium or Hybrid blockchain.

1. Public blockchain: A blockchain which is openly available to miners, developers and members of its community. All the transactions in public blockchain are transparent and accessible to everyone. Public blockchain is fully decentralized where no individual or entity is controlling it. Most of the cryptocurrencies blockchains are public such as Bitcoin and Etereum.
2. Private blockchain: Blockchain where only allowed persons can join the network and transactions are also available to its blockchain participants. Private blockchain is more centralized than the public one. Most of the enterprises do no want to disclose their sensitive data between groups of customers or want to hide their offers to specific customers. Ripple and Hyper-ledger are using it.

3.3. Blockchain Use Cases

Blockchain is used in almost every industry today. In this section, some interesting use cases will be discussed which are gaining more attention and causing an increase in blockchain importance. As the technology is growing every day, the adoption of blockchain is also climbing up exponentially.

Internet of Vehicles (IoV) is a fascinating topic and related to IoT in real-time response. In [33], it was mentioned that blockchain is effective for IoV due to its decentralized and distributed storage, and an outward transmission model was proposed by numerical and theoretical analysis. Security is essential in mobile-based IoV because there are large number of malicious attackers, so an authentication mechanism that is decentralized is proposed [34] based on the blockchain consensus algorithm. Car parking is another challenge for resident people and drivers in a highly densely populated area. A blockchain-based car parking system [35] which shares resources of paid parking between user and owner. Need for the third party is removed in this study, and resources are used intelligently. Supply chain management can be used in private blockchain [36] with the protocol of ultra-lightweight RFID. In this scenario, nodes of the supply chain are divided into four categories (distributor, retailer,end-use and manufacturer) which need a different level of access. In this study, the proposed protocol also provides security by reducing attacks,

such as a man in the middle, reply, key disclosure, and tracking. In industry, a credit-based PoW mechanism was proposed [37] which minimizes the consumption of power for nodes and increases security for malicious nodes. A novel reporting system based on blockchain ReportCoin was proposed in [38], which is used in the smart city for the management. Security of transactions and user identity is increased due to the decentralized nature of blockchain. This study creates trust between sender and receiver without disclosing the sender's identity. PETchain [39] is based on blockchain that uses it to enhance privacy. Different hospitals lag control the electronic health records (EHRs) during the information sharing process. This problem is controlled by cloud-based EHRs in which sharing of information becomes easy, but the centralization problem of the cloud emerges. In [40], the system model of blockchain-based EHRs is highlighted to overcome centralized issues. An identity-based signature scheme is used to reduce collision attacks. An electronic health wallet (EHW) system [41] is proposed that uses decentralized technologies such as blockchain to ensure data privacy and interoperability. Due to developments in blockchain and IoT, a transaction model for accounting was proposed in [42] which is capable of collecting, uploading, and recording the data automatically. IoT (Internet of things) is globally adopted in many areas, such as medical, houses, industries, etc. The combination of IoT and blockchain is also a mature topic, and there are many studies on LightChain resource-efficient blockchain for industrial IoT [43]. Access management in IoT using blockchain [1] and authentication scheme in IoT using blockchain [44] is presented. A design for IoT blockchain [45] by using PoS protocol for Bazo cryptocurrency [46] in the presence of Lora nodes and gateways is presented, which mean blockchain can be designed for IoT using consensus protocols.

4. Methodology

A consensus algorithm is designed by considering the problem statement and dataset of IoT wireless sensor network. Changes are done in the consensus by consulting cryptocurrency algorithms and explained in this section by using suitable examples. CBCIoT is ready to test according to the scenario used at the end of Section 4. BCIoT scenario is different from cryptocurrencies' transactions. As we talked about, any consensus protocol for this kind of blockchain cannot be implemented easily. Therefore, changes in consensus protocol are required to make it intelligent for BCIoT.

- In cryptocurrencies, we deal with transactions, but a massive amount of data in IoT blockchain needs to be handled. So, the broadcast domain should be limited, and we could survive from waiting for a validation process.
- If data are distributed to all nodes in the network, they will cause the network to slow down. They should be distributed to selected nodes or groups of nodes.
- In IoT, data are used instead of coins, so a proper validator/miner selection technique should be used. A validator is selected in PoS by selection techniques in cryptocurrencies, i.e., coin age-based selection (validator is selected by multiplication age of the coins in days with several coins that are being staked) and Randomized block selection (the next validator is selected by combining the lowest hash value and the size of their stakes).
- In cryptocurrencies, the miners get the reward and punishment for doing wrong, if they are selected by the PoS protocol. So here, we cannot punish, but we can lower machine ratings. We can also apply the error detection and correction method to the validator.
- We cannot rely on a single node to validate the data in our scenario. So, there should be multiple nodes for this purpose.
- There should be randomness in the consensus protocol to make it difficult for the attacker. Although security is not the prime topic in this study, we need to keep some basics in our minds because of vulnerabilities present in IoT.

4.1. Problem Solving Strategy by Consensus

The validation rate can be increased by choosing a suitable consensus algorithm, as previously discussed. So here, we will try to develop a new consensus algorithm for BCIoT. Nodes are taken in this algorithm as follows (total number of nodes = N)

$$N = mv + 1 \tag{1}$$

v (where v = 3, 5, 7, 9....) is the number of validating nodes, m (where m = 2, 3, 4.5.....) is a multiple validating nodes. Values of m and v are to be chosen carefully. On choosing small values, randomness could be compromised, which leads to compromise the security. In the case of choosing large values, the validation rate will slow down and affect the algorithm's performance. The selection of nodes should be

$$Selectionofvalidatingnodes = (N-1)/m \tag{2}$$

An answer should be a whole number or $N-1$ should be divisible by m. In this study, we mostly use m = 3 and v = 5. For example, if N = 16, $Master$ = 1, $N-1 = mv$ = 3(5) = 15.

4.1.1. Master Node

The master node will be selected by voting, which can select five nodes for validation and verification.

4.1.2. Voting

Every node can send a token (number with time stamp) called vote to only one node randomly at a time, which helps to select the master node. Voting process is also explained in [32] but, in this study, a simple voting scheme is used to select the master for one turn. After sending the token, a node with more votes(V)*ratings(R) than others must be a master node.

$$Masternodeselection = VXR \tag{3}$$

4.1.3. Rating

It is a fractional number (R) between -10 to 10, which is increased by first validating the block (the same like the winner gets the reward in PoW) and decreased by doing wrong validation and verification (such as loss of stakes in PoS). In this scenario, validation reward and punishment are in terms of rating.

4.2. Procedure

Consensus in blockchain IoT (CBCIoT) is designed according to the scenario discussed in the dataset. Voting mechanisms are discussed in the previous section of related work. Some changes are needed to be made in the voting process. The procedure of working for CBCIoT algorithm is as follows.

- Every node will send data to the master node, which is selected by all the blockchain nodes in the voting process.
- Master node will receive data for 30 s (to collect maximum one time data) explained in the dataset section; after this period, a master node will send data to 5 randomly selected nodes for validation.
- Five nodes will create a block, and only one node will first send the block to the other four nodes for verification (difficulty level can be added to check the difference).
- If a block is created by more than one node at the same time, the block with greater number of verifications will win.
- After validation and verification, the block and information are broadcasted by the master to all blockchain nodes, which will be stored in their ledger.
- After a block is created, the same procedure is repeated for the next block.

- Genesis block will be created by the blockchain standard procedure.

4.3. Points of Concern

The working procedure is explained. There are some limitations that can affect the performance of CBCIoT. So, it is needed to be carefully viewed in the below points before moving further.

- All the nodes should be time-wise synchronized.
- Master node selection procedure should be fast to collect data from other nodes.
- $N-1$ nodes have to wait a little bit to send data to master every time.
- Simulation should be flexible so that number of nodes, time, etc., can be changed to check the differences in results.

4.4. Explanation with Example

Node A is selected as a master from $N = 16$ nodes as shown in Figure 3, and it further selects five nodes for validation and sends data to 5 nodes to generate a block. All the nodes create the selection of master by a voting process. In this process, every single node can vote for one node at a time. Once a master is selected, all the blockchain nodes can send data to the master. Next, the master selects five nodes (but never selects more than one node with a rating less than 0) and sends data to them for validation. Validating nodes generate a block and, after verification, sends it back to master, which is broadcasted to all IoT blockchain nodes to store in their public ledger. Why does the master select odd (e.g., 5) nodes for validation?

- If it selects all the nodes, then validation will be slowed down as discussed in PoW.
- In case one node for validation could be a better option and blockchain performs efficiently, the attacker can insert data and validate them, if this node is compromised.
- In the case of 3 validating nodes, if one node is compromised, then the generated block will be accurate but less reliable due to one verification.
- If two nodes create two blocks simultaneously, then one block will win due to a greater number of verifications, and it is well performed in an odd number of nodes.

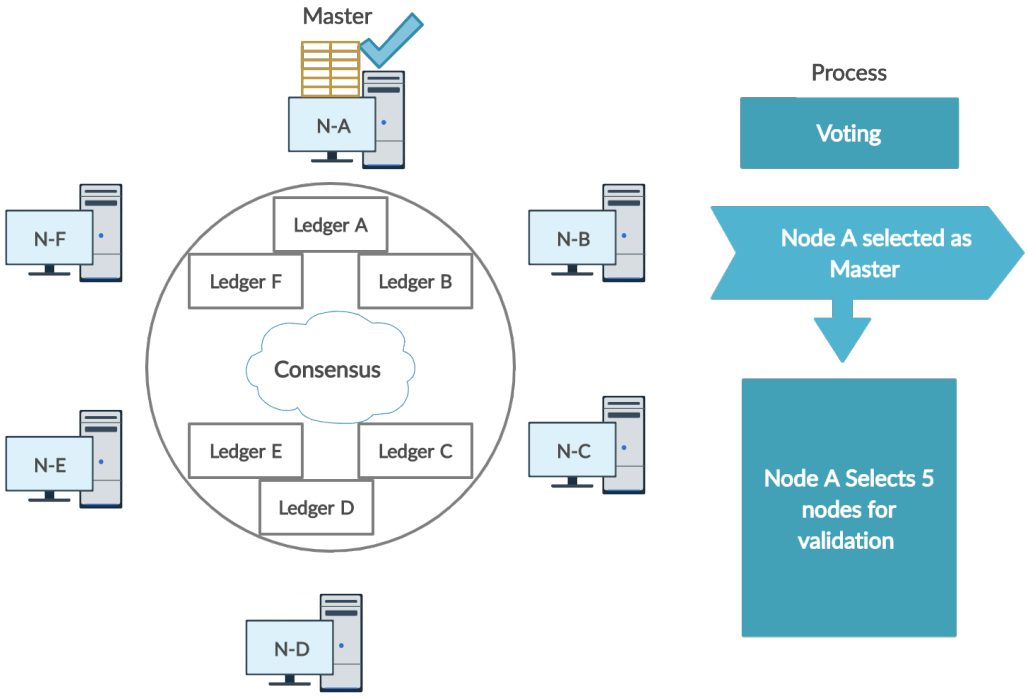

Figure 3. BCIoT working.

So, an odd number of validating nodes could be a better option, and reliability will be increased for a more significant number of verifications. In the case of validation, nodes greater than five will make the block more authentic, but the validation rate will be decreased. So, selection of validating nodes should be according to the required scenario.

4.5. Phases of Nodes in IoT Blockchain

In this proposed consensus algorithm, blockchain nodes move through different phases. The initial phase is the long phase of this consensus in which every node has no options except to collect data from IoT systems.

After the initial phase, all nodes go into a temporary phase called the election phase, in which every node has the right to cast only one vote for a random node for choosing the master node. When a master is chosen according to Equation (3), all $N-1$ nodes go back to the initial phase.

Now, the Master node has the right to select validating nodes (v) from $N-1$ nodes. Total validating nodes are shown in Equation (2). They are selected randomly and consult their ratings also with the master node. These validators validate and verify the block for IoT blockchain. When the block is broadcast to all blockchain nodes, all the nodes, validators, and masters return to the initial phase, and the process starts again. The whole process is shown in Figure 4.

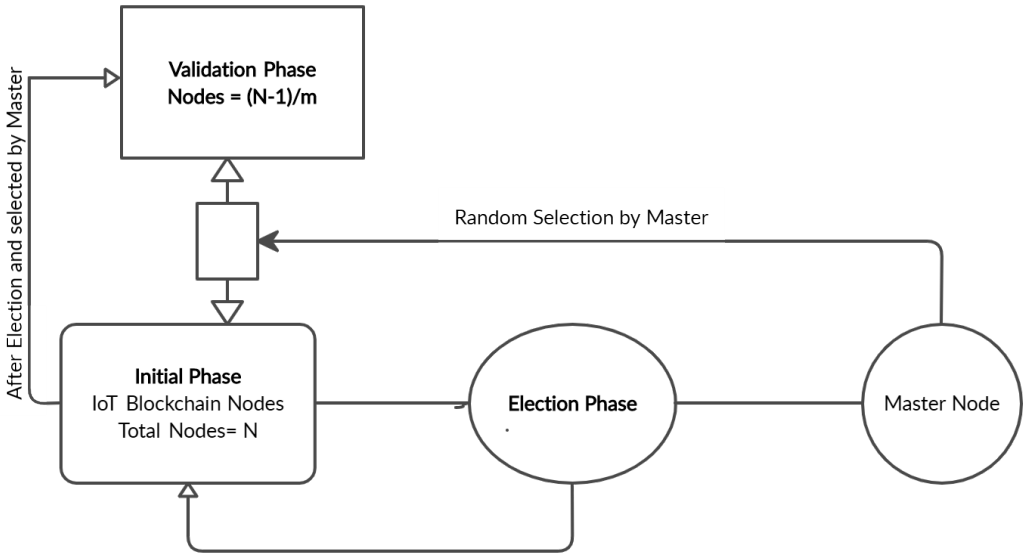

Figure 4. Different phases of BCIoT nodes.

4.6. Dataset

To verify our consensus algorithm, a dataset is chosen which is collected during deployment. This dataset is taken from the Remote Management System (RMS), also called Fuel Management System (FMS), deployed on Warid sites to control fuel theft. It is an IoT system packet with lot of information shown in Figure 5. RMS control panel makes decisions on-site (analysis and store in buffer) before sending information to the server after every 30 s. Each site sends 78 Bytes of data and the server was collecting more than 500 packets after regular intervals. An initial experiment was carried out on 1000 packets (samples) to record on the blockchain. An efficient system consists of two panels:

1. RMS Panel;
2. Fuel Sensor Box.

The RMS panel placed inside the site room to monitor phases of WAPDA (Water & Power Development Authority) and generator, room temperature, panel temperature, and voltages of the two battery banks. The code 1.1.1 means all three phases of WAPDA or generator are working. If any phase is missing, then it will show 0 instead of 1. A generator

is started by using these calculations to charge batteries or to switch on the air conditioner in the room as in VTDC (voltage and Temperature-Dependent Control).

The sensor box was the second most important part of this system which is located on the fuel tank outside the site room. It has three ultrasonic sensors to calculate the fuel in the tank. Three sensors are used for correctness in the reading. It also calculates the temperature of the fuel by the temperature sensor. Sensor boxes transmit these readings to the main RMS panel to calculate theft or usage of fuel on the site.

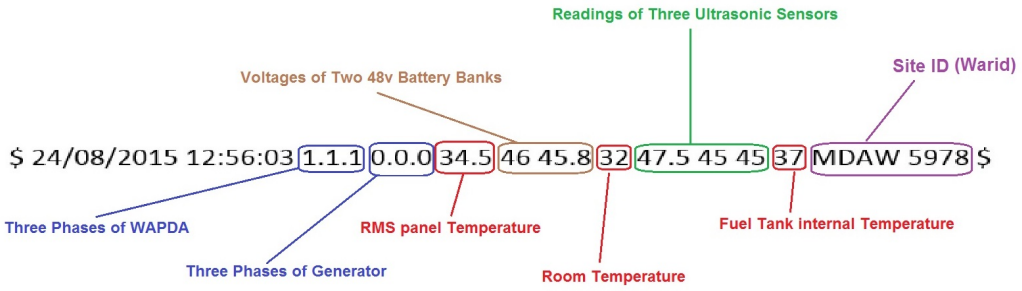

Figure 5. Remote management system.

4.7. Key Metrics of Performance

In this study, a public blockchain is used and a new consensus algorithm is proposed; so, it is needed to check its nature in terms of centralization and decentralization. Scalability of the consensus algorithm is another problem for IoT blockchain which is tested for different transactions. Throughput and verification speed of this algorithm is needed to be verified and compared with other algorithms.

5. Analysis and Results

The core section of this paper consists of a detailed analysis on the basis of its operation. Three major properties of CBCIoT algorithm are a part of this section. Worst case scenario is considered in the case study; after that, results are drawn to strengthen the blockchain IoT environment.

5.1. Sequence of Operations

In this study, the proposed algorithm's work is suitable for IoT blockchain, where a slight time delay (wait to collect data from all sites) is required. The working principle of this algorithm can be understood easily by this flow chart.

The flow chart shown in Figure 6 is used to understand the process of this consensus algorithm in the case of validating nodes (v = 5) and integral multiple (m = 3). Data receiving and voting processes are running in parallel. If the master node is selected, then data are transmitted to it, which will be further sent to validating nodes after selecting and removing duplicates from the data. Data collection by BCIoT nodes and master transmission to validating nodes is an about 30-s long process as we have seen this time delay in RMS. The master node also removes duplicate values by comparing data values. Hash for Genesis block in this consensus algorithm is created and stored in the ledger of all the blockchain nodes on the initial stages. The master node will broadcast the block to all BCIoT nodes and increase the rating of the first validator. The voting process is started to select the next master. This process runs in parallel with the previous validation process, but in this case, any prior validator or master could not be selected again. They have to wait for one turn more.

5.2. Unpredictability in the Algorithm

This consensus algorithm uses three integral multiples (m) for five validating nodes (v), which could be changed according to our scenario requirements. The selection of the master node is random by voting, and its probability is 0.0625 in case of a total of 16 nodes and goes on decreasing by increasing the total number of nodes. The master node selects

additional five nodes with a probability $p(v)$ of 0.3333 for validation, making it impossible for the attacker to compromise a node. For example, in the worst-case scenario, one or two validating nodes are compromised. Still, they cannot validate the block because the other three nodes will validate and verify the block due to the majority (discussed in the case study section of this chapter).

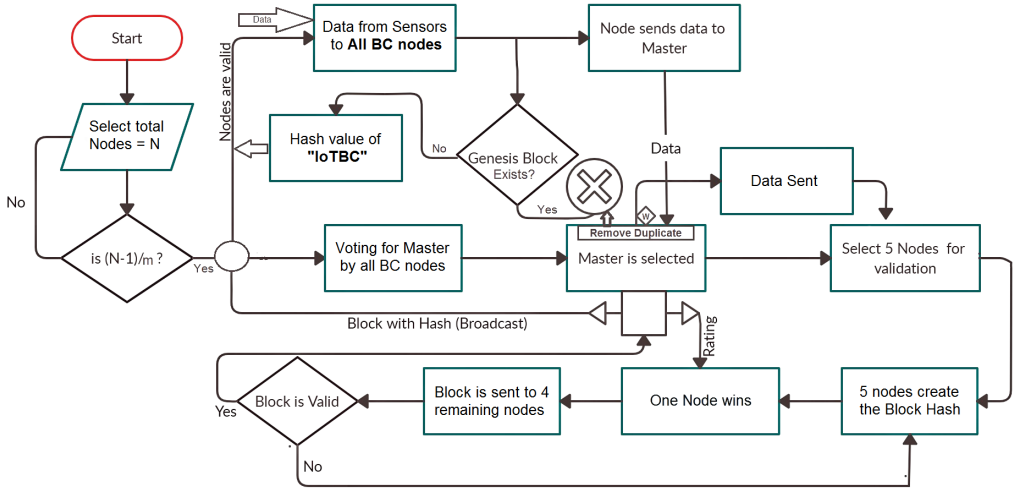

Figure 6. CBCIoT flow chart.

In this study, one node is selected as a master node, and validating nodes (v) and multiple (m) could be selected according to Equation (1) in the previous section. However, if we consider more than 5 validating nodes (v), blockchain working becomes less efficient, and computational power is increased, as we have seen in the Bitcoin scenario. Selecting v less than 5 for validating will lead the blockchain to compromise its security and fewer block verifications as security is not tested. Still, we have to assume this due to vulnerabilities present in IoT. Suppose one node produces a block and three nodes are selected for verification of this produced block. In that case, only three verifications are considered to be received in an ideal scenario. These block verifications can be decreased if one or more nodes are malicious or do not agree upon a newly generated block.

Randomness can be increased by increasing the integral multiple of validating nodes (m), increasing $N-1$ nodes by m times for a specific number of validating nodes (v). For example, in this study, we are using $m = 3$ and $v = 5$ which means $(N-1) = 15$, i.e., 3 times 5. By decreasing m, randomness could be decreased, and the probability of validating nodes selection by master $p(v)$ will increase and leads towards less security of the blockchain. In this case, master node selection by voting will also become more probable (e.g., $1/16$ for $m = 3$, $1/11$ for $m = 2$ and $1/6$ for $m = 1$). That is why the value of m is greater than 2. If we select $m = 2$, then validating the node's selection probability becomes exactly half. The value of m could not be one because the probability of validating nodes selection is equal to 1. It is much easier for the attacker to manipulate the transactions by compromising 60 percent BCIoT, validating nodes and guessing the master node. Compromising a master node could be much more harmful in our CBCIoT scenario.

The probability of validating nodes selection by master $p(v)$ decreased by increasing the number of integral multiple (m) and $v = 5$ (constant). In our scenario, for $m = 3$, probability is 0.33; this is also shown in Figure 7 and goes on decreasing for greater values of m.

Figure 8 shows the total approximate time for validation and verification per block, which goes on increasing v and $m = 3$ for 7000 samples. A time delay of 30 s is included in the graph as it can be seen that every block is validated more than 30 seconds. If the master takes more nodes for validation, then one node creates a block first, and the remaining nodes verify it and send the verified block back to the master for broadcasting in IoT blockchain. Every node on the blockchain will store this new block into its ledger. So, the

number of verifications is increased by increasing v, but the block producing time is also climbing up.

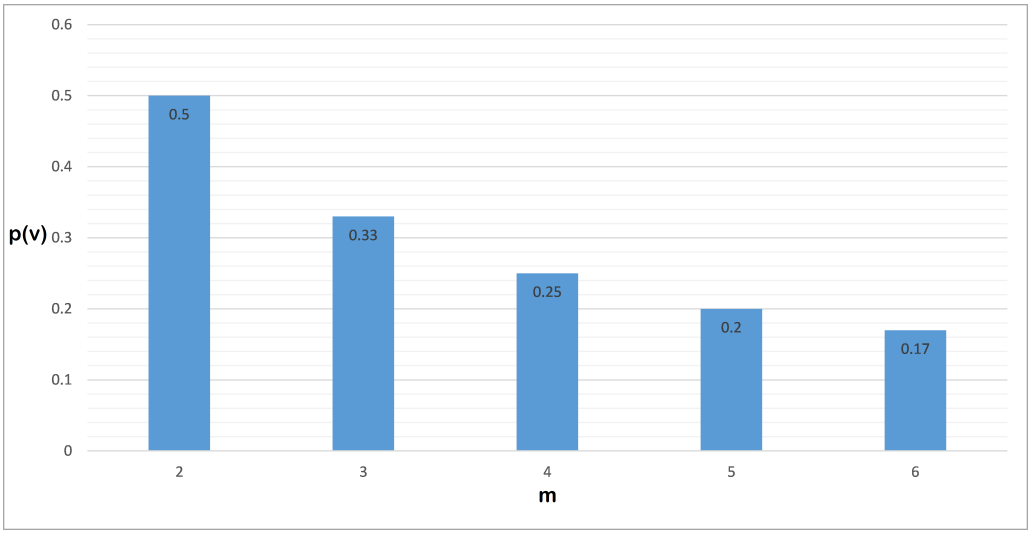

Figure 7. Probability of validating nodes.

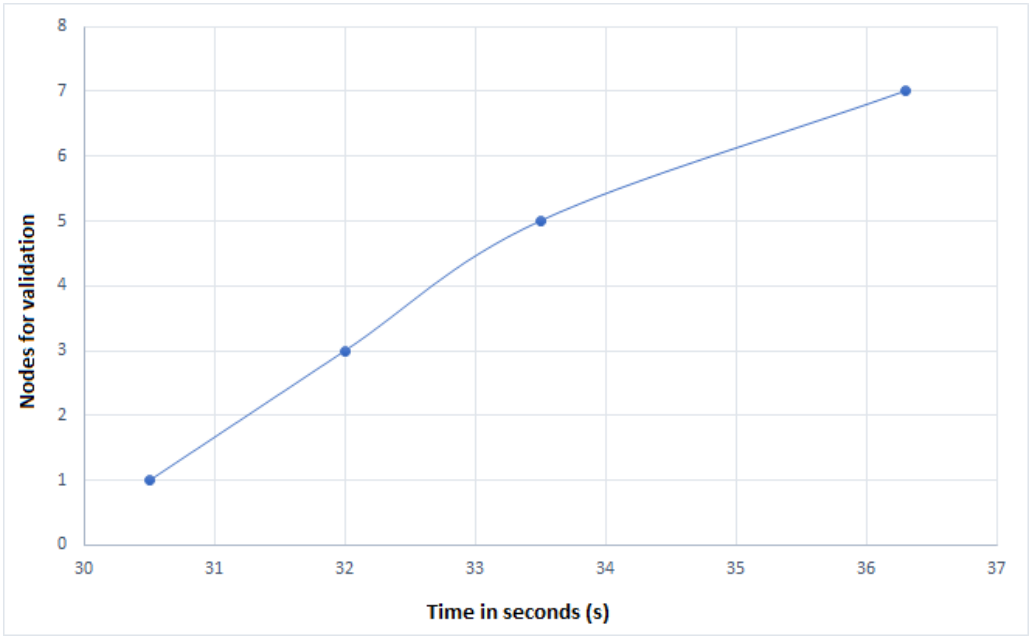

Figure 8. Total time for validation and verification per block.

5.3. Limited Broadcast Domain

It can be seen that data are transmitted to the master by all BCIoT nodes, so flooding is minimized. It could be a major threat to BCIoT, which will cause a decrease in the efficiency, validation, and verification rate. As we further see, this algorithm, validation, and verification performed by a limited number of validators will also cause an increase in the validation rate. Hypothetically, it is assumed that after validation and verification performed by all nodes, there will be greater chances of delay in the consensus.

5.4. Flexibility in the Algorithm

As it can be seen in previous sections, this algorithm is flexible and can be changed according to IoT scenario requirements. We have discussed that the integral multiplier (m) can be increased to maximize security in terms of randomness, and validating nodes (v) can also use different numbers according to the number of validations per the second

requirement. The value of N,m, and v can be selected according to Equation (1) in the previous section. We choose to select integral multiplier m to increase randomness or validate nodes v according to block generation time and verification speed.

5.5. Case Study

To check the efficiency of the proposed algorithm, we assume a case study in which the attacker can compromise two validating nodes out of five and try to manipulate or insert his data. Although, it is challenging for the attacker due to its randomness, but we are taking it as a worst-case scenario. This is clearly shown in Figure 9 where the attacker takes control of two validating nodes.

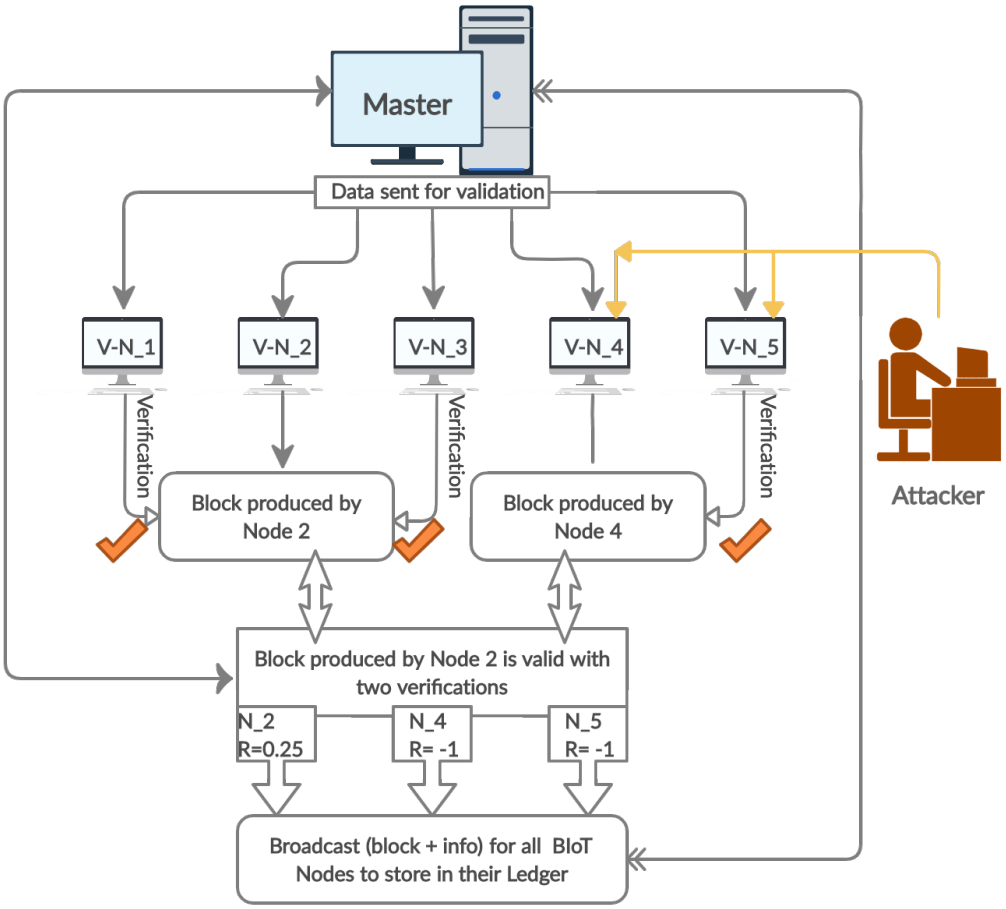

Figure 9. Case study for blockchain IoT.

Then, two blocks will be produced:

- Block produced by node 2 with two verifications;
- Block produced by malicious node 4 with one verification.

After the block is produced by node 2, it is valid due to the more significant number of verifications. So, the master increased its rating by 0.25 and decreased the ratings of malicious nodes 4 and 5 by one due to wrong validation and verification.

In the previous section, Equation (3) describes the criteria for master node selection. In our case study, the rating is decreased by -1. So, their chances of becoming masters are less than other nodes because the rating is multiplied by the number of votes, and the whole value becomes negative. These two malicious nodes can become masters only if they do four or more validations right by retaining first place in every validation. Their chances of becoming validating nodes are also affected because the master will not select more than one validating node with negative ratings. If one of them becomes a validating node again and does something wrong, its rating will be furthermore decreased in the same way.

5.6. Results

To verify our algorithm, a Java-based simulation tool is used in Ubuntu environment. This simulation tool can create a new blockchain, computer network, consensus algorithm, and changes in the existing blockchain to support PoW and PoS protocols. This simulation tool saves its output in

ROOT DIR/simulator/src/dist/output

The code 000 in hash is the difficulty level set in this blockchain 2.0. First hash

000eb84c1db7b85ffbda9315ef64ffd4c50da90cacd3a27e06c33f2ab3fc6da5

which is used in Genesis block as a previous hash, it is created by taking a hash of "IoTBC" in multiple attempts to get 000 on the leftmost position by adding a random number at the beginning of this consensus algorithm. RMS data of 1000 samples (1000 × 78 Bytes) are tested on initial level; then increased to check the performance of CBCIot. The calculated time to produce a block is different due to different amounts of data and the number of hashing performed to meet this blockchain difficulty level. The time to produce a block is about 35 s for 6–7 thousand samples, and after that, the block generation time is increased a little bit as shown in Figure 10 (slightly more than 35 s).

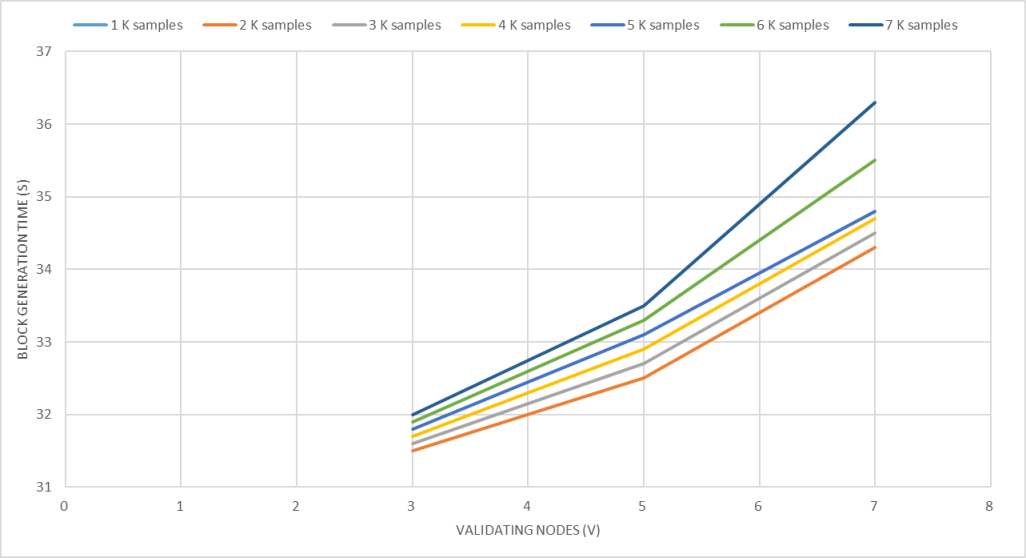

Figure 10. Block generation time for different samples.

This algorithm is tested for 5 validating nodes (v) and multiple (m) 3 on priority basis after that relation between m and v is verified according to Equation (1) by taking nodes mentioned in Equation (2). Master node selection is checked as in Equation (3) on the initial level and also verified by lower the rating (R). The performance of the proposed algorithm (CBCIoT) is fine after carefully reviewing the results, and limitations are also discussed in the next subsection.

Table 4 shows the comparisons of the proposed consensus algorithm with other consensus algorithms. Protocol data are chosen from [26,47] and compared with our proposed algorithm. CBCIoT is tested on blockchain 2.0 and it has some properties of PoW and PoS. A voting mechanism is also used for the selection of master, so it is important to compare with other voting algorithms such as SCP and RAFT. The voting process is introduced to select a master for one turn and, then, a master selects validating nodes to generate a block. In the next turn, the master and validating nodes are changed and this process is continuous for every new block. It shows that CBCIoT is decentralized in which every node has equal opportunity to become master and validator. Algorithm is tested for different samples (1000–7000) with different combinations of m and v, every time it

worked perfectly fine which shows its strong scalabilty. The verification speed of the block is better as compared to other algorithms. However, block generation time is nearly 35 s in the proposed algorithm. It is not comparable because of the 30 s delay to collect data in our scenario and less data size of a single RMS system. However, if delay is ignored, then its throughput lies in the range of (250 to 1500) TPS or samples per second for 1000 to 7000 samples, which clearly indicates that CBCIoT has a high throughput. Block is verified less than 5 s for every experiment made with up to 7000 samples, and it is same like SCP and RAFT.

Table 4. Comparisons between consensus algorithms [26,47].

Characteristics	PoW	PoS	DPoS	RAFT	SCP	CBCIoT (Study)
Accessibility	Public	Public, Private	Public	Private	Public	Public
Decentralization	High	high	High	Medium	High	High
Throughput	Low	Low	High	High	High	High
Scalibility	Strong	Strong	Strong	Weak	Strong	Strong
verification Speed	>100 s	<100 s	<100 s	<5 s	<5 s	<5 s

5.7. Limitations

In this study, the proposed algorithm can be used in different scenarios; as discussed earlier, its performance is enough to fulfill the BCIoT requirement used in our described scenario (discussed in the dataset section of the previous section). However, it is simulated and has some limitations:

- Its working is fine for block size <500 KB, and performance could be degraded a little bit for larger block size as discussed in the results section.
- Difficulty level directly affects it because the number of hashes per block is increased, and it is tested on blockchain 2.0.
- Selection of master node cannot be performed twice consecutively for a single node if its block is not broadcasted.
- Validator node cannot become a Master in the next round if it is a validator in the previous block and master does not broadcast this block in the IoT blockchain.
- This algorithm is working in public blockchain and not tested for private blockchain.
- Security is kept in mind during its construction due to vulnerable IoT systems, but it has not been not tested or tried to compromise it.
- It is suitable and working fine for IoT scenarios where a little delay is present or can be tolerated.

6. Conclusions and Future Work

Blockchain has great potential in the IoT field, but issues such as scalability, security and privacy, power and processing time and storage capacity are complex and mature due to the combination of these two giant technologies. The scalability problem could be solved by designing a proper consensus algorithm. In this study, a consensus algorithm "CBCIOT" is designed for blockchain-based IoT applications, working perfectly for IoT devices that are not delay-sensitive. Limited broadcast domain causes to increase its efficiency. It can be configured according to IoT requirements due to the flexibility present in the algorithm. In this algorithm, a delay (wait to collect maximum one time data from IoT devices) is required, so it will be a great choice to use it for other IoT scenarios where delay can be tolerated. This study is only based on a consensus algorithm, so parameters that affect its scalability and throughput are addressed adequately. Results show its reliability and make it proficient in TPS and verifications (<5 s) such as present-day voting-based algorithms in cryptocurrencies. The anonymity of master node selection and validating nodes by a master make it protective against attacks. Although it has not been tried to compromise this algorithm, this could be a topic for future work.

Author Contributions: Conceptualization, M.U., M.M. and M.K.H.; methodology, M.U. and M.M.; validation, M.U.; writing—original draft preparation, I.T.J. and M.K.H.; writing—review and editing, B.A. and N.C.; visualization, I.T.J.; funding acquisition, N.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work is partly supported with the financial support of the Science Foundation Ireland grant 13/RC/2094_P2 and partly funded from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska–Curie grant agreement No 754489.

Conflicts of Interest: The authors declare no conflict of interest

References

1. Novo, O. Blockchain meets IoT: An architecture for scalable access management in IoT. *IEEE Internet Things J.* **2018**, *5*, 1184–1195. [CrossRef]
2. Panarello, A.; Tapas, N.; Merlino, G.; Longo, F.; Puliafito, A. Blockchain and iot integration: A systematic survey. *Sensors* **2018**, *18*, 2575. [CrossRef]
3. Abou Jaoude, J.; Saade, R.G. Blockchain applications–usage in different domains. *IEEE Access* **2019**, *7*, 45360–45381. [CrossRef]
4. Atlam, H.F.; Wills, G.B. Technical aspects of blockchain and IoT. In *Advances in Computers*; Elsevier: Amsterdam, The Netherlands, 2019; Volume 115, pp. 1–39.
5. Padma, M.; KasiViswanath, N.; Swathi, T. Blockchain for iot application: challenges and issues. *Int. J. Recent Technol. Eng.* **2019**, *7*, 34–37.
6. Banafa, A. 7 IoT and Blockchain: Challenges and Risks. In *Blockchain Technology and Applications*; River Publishers: Gistrup, Denmark, 2020.
7. Qureshi, K.N.; Sandila, M.A.S.; Javed, I.T.; Margaria, T.; Aslam, L. Authentication scheme for Unmanned Aerial Vehicles based Internet of Vehicles networks. *Egypt. Inform. J.* **2021**. [CrossRef]
8. Zheng, Z.; Xie, S.; Dai, H.; Chen, X.; Wang, H. An overview of blockchain technology: Architecture, consensus, and future trends. In Proceedings of the 2017 IEEE International Congress on Big Data (BigData Congress), Honolulu, HI, USA, 25–30 June 2017; pp. 557–564.
9. Dorri, A.; Kanhere, S.S.; Jurdak, R. Towards an optimized blockchain for IoT. In Proceedings of the 2017 IEEE/ACM Second International Conference on Internet-of-Things Design and Implementation (IoTDI), Pittsburgh, PA, USA, 18–21 April 2017; pp. 173–178.
10. Mohanta, B.K.; Satapathy, U.; Panda, S.S.; Jena, D. A novel approach to solve security and privacy issues for iot applications using blockchain. In Proceedings of the 2019 International Conference on Information Technology (ICIT), Bhubaneswar, India, 19–21 December 2019; pp. 394–399.
11. Javed, I.T.; Alharbi, F.; Bellaj, B.; Margaria, T.; Crespi, N.; Qureshi, K.N. Health-ID: A Blockchain-Based Decentralized Identity Management for Remote Healthcare. *Healthcare* **2021**, *9*, 712. [CrossRef] [PubMed]
12. Alamri, B.; Javed, I.T.; Margaria, T. Preserving patients' privacy in medical IoT using blockchain. In Proceedings of the International Conference on Edge Computing, Honolulu, HI, USA, 22–26 June 2020; pp. 103–110.
13. Wang, G.; Shi, Z.; Nixon, M.; Han, S. Chainsplitter: Towards blockchain-based industrial iot architecture for supporting hierarchical storage. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 166–175.
14. Biswas, S.; Sharif, K.; Li, F.; Maharjan, S.; Mohanty, S.P.; Wang, Y. PoBT: A lightweight consensus algorithm for scalable IoT business blockchain. *IEEE Internet Things J.* **2019**, *7*, 2343–2355. [CrossRef]
15. Fernández-Caramés, T.M.; Fraga-Lamas, P. A Review on the Use of Blockchain for the Internet of Things. *IEEE Access* **2018**, *6*, 32979–33001. [CrossRef]
16. Ferrag, M.A.; Derdour, M.; Mukherjee, M.; Derhab, A.; Maglaras, L.; Janicke, H. Blockchain technologies for the internet of things: Research issues and challenges. *IEEE Internet Things J.* **2018**, *6*, 2188–2204. [CrossRef]
17. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. Technical Report. 2008. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 10 January 2021).
18. Back, A. Hashcash-a Denial of Service Counter-Measure. 2002. Available online: http://www.hashcash.org/hashcash.pdf (accessed on 16 January 2021).
19. Lo, S.K.; Liu, Y.; Chia, S.Y.; Xu, X.; Lu, Q.; Zhu, L.; Ning, H. Analysis of blockchain solutions for IoT: A systematic literature review. *IEEE Access* **2019**, *7*, 58822–58835. [CrossRef]
20. King, S.; Nadal, S. Ppcoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake. Self-Published Paper. 19 August 2012. https://www.chainwhy.info/upload/default/20180619/126a057fef926dc286accb372da46955.pdf (accessed on 16 January 2021).
21. Salimitari, M.; Chatterjee, M. A survey on consensus protocols in blockchain for iot networks. *arXiv* **2018**, arXiv:1809.05613.
22. Popov, S. The Tangle. [Online] 2016. http://www.descryptions.com/Iota.pdf (accessed on 22 February 2021).
23. Corso, A. Performance Analysis of Proof-of-Elapsed-Time (PoET) Consensus in the Sawtooth Blockchain Framework. Ph.D. Thesis, University of Oregon, Eugene, OR, USA, 2019.

24. Castro, M.; Liskov, B. Practical byzantine fault tolerance. In *OSDI*; The USENIX Association: Berkeley, CA, USA, 1999; Volume 99, pp. 173–186.
25. Swathi, B.; Meghana, M.; Lokamathe, P. An Analysis on Blockchain Consensus Protocols for Fault Tolerance. In Proceedings of the 2021 2nd International Conference for Emerging Technology (INCET), Belagavi, India, 21–23 May 2021; pp. 1–4.
26. Mazieres, D. *The Stellar Consensus Protocol: A Federated Model for Internet-Level Consensus*; Stellar Development Foundation: San Francisco, CA, USA, 2015; Volume 32.
27. Kim, M.; Kwon, Y.; Kim, Y. Is Stellar as secure as you think? In Proceedings of the 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Stockholm, Sweden, 17–19 June 2019; pp. 377–385.
28. Amsden, Z.; Arora, R.; Bano, S.; Baudet, M.; Blackshear, S.; Bothra, A.; Cabrera, G.; Catalini, C.; Chalkias, K.; Cheng, E.; et al. The Libra Blockchain. Calibra Corp. 2019. https://mitsloan.mit.edu/shared/ods/documents?PublicationDocumentID=5859 (accessed on 14 April 2021).
29. Guides, T.S. Why Cardano ADA Deserves Your Attention–Cardano Cryptocurrency Strategy. 2018. https://tradingstrategyguides.com/cardano-cryptocurrency-strategy/ (accessed on 10 January 2021).
30. Duong, T.; Fan, L.; Katz, J.; Thai, P.; Zhou, H.S. 2-hop blockchain: Combining proof-of-work and proof-of-stake securely. In Proceedings of the European Symposium on Research in Computer Security, Darmstadt, Germany, 4–8 October 2020; pp. 697–712.
31. Nguyen, G.T.; Kim, K. A Survey about Consensus Algorithms Used in Blockchain. *J. Inf. Process. Syst.* **2018**, *14*, 101–128.
32. Ongaro, D.; Ousterhout, J. In search of an understandable consensus algorithm. In Proceedings of the 2014 USENIX Annual Technical Conference (USENIX ATC 14), Philadelphia, PA, USA, 19–20 June 2014; pp. 305–319.
33. Jiang, T.; Fang, H.; Wang, H. Blockchain-based internet of vehicles: Distributed network architecture and performance analysis. *IEEE Internet Things J.* **2018**, *6*, 4640–4649. [CrossRef]
34. Wang, X.; Zeng, P.; Patterson, N.; Jiang, F.; Doss, R. An improved authentication scheme for internet of vehicles based on blockchain technology. *IEEE Access* **2019**, *7*, 45061–45072. [CrossRef]
35. Hu, J.; He, D.; Zhao, Q.; Choo, K.K.R. Parking management: A blockchain-based privacy-preserving system. *IEEE Consum. Electron. Mag.* **2019**, *8*, 45–49. [CrossRef]
36. Sidorov, M.; Ong, M.T.; Sridharan, R.V.; Nakamura, J.; Ohmura, R.; Khor, J.H. Ultralightweight mutual authentication RFID protocol for blockchain enabled supply chains. *IEEE Access* **2019**, *7*, 7273–7285. [CrossRef]
37. 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]
38. Zou, S.; Xi, J.; Wang, S.; Lu, Y.; Xu, G. Reportcoin: A novel blockchain-based incentive anonymous reporting system. *IEEE Access* **2019**, *7*, 65544–65559. [CrossRef]
39. Javed, I.T.; Alharbi, F.; Margaria, T.; Crespi, N.; Qureshi, K.N. PETchain: A Blockchain-Based Privacy Enhancing Technology. *IEEE Access* **2021**, *9*, 41129–41143. [CrossRef]
40. Tang, F.; Ma, S.; Xiang, Y.; Lin, C. An efficient authentication scheme for blockchain-based electronic health records. *IEEE Access* **2019**, *7*, 41678–41689. [CrossRef]
41. Alamri, B.; Javed, I.T.; Margaria, T. A GDPR-Compliant Framework for IoT-Based Personal Health Records Using Blockchain. In Proceedings of the 2021 11th IFIP International Conference on New Technologies, Mobility and Security (NTMS), Paris, France, 19–21 April 2021; pp. 1–5.
42. Wu, J.; Xiong, F.; Li, C. Application of Internet of Things and Blockchain Technologies to Improve Accounting Information Quality. *IEEE Access* **2019**, *7*, 100090–100098. [CrossRef]
43. Liu, Y.; Wang, K.; Lin, Y.; Xu, W. A lightweight blockchain system for industrial internet of things. *IEEE Trans. Ind. Inform.* **2019**, *15*, 3571–3581. [CrossRef]
44. Hammi, M.T.; Hammi, B.; Bellot, P.; Serhrouchni, A. Bubbles of Trust: A decentralized blockchain-based authentication system for IoT. *Comput. Secur.* **2018**, *78*, 126–142. [CrossRef]
45. Niya, S.R.; Schiller, E.; Cepilov, I.; Maddaloni, F.; Aydinli, K.; Surbeck, T.; Bocek, T.; Stiller, B. Adaptation of Proof-of-Stake-based Blockchains for IoT Data Streams. In Proceedings of the 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Seoul, Korea, 14–17 May 2019; pp. 15–16.
46. Bachmann, S. Proof of Stake for Bazo. Bachelor's Thesis, University of Zurich, Zürich, Switzerland, 2018.
47. Du, M.; Ma, X.; Zhang, Z.; Wang, X.; Chen, Q. A review on consensus algorithm of blockchain. In Proceedings of the 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Banff, AB, Canada, 5–8 October 2017; pp. 2567–2572.