electronics

*Article*

# Mathematical Analysis of Parametric Characteristics of the Consensus Algorithms Operation with the Choice of the Most Priority One for Implementation in the Financial Sphere

**Olga A. Safaryan [1,*], Kirill S. Lemeshko [1], Alexey N. Beskopylny [2], Larissa V. Cherckesova [1] and Denis A. Korochentsev [1]**

[1] Department of Cyber Security of Information Systems, Don State Technical University, 344003 Rostov-on-Don, Russia; lemeshko-1996@mail.ru (K.S.L.); chia2002@inbox.ru (L.V.C.); mytelefon@mail.ru (D.A.K.)
[2] Department of Transport Systems, Faculty of Roads and Transport Systems, Don State Technical University, 344003 Rostov-on-Don, Russia; besk-an@yandex.ru
* Correspondence: safari_2006@mail.ru; Tel.: +7-(863)-238-15-18

**Abstract:** Blockchain is one of the leading data transfer technologies that eliminate the need for centralized management through consensus algorithms. This article describes the consensus algorithms, their benefits, and their applications within a micropayment system in the financial sector. Preliminary studies have shown that the performance of distributed databases largely depends on the chosen consensus algorithm. The main task of the study is to create a mathematical model to assess their performance. The most popular crypto projects and the consensus algorithms are analyzed to determine their performance. The obtained model was tested by calculating the parameters of the distributed register based on the directed acyclic graph algorithm and calculating the parameters of other algorithms used. The result is a mathematical model for evaluating the parametric characteristics of the work of consensus algorithms with the choice of the most priority one for implementation in the financial sector. The analysis focuses on the mathematical steps taken by each consensus algorithm. The data obtained using the developed mathematical model demonstrates that PoW, PoS, and DAG algorithms depend on various resources, such as computing power, the number of connected nodes, and the speed of receiving transactions.

**Keywords:** blockchain; consensus protocol; security; throughput capacity; distributed ledger (registry) technology; Proof of Work algorithm (PoW); Proof of Stake algorithm (PoS); Directed Acyclic Graph (DAG) system

check for updates

**Citation:** Safaryan, O.A.; Lemeshko, K.S.; Beskopylny, A.N.; Cherckesova, L.V.; Korochentsev, D.A. Mathematical Analysis of Parametric Characteristics of the Consensus Algorithms Operation with the Choice of the Most Priority One for Implementation in the Financial Sphere. *Electronics* **2021**, *10*, 2659. https://doi.org/10.3390/electronics10212659

Academic Editors: Amir Mosavi and Paulo Ferreira

Received: 5 October 2021
Accepted: 26 October 2021
Published: 30 October 2021

## 1. Introduction

The active use of an electronic computer has led to the creation and development of radically new and innovative technologies, such as Distributed Ledger (Registry) Technology, aimed at digitalization and decentralizing monetary relations in society [1].

The key direction of the technology is the economic sector; because of this technology, it is possible to eliminate all problems and limitations inherent in the methods of storing, accounting, and transmitting information currently used in the economy.

The investigation is relevant because the distributed ledger (registry) has become an integral part of decentralized applications that are actively used in many areas of industry and cryptocurrencies [2].

This technology is the basis for building any modern cryptocurrency, and it underlies the creation of distributed ledgers, smart contracts, and software for IoT devices. Furthermore, due to the technological potential of the distributed ledger technology, it is possible to get rid of intermediaries and third parties who provide financial services to the end consumer to ensure the comfortable and safe interaction of participants in the economic sphere of activity. Therefore, research on the implementation of distributed ledger systems is relevant [3].

In recent years, blockchain's dispersion (variance), irreversibility, and traceability [4] have caused growing concern. Consequently, many enterprises, governments, and scientific institutions have used blockchain-based applications in various fields, such as smart contracts [5,6], the Internet of things [7,8], and security services [9,10]. As a component in the development of software systems, blockchain can offer communication services, data storage, intellectual data analysis (data mining), and computing services. UBaaS (the single blockchain platform as the service platform) [8], proposed by Microsoft and IBM [11], has arisen based on the idea of service-oriented computational paradigms. In accordance with the BaaS paradigms, blockchain-based applications can be created by calling multiple blockchain services over the Internet. Developers (users) can quickly check and test their models and concepts within the framework of BaaS [12].

Blockchain provides decentralized, transparent, and secure systems. The distributed ledger (registry) technology fixes and records the performing transactions and protects them using cryptographic methods. All transactions are written in blocks, which are connected via hashes [13].

Blockchain has become a possible solution to the long-standing problem of user trust. The appearance of the well-known cryptocurrency Bitcoin provided such architecture that allows the user to trust the decentralized system instead of trusting the third side. Working in the peer-to-peer network, the user keeps his records in the transaction book. This helps to avoid any centralized system. The whole process is carried out on the basis (framework) of consensus. The ledger is commonly shared by several organizations, which allows everyone to view it. No user can control it. It is a distributed and cryptographically secured database that keeps accounts (records) of every transaction from the very beginning.

Below are the main features of blockchain technology.

1. Decentralized computing: the blockchain consists of distributed ledgers supported by peer-to-peer networks [14]. Blockchain eliminates the role of the central entity (subject) by using the consensus protocol to verify the transactions [15].
2. Distributed transaction ledger: the shared ledger is applied to store transactions [16,17]. A copy of this book is stored on each node of a blockchain network. Furthermore, these copies of the ledger are synchronized by timely replication [18].
3. Transparency: the blockchain stores each transaction in the block. In addition, it is available to all participants of the network for verification [19].
4. Security: each block is added to the chain after verification. Each block also contains the hash of the previous block [20]. It is computationally impossible to delete or update any block because this requires recalculating all previous block hashes.
5. Fault-tolerant network: the blockchain has a peer-to-peer network of nodes. All mining nodes process the transactions in parallel [21]. Therefore, the blockchain will continue to operate even if some nodes cannot function.

The blockchain network has several key characteristics–consensus, where all participants must agree with the transaction validity to be permissible.

- Origin of the money asset: all participants know where the money asset came from and how its ownership has changed over time.
- Immutability: no participant can interfere with a transaction after it has been recorded in the blockchain [22,23]. If there is an error in the transaction, then a new transaction must be used to correct this error, after which both are visible [24]. An error in the transaction means that it either failed or was rejected by the miners.
- Finality is a single shared network that provides one place to determine ownership of the money asset or to complete the transaction.

A smart contract is a computer program that supports the transfer of money or something of value. If a specific policy is followed, these programs are launched automatically. Each smart contract contains a contract address, predefined functions, and private storage [25]. A decentralized open-source platform that executes smart contracts is Ethereum [26].

The purpose of this work is to substantiate the applicability of the distributed ledger (registry) technology in the field of finance mathematically.

## 2. Materials and Methods

The protocol and the consensus algorithm are responsible for organizing the work of the distributed ledger (registry) system. The protocol is the rules of the distributed ledger operation, according to which the network nodes interact, transmit the data, and confirm data entry. A consensus algorithm is a mechanism for reaching an agreement on the current state of data, which exercises the control and monitoring compliance with the protocol rules and the reliability of the data entered.

In other words, the consensus mechanism is responsible for ensuring that all nodes of the distributed network confirm the introduction of new data into it. This is how consensus supports the integrity and security of the network [27].

Most cryptocurrencies are based on two main mechanisms for achieving consensus—"Proof-of-Work" (PoW) and "Proof-of-Stake" (PoS), as shown in Figure 1. Bitcoin is the first project with high market value (about USD 47,000 per coin) developed on the "Proof-of-Work" mechanism. This is a classic consensus mechanism in distributed ledger technology. The "Proof-of-Stake" algorithm is used to solve the problem of PoW restriction and is the basis of various blockchain systems, such as Nxt, Lisk, EOS, and Cardano. In world practice, both of these mechanisms are often used for building distributed financial systems. However, they have serious disadvantages (low scalability, low transaction processing speed, high-energy consumption, and too-large amounts of stored data).

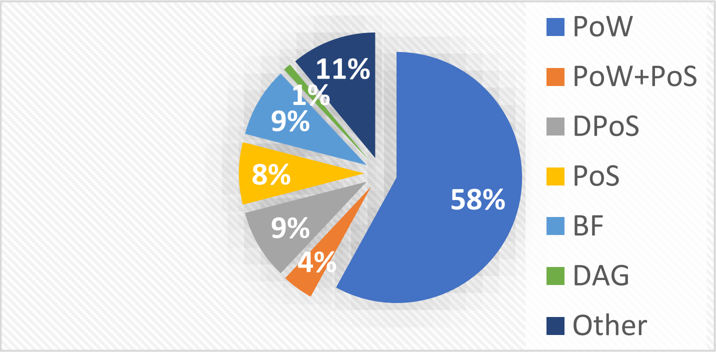

**Figure 1.** Percentage distribution of the most popular consensus algorithms.

It is the consensus algorithm that determines the main parametric characteristics of the distributed ledger (registry). These include the data processing speed, scalability, security, and volume of stored registry data, as shown in Table 1 [28].

Various permissions and roles can be assigned to the system nodes, such as:

- User (device, legal entity, or individual who has the right to make changes to the ledger);
- Miner (node to which the right to update the registry is delegated);
- Intermediary (communication between the system and external participants);
- Manager, developer (organizes the work of the system, regulates disputable situations).

In the case of central node absence that has authority to update the ledger (or registry), the nodes confirming the data correctness reach the consensus regarding the general state of a distributed ledger. Reaching the consensus consists of two main stages:

- Verification, during which every identifying node controls and checks the correctness of changes made to the ledger (registry);
- Reaching consensus on updating information in the register and approving changes (mechanisms or algorithms that do not allow conflicts to arise in the registry are used).

**Table 1.** Comparative characteristics of consensus algorithms.

| Cryptocurrency | Consensus | TPS (Transactions per Second) | Block Size, MB | Confirmation Time, s | Occupied Volume, GB |
|---|---|---|---|---|---|
| Bitcoin | PoW | 7 | 1 | 600 | 346 |
| Ethereum | PoW | 25 | 1 | 12 | 237 |
| Bitcoin Cash | PoW | 100 | 32 | 600 | 164 |
| Cardano | PoS | 200 | - | - | 8 |
| EOS | PoS | 4000 | - | 0.5 | 8 GB RAM |
| Peercoin | PoS | 8 | 1 | 510 | - |
| IOTA | DAG | 1000 | - | - | - |
| NANO | DAG | 7000 | - | ~10 | - |
| Hashgraph | DAG | 10,000+ | - | 4 | - |

### 2.1. The "Proof-of-Work" Consensus Algorithm

The "Proof–of–Work" is the consensus algorithm, which was first used in the work of the Bitcoin cryptocurrency. In 2004, Hal Finney proposed the idea of using the PoW mechanism for electronic currencies. In 2008, this algorithm formed the basis of the Bitcoin cryptocurrency. Later, the algorithm was used to create such new cryptocurrencies as Litecoin, Bitcoin Cash, Bitcoin Gold, Dash, Dogecoin, Monero, and Zcash, and others.

The essence of this mechanism is as follows: the nodes of a distributed network must solve complex mathematical problems in order to confirm transactions. The node that finds the solution first receives the corresponding reward—crypto coins. The complexity of mining allows protection of the network from possible threats in the form of DDoS attacks, 51% attacks, and other attacks. If solving mathematical problems was easy, attackers could easily hack the network.

The "Proof–of–Work" algorithm was a breakthrough for its time and allowed the first cryptocurrencies to be launched on the global financial market. The PoW algorithm ensures the decentralization of the network and allows the network to be made resistant to hacking. In the case of an attack on the distributed database, the attacker must solve the same cryptographic problem as the remaining nodes of the network, i.e., the attack will be successful only if an attacker can significantly exceed the computing resources of the remaining nodes.

The principle of operation of the proof mechanism is such that the following resources support the network security:

- A computer for performing the calculations;
- Electricity for the equipment operation.

This makes the algorithm inefficient in terms of resource consumption. To increase their remuneration, miners are forced to participate in the so-called "arms race", that is, to use more and more resources for cryptocurrency mining. This makes the cost of attacking the distributed network prohibitively high.

### 2.2. The "Proof-of-Stake" Consensus Algorithm (PoS)

The second most popular solution in the field of ensuring consensus was the method of confirming ownership of stake. The essence of this algorithm is that the right to create a new block and receive the reward is distributed randomly among all nodes that own a specific stake (share) of the system's asset.

The technical justification for the effectiveness of the stake confirmation mechanism is as follows: the nodes with the most significant stake of the system's assets have priority in maintaining network security since they will lose the most if the reputation and value of the cryptocurrency fall as a result of cyberattacks.

To carry out a successful cyberattack, an attacker must have a large amount of currency at his disposal, which will be costly if the system is popular enough.

There is no mining process in the stake confirmation algorithm. Instead of solving complex cryptographic problems, new coins are mined through a staking mechanism that allows adding new blocks by proving the ownership of network assets. Nodes in this system are called validators, and their balance is called the stake. The more coins the node has in its wallet, the more chances it has to confirm a new block and receive the corresponding reward [29]. The stacking process can be compared to a bank deposit—the more assets the node has at its disposal, the higher the reward. For users, this is an opportunity to earn passive income.

However, stacking also requires expenses. To confirm the deposit of the block and receive the reward, one must have the minimum required number of coins. Also, it is necessary to keep the equipment connected to the network, which adds to electricity cost. The main advantages of the stake confirmation algorithm:

- Power consumption is lower in comparison with the confirmation mechanism;
- There is no special equipment;
- High speed and scalability in relation to the work confirmation mechanism (for example, the speed of the EOS network is 4000 transactions per second, TPS);
- Low commissions;
- Participation in the further development of the project.

The main disadvantage of the “Proof–of–Stake” confirmation algorithm is the threat of centralization. The users with the most coins will eventually control most of the network. Therefore, new versions of the stake confirmation algorithm are being developed actively. Delegated “Proof–of–Stake” (delegated proof of ownership shares, DPoS) is a kind of stake ownership algorithm. The algorithm of delegated proof of stakes is an alternative to PoW and PoS mechanisms; its idea is to deprive validators of the probability of the system centralization. Among the well-known distributed ledgers (registries) based on the DPoS algorithm are EOS, Steemit id, and Tezos.

The main difference between DPoS and PoS algorithms is that in the delegated algorithm, coin holders transfer their right to confirm the transaction and the right to receive the reward, that is, they delegate their rights to a predetermined validator. Any node of the system can become the delegate. However, the holders can withdraw their vote back at any time—this method allows avoiding excessive centralization and the seizure of the network by unscrupulous participants.

Delegates are united into groups (pools) that have the right to change some system parameters, such as the average time of new block minings, size, etc. However, delegates are unable to cancel transactions or conduct false transactions.

The main advantages of the delegated stake ownership algorithm are high speed and greater scaling. In addition, the system has significantly fewer nodes than in PoW or PoS algorithms, which allows faster creation of new blocks.

Disadvantages of the algorithm: the threat of centralization with a small number of system participants and the threat of DDoS cyberattacks and dishonest behavior of delegates, which can cause system failures.

The “Proof–of–Importance” algorithm is another variation of the stake confirmation mechanism. In the “Proof–of–Importance” algorithm, the number of crypto coins is important, as are the user’s activity, the number of transactions made, and the time of working spent in the system. The higher the activity of the node, the higher its reputation in the community, and, accordingly, the higher the income from owning coins. Due to this, users actively use coins rather than just storing them in wallets.

The “Leased Proof-of-Stake” (LPoS) algorithm is an alternative solution based on the “Proof-of-Stake” algorithm explicitly developed for the “Waves” cryptocurrency.

The mechanism of leased proof of ownership is designed to solve the problem of “property qualification” in the classical PoS algorithm. The users with insufficient balance cannot participate in the process of confirming blocks and earning the new assets, which leads to centralization.

Within the framework of this algorithm, any node of the system can transfer its coins to the validator, obtaining the stake of the profit received in exchange. Crypto coins remain in the wallet but cannot be used, transferred, or exchanged. The rental process can be canceled at any time. It is profitable to rent coins as a validator, since this increases their cryptocurrency balance, along with the chance of the reward.

This algorithm allows earning crypto coins with only a small balance at your disposal but makes the system susceptible to centralization: validators can rent so many coins that they can take control of the system.

### *2.3. The Algorithm for Achieving Consensus Based on Directed Acyclic Graph*

An alternative to the blockchain is the Directed Acyclic Graph (DAG). This is a directed graph with no directed cycles, whose edges go in the same direction, and there are no closed loops, i.e., there is no way to go back to the beginning of the same node, and no element can be considered a child.

DAG is very similar to the linked list. An example of the directed graph structure is demonstrated in Figure 2 [30].

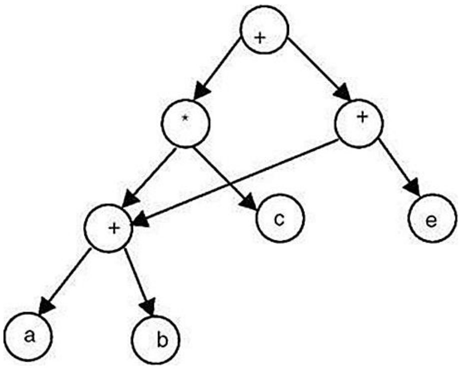

**Figure 2.** Structure of directed acyclic graph.

In the minimal form, the DAG system has 4 components:

6. The nodes representing some device or user.
7. Directed edges; directed edge from one node to another represents a specific relationship between these two nodes.
8. Root node; one of the nodes will not have parents. This is the root of DAG.
9. Leaves or nodes without child nodes.

The operation of an acyclic graph can be represented in general form:

- Network nodes exchange information according to the given protocol.
- Information, for example, about the transaction, is sent to the two random nodes that transmit them, in turn, to the two other nodes and so on, exponentially, until the number of informed nodes is not enough to verify the transaction.
- Nodes exchange with each other only information about transactions and not all the information of the network, while the data is stored not in the blocks but in the hashes, which increases the speed of the network.
- Transactions are recorded in chronological order.

The sequence of transactions is called the branch, and the longer the branch of a transaction is verified by a person, the more weight it has. Each node knows the entire transaction history, so to achieve consensus, they use “virtual voting”. There is no necessity to coordinate all the nodes among themselves—each of them already has an idea of how the other will “vote”. All nodes in the directed acyclic graph system perform the double function: not only verification, but also a representation of the verified transaction.

The reliability of the algorithm is described mathematically as asynchronous Byzantine Error Tolerance (ABET). This algorithm is almost impossible to crack in practice, and influencing the network would require the consent of more than two-thirds of users.

Currently, most distributed ledger (registries) projects operate on the principle of blockchain and have good but not sufficient performance (efficiency) indicators. Therefore, there is a global trend to find the solutions to the main problems inherent in the blockchain, modernize it, or search for alternative solutions to the organization of distributed ledger (registry) systems.

A directed acyclic graph is an excellent alternative because it absorbs all advantages of the blockchain but without its disadvantage in the form of low transaction speed. The more transactions that occur on the network, the faster it becomes. In addition, a directed acyclic graph is highly scalable and takes up fewer data volumes than blockchain [31].

Additionally, the acyclic graph's protection against quantum cyberattacks is ensured by the system of one-time signatures, which acts as a firewall against hacking attempts by quantum computers. The problem of scalability and large volumes of occupied data is solved. The analysis of consensus-building algorithms typical for distributed ledgers (registries) has shown that the PoW and PoS mechanisms have received wide mass distribution and become widespread. Most of the proposed consensus mechanisms are based on them; however, these algorithms have problems inherent in distributed ledger (registry) technology. To assess the effectiveness of consensus algorithms proposed for implementation in the financial sector, it is necessary to analyze the most typical implementations (crypto projects) of the PoW and PoS algorithms and the performance of a promising DAG algorithm. Based on the analysis performed, it is necessary to select a set of formulas and, on their basis, build a mathematical model for evaluating and comparing the parameters and efficiency of the algorithms. The data obtained during the implementation of the mathematical model will become the basis for creating proposals for the performance of distributed ledger technology in the financial sector.

A mathematical model of the algorithms must be built to compare their performance and efficiency and to make proposals for the introduction of distributed ledger (registry) technology in the financial sphere.

## 3. Results and Analysis

The algorithms for achieving consensus considered in this paper differ from other rules for building and operating and have specific positive and negative characteristics. To identify the most suitable and effective algorithm from the ones presented earlier, it is necessary to conduct a mathematical analysis of these algorithms. Based on the results obtained, the most effective solution for using the distributed ledger (registry) in the financial sphere of activity will be proposed.

For comparison, it was assumed that in the distributed system, $n$ nodes are operating on the base of the PoW, PoS, and DAG consensus mechanisms, where all nodes are connected directly to a single-speed network. The transfer delay is not included in the calculation since it is a lot less than the calculation time and the interval for the arrival of a new transaction. In the DAG mechanism, the new transaction must approve the previous transactions as soon as possible. Thus, any vertex that records only one transaction can be considered a block. Therefore, the maximum number of new transactions is $K = 1$. The new transaction follows the Poisson distribution, with the speed of arrival $\lambda_i$ at node $i$, and the weight of each transaction equal to one.

In order to perform the mathematical analysis of the parametric characteristics of the consensus algorithms and to identify the most priority algorithm for implementation in the financial sphere, an algorithm efficiency parameter was introduced, which is determined based on the ratio of the average time of generation of the transaction, the confirmation delay, and the number of transactions per second:

$$F = \frac{\mathrm{C} * \mathrm{TPS}}{V} \tag{1}$$

where $V$ is the average transaction generation time, which determines the time for which the new block will be added to the ledger (registry); TPS is the number of transactions per second; and $C$ is the transaction confirmation delay.

The average time for the creation of the new block $V$ determines the time for which this new block will be added to the ledger (registry) and shows the block processing speed.

The average block generation time in the PoW algorithm is defined as:

$$V_{pow} = \frac{D}{\sum_{i=1}^{n} r_i} \tag{2}$$

where $D$ is the target complexity; and $r_i$ is the computing power of the node.

The average block generation time in the PoS algorithm is defined as:

$$V_{pos} = \frac{D}{\sum_{i=1}^{n} b_i * t_i} \tag{3}$$

where $b_i$ is the age of the asset; and $t_i$ is the time since the moment of the last block generation.

The average block generation time in the DAG algorithm is defined as:

$$V_{dag} = \frac{1}{\sum_{i=1}^{n} \lambda_i} \tag{4}$$

The confirmation delay $C$ shows the processing speed of the transaction from the moment of its creation (generation) to the moment of its confirmation.

The confirmation delay in the PoW algorithm is defined as:

$$\mathrm{C}_{PoW} = k_{PoW} * V_{PoW} \tag{5}$$

where $k_{PoW}$ is the number of blocks.

The confirmation delay in the PoS algorithm is defined as:

$$\mathrm{C}_{PoS} = k_{PoS} * V_{PoS,} \tag{6}$$

where $k_{PoS}$ is the number of blocks.

The confirmation delay in the DAG algorithm is defined as:

$$C_{dag} = \frac{W}{H(t)} \tag{7}$$

where $W$ is the cumulative weight threshold for confirmation in the DAG; and $H(t)$ is the cumulative weight, which is defined as:

$$H(t) = \begin{cases} 1 + \sum\limits_{i=1}^{n} \lambda_i, t \in (0,\infty), \sum\limits_{i=1}^{n} \lambda_i \le \frac{1}{h_r} \\ 2\exp\left(\frac{0.325t}{h_r}\right), t \in (0,\infty), \sum\limits_{i=1}^{n} \lambda_i > \frac{1}{h_r} \\ 2\exp\left(\frac{0.325t_0}{h_r}\right) + \sum\limits_{i=1}^{n} \lambda_i(t - t_0), t \in (0,\infty), \sum\limits_{i=1}^{n} \lambda_i > \frac{1}{h_r} \end{cases}. \tag{8}$$

The number of transactions per second (TPS) is necessary in order to show the throughput capacity in the distributed ledger (registry).

The TPS in the PoW algorithm is defined as:

$$\mathrm{TPS}_{pow} = \begin{cases} \sum\limits_{i=1}^{n} \lambda_i, \sum\limits_{i=1}^{n} \lambda_i \le \frac{K}{V_{pow}} \\ \frac{K}{V_{pow}}, \sum\limits_{i=1}^{n} \lambda_i > \frac{K}{V_{pow}} \end{cases} \tag{9}$$

where $\sum\limits_{i=1}^{n} \lambda_i \le \frac{K}{V_{pow}}$ is a low load scenario; and

$\sum\limits_{i=1}^{n} \lambda_i > \frac{K}{V_{pow}}$ is a heavy load scenario.

The TPS in the PoS algorithm is defined as:

$$\mathrm{TPS}_{pos} = \begin{cases} \sum\limits_{i=1}^{n} \lambda_i, \ \sum\limits_{i=1}^{n} \lambda_i \leq \frac{K}{V_{pos}} \\ \frac{K}{V_{pos}}, \ \sum\limits_{i=1}^{n} \lambda_i > \frac{K}{V_{pos}} \end{cases} \tag{10}$$

The TPS in the DAG algorithm is defined as:

$$\mathrm{TPS}^{dag} = \sum_{i=1}^{n} \lambda_i \tag{11}$$

The probability of the failure $p$ of the transaction confirmation shows with which probability the new transaction can be rejected by the system.

The probability of the PoW confirmation failure is defined as:

$$P_{pow} = \begin{cases} 0, \ \sum\limits_{i=1}^{n} \lambda_i \leq \frac{K}{V_{pow}} \\ 1 - \frac{K}{\sum_{i=1}^{n} \lambda_i * V_{pow}}, \ \sum\limits_{i=1}^{n} \lambda_i > \frac{K}{V_{pow}} \end{cases}. \tag{12}$$

The probability of the failure of the PoS confirmation is defined as:

$$P_{pos} = \begin{cases} 0, \ \sum\limits_{i=1}^{n} \lambda_i \leq \frac{K}{V_{pos}} \\ 1 - \frac{K}{\sum_{i=1}^{n} \lambda_i * V_{pos}}, \ \sum\limits_{i=1}^{n} \lambda_i > \frac{K}{V_{pos}} \end{cases} \tag{13}$$

The probability of the failure of the DAG confirmation is zero, since each incoming transaction becomes a new vertex of the graph until the other nodes of the system confirm it.

Evaluating the effectiveness of the algorithms of the PoW, PoS, and DAG mechanisms in terms of the average time of the new block generating, confirmation delay, TPS, and the probability of confirmation failure is the following. Suppose that distributed network includes ten nodes ($n$ = 10). The detection time in the DAG algorithm is $h_r$ = 1 s (one second), and the weight threshold for confirmation is $W$ = 200. The weight of each transaction is equal to 1, and thus the total weight is gradually increased by 1.

The average block creation time in PoW, PoS, and DAG is affected by computing power, the coin's age, and the speed of new transaction receipts, as shown in Figure 3 ($i$, the step of increasing computing power).

It is clear that with the increase of resources for consensus reaching, the average time for new block generating becomes lower. The mechanism of the PoW algorithm requires a lot of computing power, which may be impractical for light nodes (for example, for mobile equipment). Therefore, the PoS algorithm that has the same procedure, but does not require high computing power, may be the best choice.

Unlike the PoW and PoS mechanisms, in the DAG, there is no miner, and the new transaction must confirm the previous transactions in order to be confirmed in turn. Thus, the faster the arrival speed, the less time it takes to create any new block. To solve this problem, DAG is more suitable for a scenario in which the transactions occur frequently (for example, in micropayments).

The average time of new block generation in the DAG algorithm is much faster than in the PoW and PoS algorithms, but the block size in the DAG algorithm (in which the block stores only one transaction) is much smaller than in the PoW and PoS algorithms (where hundreds of transactions are stored in the block).

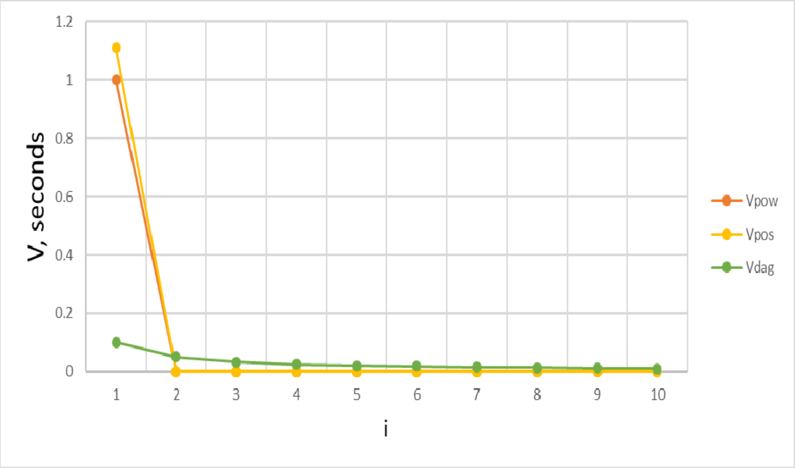

**Figure 3.** Average time to create the new block.

It is noticeable that with an increase in computing power resources in the balance of coins and transactions, a consensus can be achieved much faster. Due to different requirements for conformation (the typical value in bitcoin $k_{PoW}$–9 in Peercoin, $k_{PoS}$ = 15 and W = 200 in the directed graph system), delays in the confirmation of consensus mechanisms differ (vary) greatly, as shown in Figure 4.

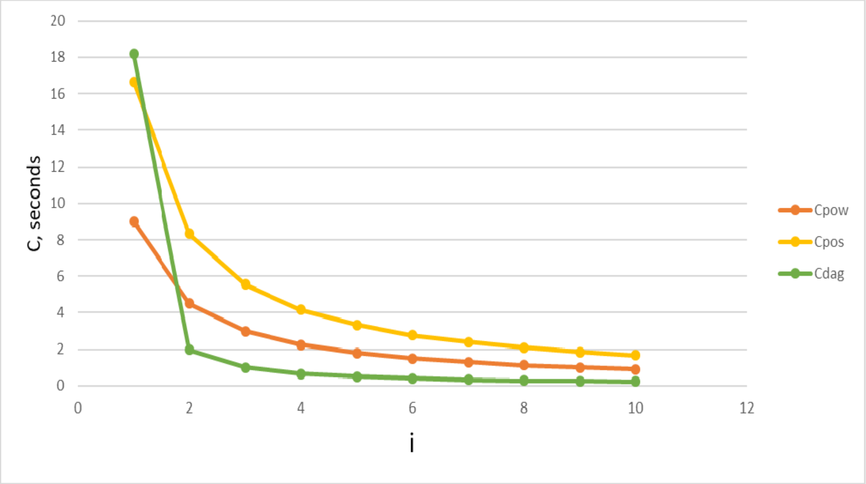

**Figure 4.** The delay of algorithm confirmation.

In the DAG mechanism, a transaction is confirmed only when its total weight reaches the threshold value *W*. It is noticeable that an increase in the number of incoming transactions reduces the transaction confirmation delay to zero.

In order to show the impact of the network load on the network operation, the entering speed of new transactions is gradually changed from the low load mode to the high load mode, as shown in Figure 5.

The TPS parameter (the number of transactions per second) in the PoW and the PoS algorithms first increases linearly to the value TPS = $K/V$, as they have reached the block size limit. TPS in the DAG mechanism, on the contrary, always increases without restrictions when new transactions arrive. The DAG-based distributed ledger (registry) allows the application of new transactions to achieve a consensus, and its TPS, technically, has no upper bound. However, it is worth noting that the TPS parameter in the DAG algorithm may decrease significantly if a new transaction is slow to arrive, and in some extreme conditions, a consensus may not even be achieved.

For the mechanisms of operating the PoW and the PoS algorithms, the increase in the speed of new transaction arrivals overloads the network, giving rise to the probability of failure. Due to this, the node may discard the new transaction.

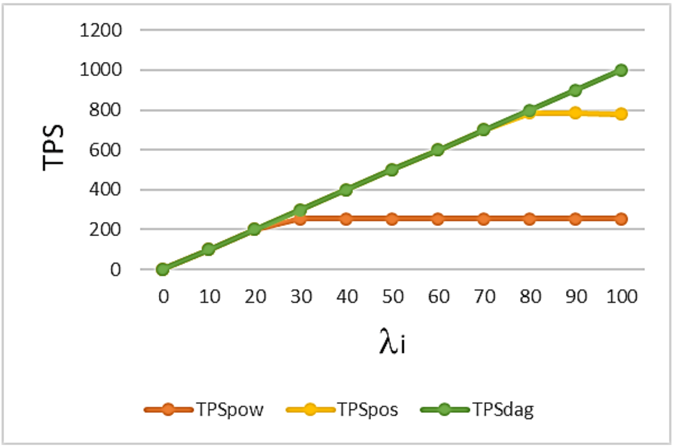

**Figure 5.** The comparison of the TPS parameter for different algorithms.

In the DAG mechanism, this probability of failure is absent, since any new transaction will be the vertex in the DAG until other nodes confirm it, and the failure cannot be caused by restrictions on the size of the block and the queue. Therefore, the corresponding probability of confirmation failure in the DAG remains zero at all times, as shown in Figure 6.

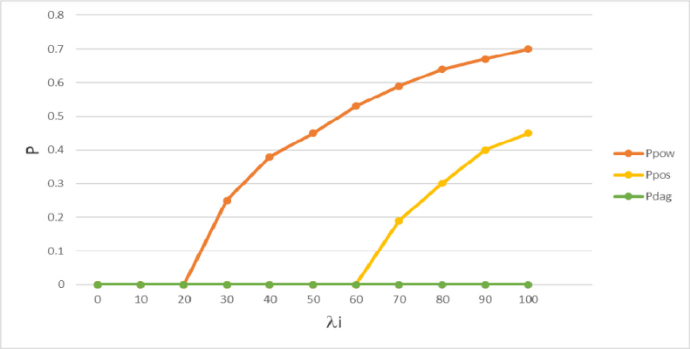

**Figure 6.** The probability of the transaction confirmation failure.

The cumulative distribution function of the directed graph increases the fastest; the reason being that the DAG algorithm allows the new transaction to join the distributed network at the time of its creation. The cumulative function of the PoW mechanism during the first second is higher than that of the PoS algorithm, but when the time is more than 1 s, the cumulative function of the PoS algorithm becomes higher than that of the PoW algorithm. This occurs because the age of the coin increases over time, and the probability of calculating the value for the hash operation increases with it, as shown in Figure 7.

Based on the calculations performed, and in accordance with Formula (1), the graph of the algorithms' operating efficiency dependence on the various parameters was constructed. The graph shows that with 10 nodes connected to the system, the work efficiency of the PoW and PoS algorithms increases linearly to a certain level, and then remains there. This is due to the dependence of the system on the size of the blocks and node resources. The efficiency of the DAG mechanism's operating continues to increase without restrictions, since this algorithm does not consume a large number of resources and depends only on the number of nodes connected to the system.

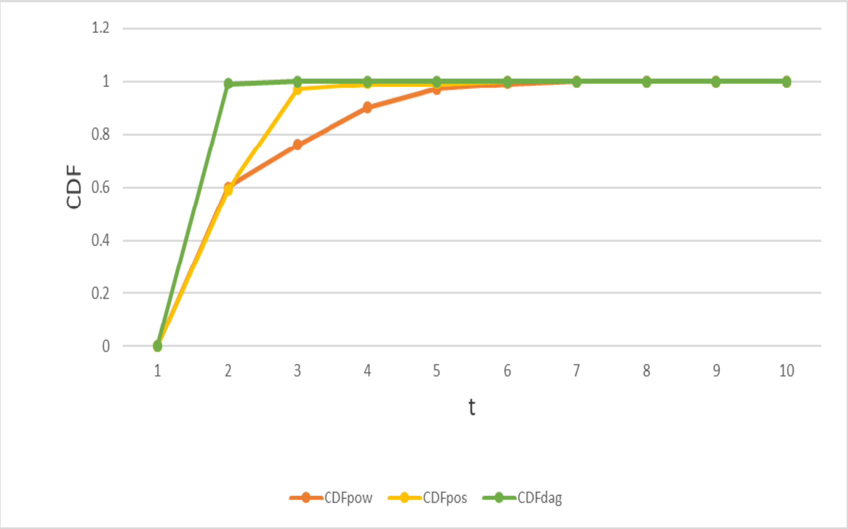

**Figure 7.** Cumulative distribution function.

To show the impact of nodes on algorithm work efficiency, calculations were performed for 5 and 25 nodes connected to the distributed ledger (registry) based on the PoW, PoS, and DAG algorithms. It is evident that even with a small number of connected nodes, the efficiency of the DAG algorithm continues to grow, while the PoW and PoS algorithms have reached their limit and remain at the same level, as shown in Figure 8.

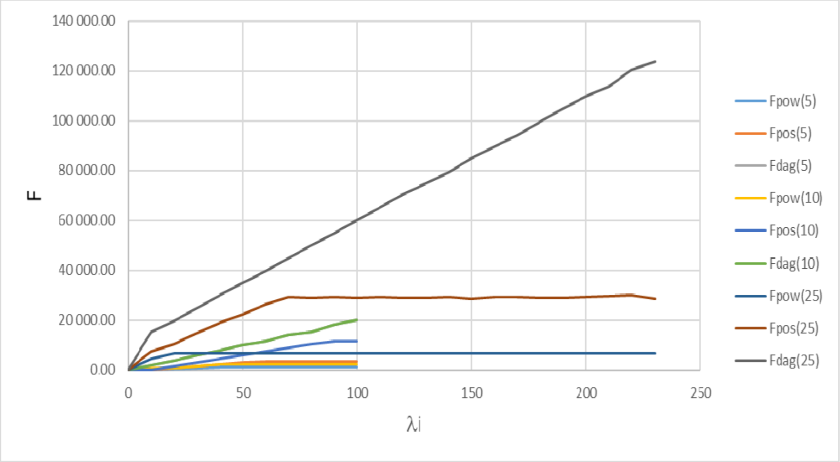

**Figure 8.** Efficiency of the work of algorithms operating at 5, 10, and 15 nodes.

Based on the obtained results of the constructed mathematical model, it can be concluded that the operating of the PoW, PoS, and DAG algorithms depends on various resources, including the computing power, the number of connected nodes, and the entering (receiving) speed of the transaction. The efficiency of the DAG algorithm work depends only on the presence of the connected nodes in the system; even at their small number, the system remains operational at a high level, while the PoW and PoS algorithms are limited by block sizes and the computing power of nodes. The overall efficiency of the algorithms is presented in Table 2.

It must be clarified that PoW and PoS are mechanisms based on competition, and DAG is a mechanism focused on accumulation. The PoW and PoS algorithms have similar tendencies because they both use the hash algorithm to perform the consensus process. On the contrary, the DAG algorithm motivates new transactions to confirm previous ones, rather than using heavy hash calculations.

**Table 2.** Overall efficiency of algorithms.

| n, Nodes | Parameter | PoW | PoS | DAG |
|---|---|---|---|---|
| 5 nodes | Average time of block generation, s | 0.2 | 0.5 | 0.004 |
| | Delay of confirmation, s | 1.8 | 7.5 | 0.99 |
| | TPS (Number of transactions per second) | 127.5 | 223 | 250 |
| | Effectivity, tr/s | 0.1 | 3.3 | 20 |
| 10 nodes | Average time of block generation, s | 0.1 | 0.11 | 0.01 |
| | Delay of confirmation, s | 0.9 | 1.7 | 0.22 |
| | TPS (Number of transactions per second) | 255 | 784 | 1000 + |
| | Effectivity, tr/s | 0.3 | 12 | 20 |
| 25 nodes | Average time of block generation, s | 0.04 | 0.017 | 0.0016 |
| | Delay of confirmation, s | 0.36 | 0.25 | 0.03 |
| | TPS (Number of transactions per second) | 765 | 1960 | 6000 |
| | Effectivity, tr/s | 6.9 | 28 | 124 |

## 4. Discussion

The authors presented the results of their research with the involvement of major domestic and foreign specialists in the field of distributed ledger (registry) technology.

Practical implementation of research results can lead to the analysis and evaluation of this technological application in the field of finance. Today, decentralized computing systems and data exchange are very popular in various fields.

This article explains the consensus algorithms associated with blockchain and their advantages and applications within the micropayment system in the financial sector.

In addition, the article analyzes the usefulness of the application of blockchain technology and provides a brief overview of various protocols (consensus algorithms), along with the advantages and disadvantages of their usage. The consensus methods considered in other scientific papers allow for the analysis of the work of blockchain technology, i.e., an ecosystem is being created in which security is initiated within the framework of this technology, when cryptocurrencies, such as bitcoin, are distributed to other blockchains. This type of mining is represented in consensus algorithms, where the current state of the blockchain is presented to another blockchain. In this direction, publication materials about the current state of the blockchain appear periodically.

The results of studies from other scientific groups confirm the conclusions obtained in the course of this study. Thus, in [32], it is indicated that an ideal consensus algorithm is still unattainable since almost all algorithms, in one way or another, have significant drawbacks in terms of their safety and performance. However, other distributed accounting systems do not rely on any structure similar to blockchain. Instead, they use different structures to represent their respective ledgers. Examples of two such well-known blockchain systems are IoTA (IOTA, 2019) and NANO (Nano, 2019). Both of these distributed ledgers are DAG-based. The two systems have received considerable attention due to their commission-free structure and fast transaction rates.

Having considered the material described earlier within the framework of this study, a mathematical model was built that proves the possibility of an application and evaluation of the PoW, PoS, and DAG consensus algorithms in the field of finance. The DAG algorithm is defined as ideal for micropayment systems because it has "almost zero" commissions.

## 5. Conclusions and Future Work

This paper proposed an effective and applicable consensus algorithm and showed that the DAG algorithm is ideal for the micropayment system because it has "almost zero" commissions. Due to its architecture and the absence of miners in the network, the DAG-based network users will be able to send transactions with minimal or no commissions at all. Users serve the network themselves every time they send a new transaction. Therefore, there are usually no miners and "master nodes" in the DAG algorithm. Still, participants must determine the order of all records because, without an exact order, a "double spending" cyberattack is still possible. The presence of these participants and the process of transaction validation (confirmation) on the user's side together form the consensus algorithm for records in the DAG networks.

In addition, in the PoW and PoS algorithms, single-chain architecture is necessary to prevent branching, thus limiting the corresponding productivity. The confirmation delay can be as long as 60 min (or 7 TPS) in Bitcoin and 3 min (or 20 to 30 TPS) in PeerCoin, which is excessively long. Due to the multi-chain architecture, the new transaction can be inserted into the DAG-chain with the maximum possible speed. Technically, there is no upper bound on TPS.

The article implements several scenarios for the PoW, PoS, and DAG consensus mechanisms, where all nodes are connected directly to the single-speed network. The transfer delay is not taken into account since it is much less than the computing time and the interval of the new transaction's arrival. In the DAG mechanism, the new transaction must confirm (approve) the previous transactions as soon as possible. Thus, any vertex can be considered a block that records only one transaction.

The authors have investigated the influence of each factor separately when conducting the mathematical analysis with parametric characteristics of the operation of consensus algorithms and have identified the most priority algorithm for its implementation in the financial sphere—the Directed Acyclic Graph (DAG) algorithm.

**Author Contributions:** Conceptualization, O.A.S., K.S.L., L.V.C., A.N.B. and D.A.K.; methodology, O.A.S., K.S.L. and L.V.C.; software, O.A.S., K.S.L. and D.A.K.; validation, O.A.S., K.S.L. and L.V.C.; formal analysis, O.A.S., K.S.L. and L.V.C.; investigation, O.A.S., K.S.L. and L.V.C.; resources, A.N.B.; data curation, O.A.S., K.S.L. and L.V.C.; writing—original draft preparation, O.A.S. and A.N.B.; writing—review and editing, O.A.S. and A.N.B.; visualization, O.A.S.; supervision, L.V.C. and A.N.B.; project administration, L.V.C. and D.A.K.; funding acquisition, A.N.B. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** The study did not report any data.

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

## References

1. Safaryan, O.A.; Lemeshko, K.S.; Aldyrev, M.N. Analysis of the practical implementation of distributed ledger (registry) technology. In Proceedings of the Progressive Technologies and Processes, Kursk, Russia, 24–25 September 2020; Southwest State University Kursk: Kursk, Russia, 2020; pp. 87–92. Available online: https://www.elibrary.ru/item.asp?id=44065996 (accessed on 25 October 2021).
2. Safaryan, O.A.; Lemeshko, K.S.; Cherkesova, L.V. Distributed ledger (registry) as technological basis of decentralized system of finance. In Proceedings of the Actual Problems of Science and Technology, Rostov-on-Don, Russia, 25–27 March 2020; Don State Technical University: Rostov-on-Don, Russia, 2020; pp. 932–935. Available online: https://www.elibrary.ru/item.asp?id=44086179 (accessed on 25 October 2021).
3. Safaryan, O.A.; Aldyrev, M.N.; Cherkesova, L.V. Application of distributed registry technology in Russia. In Proceedings of the Infocommunication Technologies: Current Issues of the Digital Economy, Yekaterinburg, Russia, 17–18 February 2021; Siberian State University of Telecommunications and Informatics: Novosibirsk, Russia, 2021; pp. 197–200. Available online: https://www.elibrary.ru/item.asp?id=45847218 (accessed on 25 October 2021).
4. Zheng, P.; Zheng, Z.; Luo, X.; Chen, X.; Liu, X. A Detailed and real-time performance monitoring framework for blockchain systems. In *ICSE-SEIP'18: Proceedings of the 40th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP), Gothenburg, Sweden, 30 May–1 June 2018*; IEEE Computer Society: Los Alamitos, CA, USA, 2018; pp. 134–143. [CrossRef]
5. Zhang, Y.; Kasahara, S.; Shen, Y.; Jiang, X.; Wan, J. Smart contract-based access control for the internet of things. *IEEE Internet Things J.* **2019**, *6*, 1594–1605. [CrossRef]
6. Wang, S.; Ouyang, L.; Yuan, Y.; Ni, X.; Han, X.; Wang, F. Blockchain-enabled smart contracts: Architecture, applications, and future trends. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *49*, 2266–2277. [CrossRef]
7. Liu, Y.; Wang, K.; Qian, K.; Du, M.; Guo, S. Tornado: Enabling Blockchain in Heterogeneous Internet of Things through a Space-Structured Approach. *IEEE Internet Things J.* **2020**, *7*, 1273–1286. [CrossRef]
8. Liang, W.; Huang, W.; Long, J.; Zhang, K.; Li, K.; Zhang, D. Deep reinforcement learning for resource protection and real-time detection in IoT environment. *IEEE Internet Things J.* **2020**, *7*, 6392–6401. [CrossRef]

9. White, B.S.; King, C.G.; Holladay, J. Blockchain security risk assessment and the auditor. *J. Corp. Account. Financ.* **2020**, *31*, 47–53. [CrossRef]
10. Liang, W.; Fan, Y.; Li, K.; Zhang, D.; Gaudiot, J. Secure data storage and recovery in industrial blockchain network environments. *IEEE Trans. Ind. Inform.* **2020**, *16*, 6543–6552. [CrossRef]
11. Lu, Q.; Xu, X.; Liu, Y.; Weber, I.; Zhu, L.; Zhang, W. uBaas: A unified blockchain as a service platform. *Future Gener. Comput. Syst.* **2019**, *101*, 564–575. [CrossRef]
12. Sumaira, J.; Naveed, A.; Warda, A.; Haitham, C.; Amad, D. Research and applied perspective to blockchain technology: A comprehensive survey. *Appl. Sci.* **2021**, *11*, 6252. [CrossRef]
13. Velde, F. Bitcoin: A Primer. In *Chicago Fed Letter*; Federal Reserve Bank of Chicago: Chicago, IL, USA, 2013. Available online: http://blog.philippe-poisse.eu/public/monnaie_locale/bitcoin/cfldecember2013_317.pdf (accessed on 25 October 2021).
14. Swan, M. Blockchain thinking: The brain as decentralized autonomous corporation. *IEEE Technol. Soc. Mag.* **2015**, *34*, 41–52. [CrossRef]
15. Bahga, A.; Madisetti, V. Blockchain Applications: A Hands-On Approach, Vpt. 2017. Available online: https://www.semanticscholar.org/paper/Blockchain-Applications%3A-A-Hands-On-Approach-Bahga-Madisetti/57e6db297a1e03c8fd4402102523faadaa3e71c0 (accessed on 6 October 2018).
16. El Ioini, N.; Pahl, C. A review of distributed ledger technologies. In Proceedings of the OTM Confederated International Conferences: On the Move to Meaningful Internet Systems, Valletta, Malta, 22–26 October 2018; Springer: Cham, Switzerland, 2018; pp. 277–288. [CrossRef]
17. Anceaume, E.; Guellier, A.; Ludinard, R.; Sericola, B. Sycomore: A permission less distributed ledger that self–adapts to transactions demand. In Proceedings of the 2018 IEEE 17th International Symposium on Network Computing and Applications (NCA), Cambridge, MA, USA, 1–3 November 2018; pp. 1–8. [CrossRef]
18. Hughes, A.; Park, A.; Kietzmann, J.; Archer-Brown, C. Beyond bitcoin: What blockchain and distributed ledger technologies mean for firms. *Bus. Horiz.* **2019**, *62*, 273–281. [CrossRef]
19. Rizal, B.F.; Ubacht, J.; Janssen, M. Unraveling transparency and accountability in blockchain. In Proceedings of the 20th Annual International Conference on Digital Government Research, Dubai, United Arab Emirates, 18–20 June 2019; pp. 204–213. [CrossRef]
20. Karame, G. On the security and scalability of bitcoin's blockchain. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; pp. 1861–1862. [CrossRef]
21. Hofmann, F.; Wurster, S.; Ron, E.; Böhmecke–Schwafert, M. The immutability concept of blockchains and benefits of early standardization. In Proceedings of the 2017 ITU Kaleidoscope: Challenges for Data-Driven Society (ITU K), Nanjing, China, 27–29 November 2017; pp. 1–8. [CrossRef]
22. Landerreche, E.; Stevens, M. On the immutability of blockchains. In Proceedings of the 1st ERCIM Blockchain Workshop 2018, Amsterdam, The Netherlands, 8–9 May 2018; European Society for Socially Embedded Technologies (EUSSET): Zurich, Switzerland, 2018. Available online: https://dl.eusset.eu/handle/20.500.12015/3160 (accessed on 25 October 2021).
23. Aste, T.; Tasca, P.; Di Matteo, T. Blockchain technologies: The foreseeable impact on society and industry. *Computer* **2017**, *50*, 18–28. [CrossRef]
24. Kosba, A.; Miller, A.; Shi, E.; Wen, Z.; Papamanthou, C. Hawk: The blockchain model of cryptography and privacy—Preserving smart vontracts. In Proceedings of the 2016 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22–26 May 2016; pp. 839–858. [CrossRef]
25. Buterin, V. Next–generation smart contract and decentralized application platform. *White Pap.* **2014**, *3*, 1–36. Available online: https://www.ethereum.org/ (accessed on 25 October 2021).
26. Bach, L.; Mihaljević, 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 (MIPRO), Opatija, Croatia, 21–25 May 2018; pp. 1545–1550. [CrossRef]
27. SCOOP. Blockchain and the Internet of Things: The IoT Blockchain Opportunity and Challenge. Available online: https://www.iscoop.eu/blockchain-distributed-ledger-technology/blockchain-iot/ (accessed on 25 October 2021).
28. WikiQ. Blockchain vs DAG.; Website—Updated during the Day. Available online: https://wikiq.ru/blockchain-vs-dag/ (accessed on 25 October 2021).
29. Netecon. Proof of Activity: Extending Bitcoin's Proof of Work via Proof of Stake// Israeli Bitcoin Association. Website—Updated during the Day. Available online: https://www.semanticscholar.org/paper/Proof-of-Activity%3A-Extending-Bitcoin%27s-Proof-of-via-Bentov-Lee/6f40066eb122c06dbff24241ba203e87b3d3c4ff (accessed on 25 October 2021).
30. Saad, A. Decentralized directed DLT network based on acyclic graphs. In Proceedings of the COINS'19: Proceedings of the International Conference on Omni-Layer Intelligent Systems, Crete, Greece, 5 May 2019; pp. 1–3. [CrossRef]
31. Ruscoins. Overview of the Radix DLT ICO Project. Website—Updated during the Day. Available online: https://ruscoins.info/ico/obzor-proekta-radix-ltd (accessed on 25 October 2021).
32. Ferdous, M.S.; Morshed Chowdhury, M.J.; Hoque, M.A. A survey of consensus algorithms in public blockchain systems for crypto-currencies. *J. Netw. Comput. Appl.* **2021**, *182*, 103035. [CrossRef]