sustainability

MDPI

Article

Anti-Counterfeiting and Traceability Consensus Algorithm Based On Weightage to Contributors in a Food Supply Chain of Industry 4.0

Ji Tan [1], S. B. Goyal [1,*], Anand Singh Rajawat [2], Tony Jan [3], Neda Azizi [4] and Mukesh Prasad [5,*]

[1] Faculty of Information Technology, City University, Petaling Jaya 46100, Malaysia
[2] School of Computer Sciences & Engineering, Sandip University, Nashik 422213, India
[3] Centre for Artificial Intelligence Research and Optimization, Design and Creative Technology Vertical, Torrens University, Sydney 2007, Australia
[4] School of Information Systems, Torrens University, Sydney 2007, Australia
[5] School of Computer Science, Faculty of Engineering and IT (FEIT), University of Technology Sydney, Sydney 2007, Australia
* Correspondence: sb.goyal@city.edu.my (S.B.G.); mukesh.prasad@uts.edu.au (M.P.)

Abstract: Supply chain management can significantly benefit from contemporary technologies. Among these technologies, blockchain is considered suitable for anti-counterfeiting and traceability applications due to its openness, decentralization, anonymity, and other characteristics. This article introduces different types of blockchains and standard algorithms used in blockchain technology and discusses their advantages and disadvantages. To improve the work efficiency of anti-counterfeiting traceability systems in supply chains and reduce their energy consumption, this paper proposes a model based on the practical Byzantine fault tolerance (PBFT) algorithm of alliance chains. This model uses a credit evaluation system to select the primary node and integrates the weightage to contributors (WtC) algorithm based on the consensus mechanism. This model can reduce the decline in the algorithm success rate while increasing the number of malicious transaction nodes, thereby reducing the computing cost. Additionally, the throughput of the algorithmic system increases rapidly, reaching approximately 680 transactions per second (TPS) in about 120 min after the malicious nodes are eliminated. The throughput rapidly increases as the blacklist mechanism reduces the number of malicious nodes, which improves the system's fault tolerance. To validate the effectiveness of the proposed model, a case study was conducted using data from the anti-counterfeiting traceability system of the real-life supply chain of a food company. The analysis results show that after a period of stable operation of the WtCPBFT algorithm in the proposed model, the overall communication cost of the system was reduced, the throughput and stability were improved, and the fault-tolerant performance of the system was improved. In conclusion, this paper presents a novel model that utilizes the PBFT algorithm of alliance chains and the WtC algorithm to improve the efficiency and security of anti-counterfeiting traceability systems in supply chains. The results of the case study indicate that this model can effectively reduce communication costs, improve throughput and stability, and enhance the fault tolerance of the system.

Keywords: consensus algorithm; food supply chain; blockchain anti-counterfeiting

Citation: Tan, J.; Goyal, S.B.; Singh Rajawat, A.; Jan T.; Azizi, N.; Prasad, M. Anti-Counterfeiting and Traceability Consensus Algorithm Based On Weightage to Contributors in a Food Supply Chain of Industry 4.0. *Sustainability* **2023**, *15*, 7855. https://doi.org/10.3390/su15107855

Academic Editors: Wenyan Song and Hao Li

Received: 2 March 2023
Revised: 5 May 2023
Accepted: 9 May 2023
Published: 11 May 2023

1. Introduction

Since its birth in 2009, blockchain technology has attracted increasing attention [1]. With time, the application of blockchain technology in various industries has become increasingly widespread [2]. As more companies and governments begin to explore blockchain technology, it has become the foundation of a new economic and social model [3]. With the continuous development of blockchain technology, it is no longer limited to digital currency [4]. It has been widely used in financial services, public services, social life, and

other fields [5]. Blockchain technology is expected to reconstruct various industries' business models and operational methods, promoting information transparency and efficiency improvement [6]. After Germany proposed IR4.0 in 2013, the manufacturing industry changed to digitalized, intelligent, fast, and effective personalized industry [7]. Industrial manufacturing automation, data flow automation, and economic operation automation are the characteristics of IR4.0 [8]. These features require blockchain as the underlying supporting technology [9]. Because blockchain has the characteristics of decentralization, openness, autonomy, invariance, and anonymity, these characteristics are also important in the anti-counterfeiting traceability field in supply chains [10].

In existing research, adding blockchain technology to a supply chain system can perfectly realize the decentralization of data storage in the process of anti-counterfeiting traceability. However, owing to the restriction of the consensus algorithm, the increase in data volume and the growth of the number of nodes cause the consensus process to be slower, resulting in higher delays; thus, storage and query speed are reduced. Therefore, consensus algorithms and system efficiency have gathered major research interest. Xiong et al. [11] studied, analyzed, and compared consensus algorithms in blockchain and summarized the characteristics, performance, and applicable scenarios of consensus algorithms. Xu et al. [12] proposed the Score Grouping-PBFT (SG-PBFT) algorithm; the traditional Byzantine algorithm has been studied and improved, and the consistency process has been optimized, finally changing and improving the grouping efficiency [12]. Mazzoni et al. [13] evaluated the quorum blockchain and explored the actual scalability and applicability of its consensus algorithm. On this basis, they defined a method that can be extended to any licensed blockchain technology. Zhang et al. [14] systematically investigated the different research directions of the BFT algorithm. Because the BFT algorithm has advantages in both enterprise and industrial production, the author comprehensively compared these algorithms in terms of message delivery and time complexity and then demonstrated the advantages and disadvantages of different algorithms. Although each research study has its advantages, it is difficult to consider each algorithm in terms of throughput, delay, power consumption, etc., and improving the efficiency of the consensus algorithm is the most important task. The efficiency of the consensus algorithm affects the query speed and the overall operation of the blockchain system, which requires further research.

To improve the efficiency of anti-counterfeiting traceability queries in supply chain systems and reduce energy consumption, after studying the consensus algorithm and the PBFT algorithm in alliance chains, a credit evaluation system was used to select the primary node and integrate the weightage to contributors (WtC) algorithm based on the PBFT consensus mechanism. Thus, the algorithms were improved. WtC is a hybrid consensus mechanism that improves the system efficiency and reduces system consumption by recording the measures with which block producers successfully publish blocks and reward them to ensure the security and efficiency of the long-term operating environment of the system. In terms of verification, all block producers can easily reach consensus on the different difficulty coefficients of different addresses using historical block records [15–18]. In addition, the consensus algorithm proposes a blacklist mechanism. Each block producer generates a consensus blacklist locally to punish the dishonest behavior of some block producers. Finally, a case study was conducted using data on an anti-counterfeiting traceability system in an existing supply chain of a food company [19–21], and the results are here discussed. This study aimed to improve the efficiency of anti-counterfeiting traceability systems in blockchain-based food supply chains by utilizing improved consensus algorithms and exploring energy-saving methods.

In this study, Section 2 studies and compares different types of blockchain and consensus algorithms. Section 3 proposes an improved PBFT algorithm. This algorithm adds a weightage to contributors mechanism based on the original PBFT. It selects primary nodes based on the contribution of block producers, forming a new WtCPBFT algorithm. On this basis, a blacklist mechanism is added to reduce source data fraud. Section 4 reports a case study and data analysis. It was found that the WtCPBFT algorithm in the model,

after a period of stable operation, reduced the overall communication cost of the system, improved throughput and stability, and improved the fault tolerance performance of the system. Section 5 presents conclusions on the comprehensive study.

2. Background

2.1. Blockchain Types

There are three types of blockchain: public, private, and consortium chains [22–25]. In public chain applications, such as Bitcoin, all information is completely open and transparent, and everyone can participate and complete transactions. All actions performed in a public chain are public, but in a public chain, this being public is not controlled by anyone and is not exclusively owned by any unit or organization [26]. This is a "completely decentralized" blockchain. Alliance chains are semi-public. In an alliance chain, multiple organizations or institutions jointly manage the blockchain. In an alliance chain, each institution runs one or more nodes. The data in this chain can only be read, written, and traded by the institutions within the system. The participants jointly manage the transaction data [27]. Although shared in public, it is not open to everyone; therefore, it is a partially decentralized blockchain. A private chain is a completely closed blockchain model. In a private chain, the writing authority is controlled by an organization or institution [28]. Resource allocation is strictly limited. Table 1 compares the characteristics of the three blockchains.

Table 1. Comparison of public vs. alliance vs. private blochchains.

Type	Public Chain	Alliance Chain	Private Chain
Participant	Anyone	Alliance member	Members of an institution or organization
Degree of decentralization	Decentralized	Polycentric	Centralized
Node joining method	Free to join	Internal controls	Authorization required
Excitation mechanism	Need	Optional	Not required
Consensus mechanism	PoW/PoS/DPoS	PBFT	RAFT
Transaction throughput	3∼20 TPS	1000∼10,000 TPS	1000∼100,000 TPS
Transaction speed	Slow	Fast	Quick
Application scenarios	Virtual currency	Payment and settlement	Audit and issuance

As shown in Table 1, in terms of technology selection, public chains have the highest degree of openness. However, due to the limitation of their transaction throughput, the transaction speed is slow, and it is not suitable for large-scale anti-counterfeiting and traceability applications. Private chains are more suitable for individual and internal use within organizations and cannot assure inter-company trust relationships. Supply chains generally include raw-material supply, processing, transportation, storage, sales, consumers, and regulators. The different participating roles in the same product supply chain are interconnected to form a similar organization. In an alliance system, the data in the system are maintained by each participant. Therefore, the transaction speed is comparatively faster in supply chain systems, and a multi-centralized alliance chain is a useful model in supply chain management.

2.2. Consensus Algorithm

A blockchain can be regarded as a distributed public ledger that records all transactions. All participants in the blockchain can own this ledger. The historical data of the public ledger cannot be tampered with. New data can only be added to the back of the blockchain. There is a problem, because each node has the same authority, which is responsible for writing new blocks to the ledger; then, a consensus algorithm is needed to solve the consistency problem of decentralized nodes.

A blockchain system combines a consensus mechanism so that the ledger of each node can be consistent with the ledgers of the nodes in other networks [29–31]. However, this kind of situation is almost impossible in a real-world centralized server, because the

transaction parties always have to maintain a trust-based relationship with the participating third-party institutions. Therefore, the consensus algorithm is the foundation and core of blockchain technology [32]. If a blockchain does not have a consensus algorithm mechanism, then it is no different from an ordinary database. The consensus algorithm solves the problem of mutual trust among nodes based on decentralization. Efficiently reaching consensus in distributed systems is an important research issue in the field of distributed computing. The more nodes in the blockchain network there are, the higher the degree of decentralization is, the smaller the decision-making power of the nodes is, and the lower the efficiency of the system in reaching a consensus is.

In the early stage, each participant holds relatively consistent computing power, and miners obtain bookkeeping rights and rewards with probability. Because of the existence of zero-sum games, miners usually choose to join mining pools to obtain higher and more stable returns. At present, mainstream mining pools use various mechanisms, such as PPLNS (Pay Per Last N Share), PPS (Pay Per Share), SOLO, and other mechanisms to distribute income to miners (check ref) [33]. However, with an increase in miners, the mining efficiency of each miner continues to decrease. The most intuitive solution to improve mining efficiency is to increase the computing power of miners. Therefore, some studies have designed more efficient and powerful computing units for Bitcoin mining. Bitcoin mining rigs have been upgraded from CPUs to graphics processors and from GPUs to application-specific integrated circuits (ASICs) [34]. Compared with ordinary computers, the mining efficiency of miners using ASICs is greatly improved; thus, an increasing number of users are investing in expensive Bitcoin mining equipment. However, this solution consumes more energy. Data from Blockchain.info show that from 2009 to 2018, the overall computing power in the Bitcoin network increased, reaching a maximum of 6×10^7 TH/S [35].

In addition, many scholars have conducted research on the energy consumption and efficiency of blockchain mining. In China, research on consensus algorithms began late. At the beginning of 2016, the technology company VeChain released a PoW algorithm-based NFC anti-counterfeiting chip and mobile terminal applications. This is China's first authentic identity anti-counterfeiting identification and transparent supply chain management platform based on blockchain technology. It can track products and verify them using interfaces. The authenticity of the situation ensures the transparency of product information [36], and large Internet companies, such as Huawei, Ali, and Tencent, have also carried out their research on blockchain application projects, which has boosted Chinese scholars' research on blockchain. In October 2016, Walmart announced a partnership with IBM and Tsinghua University to use blockchain technology to change participants in the food supply chain [17,37–42]. The largest blockchain application project to date has won unanimous praise from the industry worldwide [43]. Scholars worldwide have also researched the combination scheme of blockchain and anti-counterfeiting traceability. For example, Henry and others used the form of smart contracts to express and perform source tracking with the analysis of traceability products, realizing product traceability on the blockchain [44]; Abeyratne et al. [45] analyzed some problems in supply chain management and looked forward to the future of blockchain in supply chain management; Abadin et al. [46] used a PoW algorithm, and a new consensus algorithm model was proposed to help complete supply chain management. The algorithms proposed by these solutions can generally be divided into the three categories below.

2.2.1. Competency-Based Proof Algorithm

This kind of algorithm is also called the "Proof of X" series consensus algorithm. The idea is to replace the computing power on which the PoW relies with other capabilities. The typical proof of luck (POL) algorithm relies on a trusted hardware environment and obtains billing rights randomly [47]. This algorithm can solve the problems of high energy consumption and low mining efficiency. However, the downside is that all users who participate must deeply trust the manufacturer of parts, which is contrary to the decentralized

characteristics of blockchain. Another consensus algorithm, proof of burning (PoB), relies on the ability to bind tokens to a specific address; the greater the number of tokens bound is, the greater the probability of obtaining billing rights is. The problem with this algorithm is that it is not conducive to the circulation of tokens, that is, all users tend to bind a certain number of tokens to a specific address instead of using them for transactions [48]. There is also a PoSP consensus algorithm, which mainly depends on the storage capacity of users. A large amount of storage space consumption also introduces a series of new problems [49].

2.2.2. Hybrid Consensus Algorithm

The main idea of this type of algorithm is to absorb the advantages of different consensuses and appropriately use the PoW in some links to reduce energy consumption. Peercoin was the first project to use the proof of stake (PoS). Peercoin used a miner's right to hold a specific amount of currency as equity to mint new coins. Equity in the PoS is the coin age, also known as the coin day, which is the product of the miner's coin holding amount and holding time. If a miner's coin age is successfully used to mint new coins, the system resets the coin age to 0. The consensus based on the PoS has the problem of "protein at stake" at the time of the fork; that is, when miners vote in the face of the fork, no matter which fork they choose, they do not need to pay the cost, so they usually adopt the all-selection strategy to maximize their benefits [50].

2.2.3. Common Algorithms in Alliance Chains

Alliance chains have the characteristics of reasonable privacy, high TPS, and low energy consumption, because most of their consensus algorithms can effectively solve some shortcomings of the PoW, such as high energy consumption. However, this type of algorithm requires that the participating nodes only join the network with permission; therefore, not all nodes can equally and freely join the network, and the application scenarios of the algorithm are limited. Typical algorithms include RAFT, DPoS, and PBFT [51,52]. This kind of consensus algorithm solves the problem of high energy consumption from another perspective, but it is not suitable for public chain applications.

Table 2 shows the comparison of the above consensus algorithms. It can be seen that the PBFT algorithm can solve the Byzantine fault tolerance problem at a faster speed and with higher efficiency under the premise of multi-centralization, which is a good choice for the field of anti-counterfeiting traceability in supply chains. Therefore, the PBFT consensus algorithm was selected as the underlying optimization algorithm in this study.

Table 2. Comparison of various consensus algorithm mechanisms.

Type	PoW	PoS	DPoS	RAFT	PBFT
Scenes	Public	Public/Alliance	Alliance	Private/Alliance	Alliance
Mode	Decentralized	Decentralized	Decentralized	Centralized	Polycentric
Bookkeeping node	Whole network	Whole network	Election of rep.	Selection of leader	Dynamic decision
Response time	10 min	1 min	3 s	Second order	Second order
Storage efficiency	Full ledger	Full ledger	Full ledger	Full ledger	Full account book + partial account book
Throughput	About 7 TPS	About 15 TPS	About 300 TPS	Thousands or ten thousand transactions per second	About 1000 TPS or higher
Fault-tolerant	50%	50%	50%	50%	33%
Byzantine fault tolerance	Yes	Yes	No	No	Yes

3. Consensus Algorithm Based on Contributor Weight Proof

This paper proposes an optimized consensus mechanism, hereinafter referred to as practical Byzantine fault tolerance mechanism based on weight to contributors, referred to as WtC-PBFT (weightage to contributors practical Byzantine fault tolerance). Using the global credit value to determine the copy of the direct primary node and participate in the consensus, we adopt the PBFT algorithm integrating contributor weights to reduce network overhead and improve system efficiency. With the analysis of the primary node's behavior and its global credit values, the detection mechanism of Byzantine nodes is introduced, downgrading the Byzantine nodes to increase the maximum number of malicious nodes that the system can tolerate.

3.1. Primary Node Selection Method and New Node Protection Mechanism

There are only two roles in the PBFT algorithm: primary and replica nodes. These two roles can be converted into one another. The view number and the set of node numbers determine the primary node. At the beginning of the algorithm, the primary node is defined by Equation (1).

$$p = v \mod n \tag{1}$$

where p is the primary node number, v is the view number, and n is the number of nodes. As can be seen from the formula, with an increase in v, p changes constantly, and the primary node is currently chosen by the rotation holding system, whose result is predictable and almost loses the significance of decentralization.

In this study, the global credit value of a node is considered an important criterion for selecting a primary node using the global credit model. To ensure that all the primary nodes are online, a status identifier is set for each node. Then, according to Formula (2),

$$C_w = \frac{G_c \times states}{t} \tag{2}$$

$$V = v \times C_w \tag{3}$$

where C_w (credit weightage) is the node credit value weight, G_c (global credit) is the node global credit value, and *states* is the state of the node. In the normal state, the state value of the node is 1. When the node fails to operate normally because of a hang-up, attack, self-failure, network problem, or offline state, the state is 0. When the node is blacklisted, the state value is -1. t is the existence time parameter of the node in the system, and the existence of t is the protection of the new node C_w value. The credit value weight of each node is obtained with the following formula: The top 100 digits are selected to form the primary node set according to the credit value weight from high to low. Each node is the same as the other nodes. The top 50 digits are selected to form the preliminary primary node set. The number of centralized nodes can be determined dynamically using random numbers. Equation (3) combines v with the node credit value weight to obtain an updated v V value and then modifies it.

When the primary node set is used for block generation and verification, the preliminary primary node set must change dynamically, because the state of the node may change. As the transaction progresses, node evaluation and feedback messages are continuously updated, and these evaluation and feedback messages are required to have recent validity to update the t-value to ensure that the global credit value of the node changes dynamically. In this way, the problem of inconsistent primary node selection is solved. The disadvantage of changing or selecting the primary node in a particular order in conventional PBFT algorithms is also avoided.

3.2. Weightage to Contributors

In WtC, the number of complete releases of participating blocks is used to measure the contribution of participating nodes. The possibility of Byzantine nodes appearing in the PBFT consensus algorithm changes based on the contribution. The global contribution weight is recorded as D, which is jointly determined by the successful times of all nodes participating in the release of blocks and is updated every 2016 blocks. Different nodes participate in different block publishing successful times, and the contribution weight corresponding to node a_i is denoted as D^c. Because the accumulation of contributions is based on the number of times a node participates in successfully publishing blocks and the calculation of the contribution weight is based on node changes, the primary node method can calculate different contribution weights according to the number of successful nodes used. The nodes that participate more in the successful release of blocks increase the credit value, and the probability of becoming the primary node increases. The specific transformation relationship between the number of successes and contribution weight is shown in Equation (4).

$$\beta = \begin{cases} \frac{1}{r+e^{-(c-\mu)/\gamma,}}, & c \geq 1 \\ 1, & c = 0 \end{cases} \tag{4}$$

$$D^c = D \cdot (\beta) \tag{5}$$

where β denotes the contribution weight index. Equation (4) is a piecewise function designed based on a logistic function. It includes the transformation relationship between the weight index of the contribution and the number of successes. With the accumulation of success times, its growth rate gradually slows down and finally converges toward the upper bound.

Additionally, parameter R determines the unique upper bound of the function. By adjusting parameter R, the maximum weight of the contribution obtained by each node can be changed, and the maximum weight is $1/r$. The system can dynamically change the reward upper bound by adjusting R, which is equivalent to the switch of the contribution weight in the WtC algorithm. In special cases, when $r = 0$, WtC becomes a consensus algorithm based on virtual equity, which does not require the calculation of the contribution. When $r = 1$, WtC completely depends on its contribution to the consensus. In addition, parameter μ in Equation (4) represents the mean value and determines the position of the center of the curve in Figure 1. To ensure that the value was greater than 1, $\mu = -9$ was set in the experiment. The size of γ determines the growth rate near the curve mean. The smaller the value of γ is, the faster the convergence rate is. Setting $\gamma = 10$ ensured that each address converged toward the upper bound after successfully participating in publishing blocks approximately 50 times.

The current contribution weight DC corresponding to each node can be obtained using formula [5], where C represents the number of times each node participates in successfully publishing blocks. It is easy to see from Equation (5) that the greater the contribution weight of the node is, the greater the contribution weight is. When the primary node is selected again to publish new blocks, the probability that a node with a successful rating is selected as the primary node is higher than that of the other nodes, and it has a higher probability of obtaining credit value rewards. It is worth noting that because the function is convergent, there is an upper limit to the probability reward for a node, and there are no infinite rewards for a node. This is also a protection for the newly added nodes. With the increase in participating nodes and the continuous issuance of new credit value awards, the impact is smaller and smaller. It can be seen that for nodes with only zero or a small number of success times, the protocol is friendly, and there is still a chance to obtain credit value rewards. The credit value generated by the reward can effectively improve the efficiency of primary node selection and reduce the actual energy consumption.

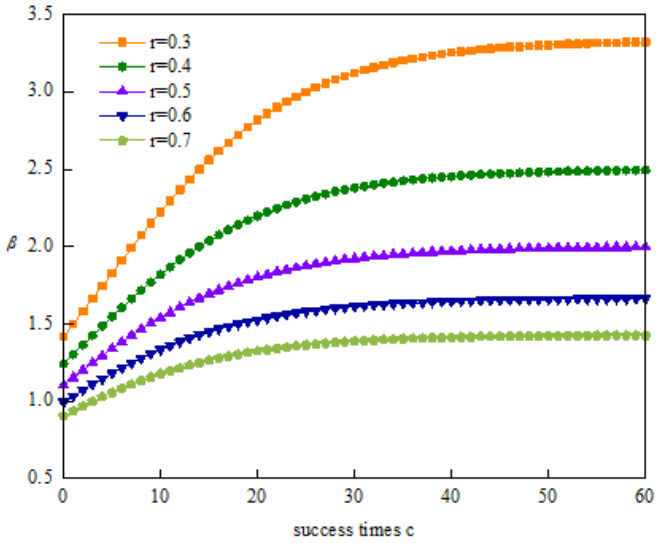

Figure 1. Variation relationship between success times c and difficulty index for a given r value.

3.3. WtCPBFT Consensus Algorithm

3.3.1. Concept of the Algorithm

To ensure the activity and security of the PBFT consensus algorithm, when the total number of nodes is $3f + 1$, the maximum number of Byzantine nodes that it can tolerate is f. In this algorithm, the consensus protocol ensures that each normal node executes the client's request messages in the same order; when the primary node has a system error or becomes a Byzantine node, the primary node is replaced by the view replacement protocol, so that the client requests executed by the normal node are not tampered with; the checkpoint protocol is used to clear log records, set watermark values (h and H), and synchronize the node state.

To ensure the real-time effectiveness of the system, the WtCPBFT consensus algorithm must update the primary node set after some time. If a malicious node is found in the primary node set within this time interval, the system assigns a lower global reputation value and contribution to the malicious node and eliminates it. In the primary node set, the next node in the prepared primary node set replaces the node that participates in the consensus. With the operation of the system, the more frequent the transactions of normal nodes are, the higher the global reputation value is, and the higher the node contribution is; all types of malicious nodes can be effectively identified by the global trust model. Given a lower global reputation value, the system begins a virtuous circle, and the probability of malicious nodes being selected as primary is very low. The number of Byzantine nodes that can be tolerated in the consensus algorithm remains unchanged, but the possibility of malicious nodes being selected as primary nodes is reduced, so the number of malicious nodes that the entire system can tolerate increases dynamically.

3.3.2. Algorithm Flow

The algorithm includes four steps: primary node selection, replica contribution weight verification, Byzantine node judgment, information verification, and information feedback. The specific process is reported below (Step 3 includes two steps: information verification and feedback).

Step (1): Primary node selection.

Assuming that there are n nodes in total, the nodes are numbered from 0 to $n-1$; the same state information of all nodes is called a view, and the views are numbered

incrementally from 0 simultaneously. A primary node is required in the view, and the primary node is selected using Equation (6):

$$p = (v + h) \mod n \tag{6}$$

where p represents the node number, h represents the current consensus block height, v represents the credit value-weighted view number, and the remaining primary nodes are called slave nodes (replica) to select the new primary node.

Step (2): Replica node contribution weight judgment.

The contribution weights of all the replica nodes are obtained using Equations (4) and (5), and they are sorted. A larger contribution weight indicates that it has more contributions and is more likely to be a non-Byzantine node. After the system has run for a long time, a contribution weight threshold is obtained, and a node with a weight higher than this threshold can be considered a non-Byzantine node.

Step (3a): Specific steps for WtCPBFT consensus without Byzantine nodes in the primary node.

(a) The primary node packs and sorts after receiving the client request message and then broadcasts the pre-preparation message to the replica node.

(b) The node executes the request and returns the result to the client. If the client receives the information fed back by the $3f + 1$ nodes and the information is the same, the consensus is successful.

(c) If the client does not receive feedback information from all nodes or there is different feedback information, it means that the consensus has failed, and there are Byzantine nodes in the primary node; they go to the consensus protocol consensus process to reach a consensus.

The WtCPBFT consensus process adopts a new consensus process, as shown in Figure 2.

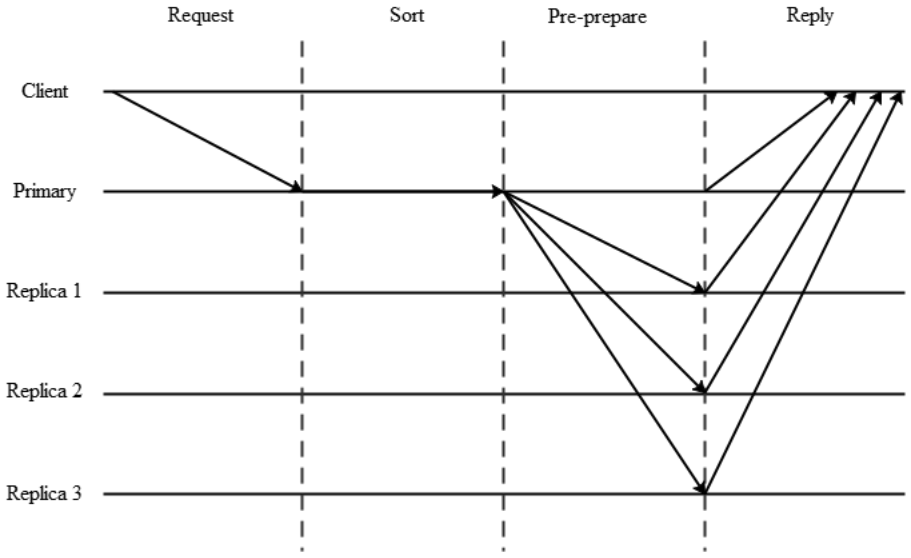

Figure 2. Consensus flow chart without Byzantine nodes.

Step (3b): Specific steps of the Byzantine node WtCPBFT consensus in the primary node.

(a) The transaction information is sent by the client to the master node.

(b) The primary node packages the message, checks its validity, deletes illegal transaction information, and assigns a serial number. Then, it sends a message to the replica node that we call pre-prepare; its structure is <<pre-prepare,v(s,l),n,d>m>. In this structure, pre-prepare refers to preparing the protocol phase of the current message in advance. v (s, l) is the view structure, where v represents the view number, s represents the current node status, l is the blacklist, n is the increment sequence number representing the unique increment of the master node, d is the summary format, and m is the message content.

(c) The message passing the verification mark is that the replica node receives the information sent by pre-prepare (verification includes the view number, message sequence number, digest, and signature). The algorithm enters the third stage, namely, the preparation stage, and sends the prepared message, <Prepare, v (s), n, d, i>.

(d) After the node receives 2f + 1 (including the node itself) preparation messages, legitimacy verification is performed, and the log is written. The preparation message must be written in the log, and the preparation message can be selected and recorded. The writing on the log shows the following: The preparation phase is complete. The confirmation message, <Commit, v(s), n, D(m), i>, is sent to enter the committing stage.

(e) The node receives 2f + 1 (including the node itself) confirmation messages, representing a consensus; then, the node executes the request, writes the data, and finally feeds back the information to the client.

The WtCPBFT consensus process was based on the traditional consensus protocol process, as shown in Figure 3. With the operation of the system, the global reputation value of the nodes becomes increasingly accurate, and the possibility of Byzantine nodes in the primary node is greatly reduced. Therefore, the system adopts the algorithm process of (3a) for a long time to achieve consensus and maintain a virtuous circle, thereby reducing the probability of consensus. This results in response delay, increased system throughput, reduced computing power, and reduced power consumption.

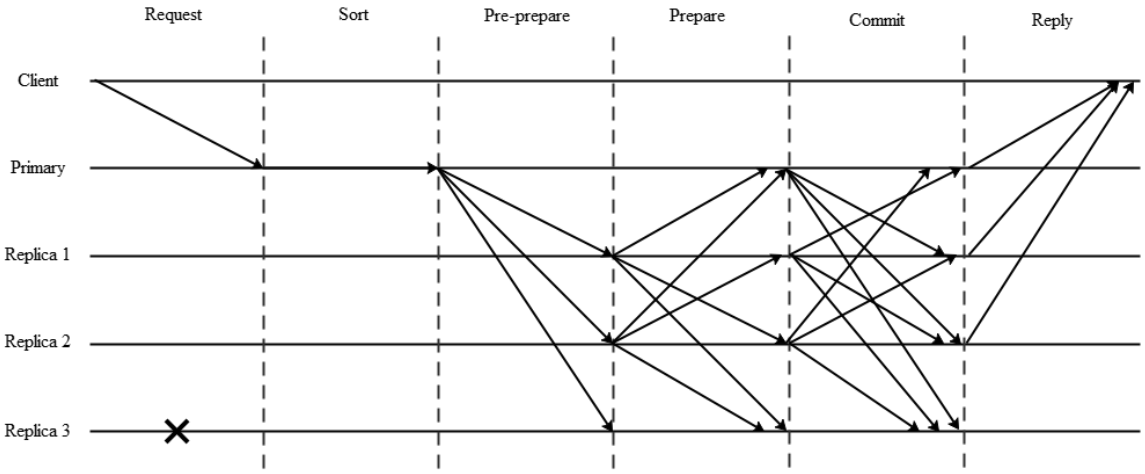

Figure 3. Consensus flow chart with Byzantine nodes (with × denoting a Byzantine node).

3.3.3. Blacklist Mechanism

During the period when the primary node set is not updated, the primary node may experience errors such as system failure, network delay, or malicious attacks; therefore, the node is called a Byzantine node. Therefore, when the consensus stage is completed, if the node fails to send feedback messages to the client or if the feedback messages are inconsistent, an error is considered to have occurred. At this time, the algorithm reduces its global credit value, immediately removes the primary node set, and adds it to the blacklist. At the same time, a node in the set of reserve nodes participates in the consensus. This further reduces the possibility of Byzantine nodes in the primary node and improves system efficiency.

The blacklist is part of the WtCPBFT consensus mechanism that stores Byzantine nodes in the message structure of <<pre-prepare,v(s,l),n,d>m>. When any node detects malicious attack behavior in the network and either the credit value or the contribution degree used by the attacker is low, the attacking node can be recorded in blacklist l, and its state value can be modified to -1. The nodes in the blacklist cannot be used as the primary nodes for generating new transactions, and the accumulated number of successfully published participating blocks is zero, which is meaningless. If a node finds that the primary node appears on the blacklist when verifying a new block, it can directly discard the block.

Because there are many attack behaviors and forms in blockchain, the detection methods for each attack are different, and there is no unified paradigm to describe them. Therefore, how is each attack detected? How can we confirm this with the other nodes? These issues require further research and analysis. This study only focuses on how blockchain is used to maintain blacklists without changing the existing data structure.

Figure 4 shows the flowchart of the complete consensus algorithm of the system.

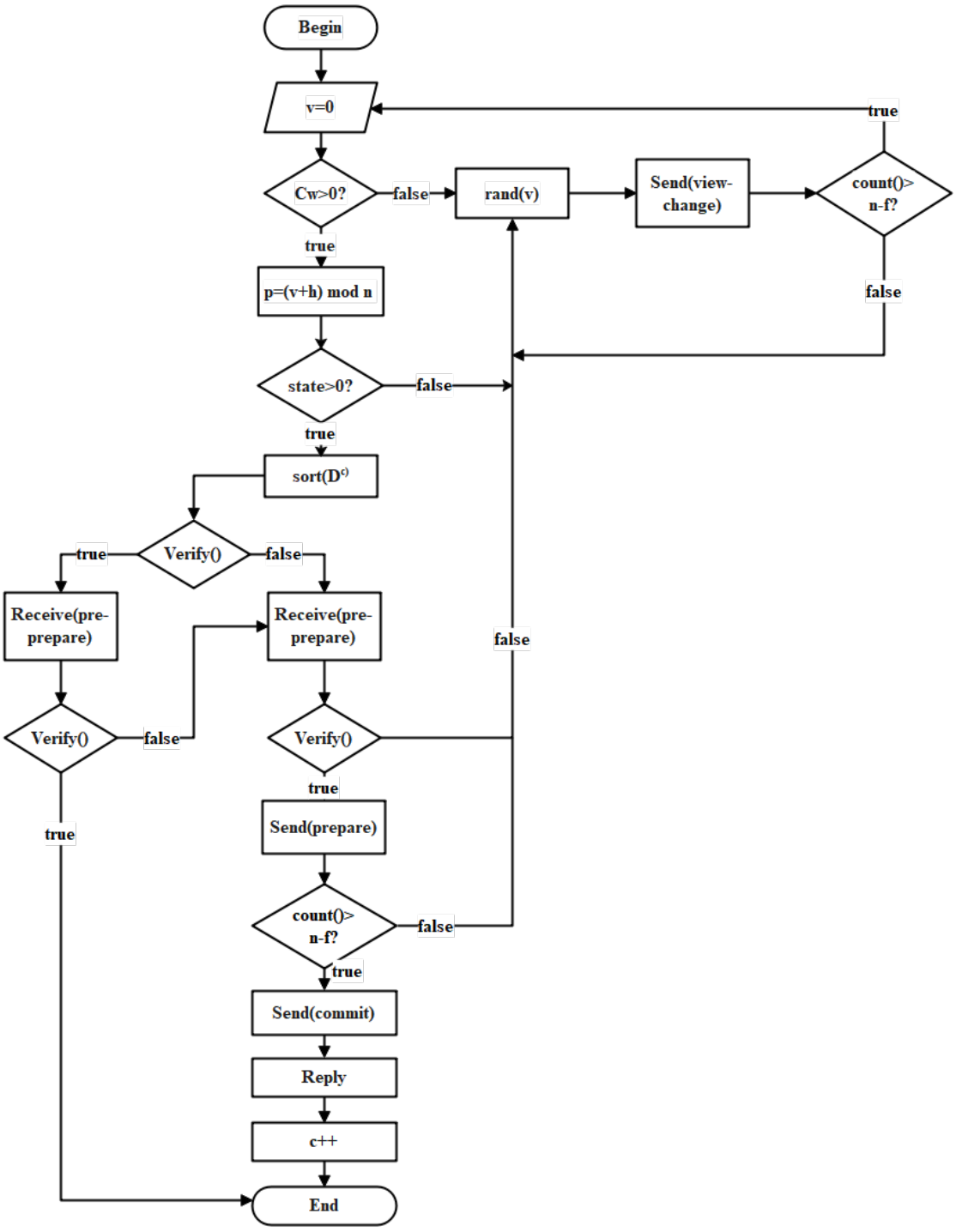

Figure 4. Flow chart of the algorithm.

4. Case Study and Analysis of Results

Based on PBFT, the WtCPBFT algorithm proposed in this study uses the global credit value to select consensus nodes, improves consensus efficiency, and enhances system fault tolerance by adjusting the static consensus node mechanism in PBFT to a dynamic add-on, elimination, and blocklist mechanism, which is suitable for private chain systems and large-scale consortium chain systems. The comparative analysis of WtCPBFT and PBFT in the three aspects of computing power overhead, transaction throughput, and fault tolerance performance is reported below.

4.1. Algorithm Data Collection

The algorithm data of both PBFT and WtCPBFT are from simulation experiments. The design idea for the experimental process comes from a company in China called Xiangnian Food Co., Ltd. The enterprise has achieved complete industrial chain control of food safety traceability from the field to the dining table by controlling every step of wheat cultivation, acquisition, storage, processing, and sales. The anti-counterfeiting and tracing platform adopted by enterprises can achieve data collection, monitoring, intelligent warning, and other functions throughout the entire industry chain. The underlying algorithm applies an alliance chain architecture based on the PBFT algorithm. We researched the system, simulated its processes in a natural operating environment, and collected data.

We conducted data collection, training, and testing on two laptops and a desktop computer. Owing to the limitations of the algorithm communication protocol, the three computers were located on the same intranet. These two laptops had Windows 10 operating systems. One was computer A, for algorithm implementation, and the other was computer B, with a poor configuration, namely, Lenovo Saver 15-ISK (CPU model: Intel Core i56,300HQ; memory: 8 GB; Windows 10 HomeBasic 64-bit system). The desktop CPU was an Intel Xeon E3 1231V3 processor with 16 GB memory, and the system was Windows 10 HomeBasic 64 bits. Other software environments were consistent with computer algorithms.

To better reflect the entire operating environment of the supply chain, each of the three computers enabled 5 nodes, of which laptop A enabled the client, and the remaining 14 were replica nodes. Owing to the use of personal computers, the operating environment of a supply chain could only be simulated manually. Therefore, the below three methods were adopted for data collection.

(1) After collecting 200 operation feedbacks, complete regular operation.

(2) During operation, manually turn off the N0 node on laptop A, the N2 node on laptop B, and the N3 node on desktop C, which are offline nodes, and collect 200 operation feedbacks.

(3) Close five nodes on notebook B, and collect 200 operation feedbacks.

Owing to code limitations, the nodes only had online and offline statuses and could not simulate other complex operating environments. Because the internal network environment of the experimental network was relatively stable, there were no large-scale network fluctuation, disconnection, etc.; however, the data obtained fluctuated within a reasonable range owing to unexpected packet loss of the router and other reasons. To ensure the stability and validity of the test data, the above three rules were used to conduct three groups of the same tests at different times in the same network environment and to average the results. At the same time, the PBFT algorithm was simulated in the same environment and operation mode. The running data results of the two algorithms were compared and analyzed. The data collection results are presented in Table 3.

NSTP denotes the number of successful transactions in the PBFT test/200 transactions.

Table 3. Comparison of various consensus algorithm mechanisms. NSTP denotes the number of successful transactions in the PBFT test/200 transactions.

Number of Nodes		No. of Nodes Shut Down	Closing of Nodes A-N0, NB-N2, and C-N3	Shutdown of 5 Nodes
NSTP	Test 1	198	181	153
	Test 2	197	182	151
	Test 3	200	187	157
NSTP	Test 1	181	159	149
	Test 2	185	157	144
	Test 3	184	155	140
NSTP	Test 1	197	195	187
	Test 2	200	194	187
	Test 3	196	193	188
NSTP	Test 1	80	67	59
	Test 2	87	69	61
	Test 3	89	70	56

4.2. Computing Power Cost

In the PBFT consensus algorithm, there are two processes of preparing and committing for all network nodes to communicate with each other, and the communication cost is high, which limits the applicability of the consensus in public chain and large-scale alliance chain systems. Compared with PBFT, WtCPBFT greatly reduces the process of two-to-two communication interactions; thus, its communication cost is greatly reduced.

Because WtCPBFT refers to the global credit model, the communication cost increases, but its complexity is far lower than that of PBFT. At the same time, WtCPBFT combines traditional consensus protocols with blacklist technology, and malicious nodes with low global credit values participate in consensus. This probability is very low. Byzantine nodes were detected using the blacklist mechanism proposed in this study. If the system runs stably for a period, the Byzantine nodes in the system may completely disappear. Therefore, in most consensus processes, the system uses the Byzantine protocol based on blacklist technology for block generation and verification.

Next, the global credit model was used to solve the probability of malicious nodes in the selected consensus nodes. The successful transaction rate (STR) was introduced into the performance evaluation of the model, that is, the ratio of the number of successful transactions to the total number of transactions in the system. Here, it can be considered that one node selects another node to conduct a transaction. If the transaction node is normal, the transaction is successful; otherwise, the transaction fails. It can be considered that the successful transaction rate also represents the probability that the node selects a normal node. Malicious nodes include simple malicious nodes (SMS), dishonest recommended nodes (SMR), coordinated malicious nodes (CM), and strategic malicious nodes (SMP). Because an SMP node can evaluate according to a specific situation to avoid being identified by the model and provide a lower credit value, its impact on the system is greater than that of other types of malicious nodes. Assuming that all malicious nodes in the system were SMP nodes, the change in the STR with SMP nodes is shown in Figure 5.

The information in Figure 5 clearly shows that when the proportion of SMP nodes was 40%, the STR was still greater than 80% (approximately 81.3%). With the Byzantine detection mechanism, when the number of malicious nodes in the system appropriately increased, it did not affect the system. However, when the proportion of SMP nodes increased to 55%, the STR sharply dropped. When the proportion of SMP nodes reached approximately 60%, the STR decreased to 0.

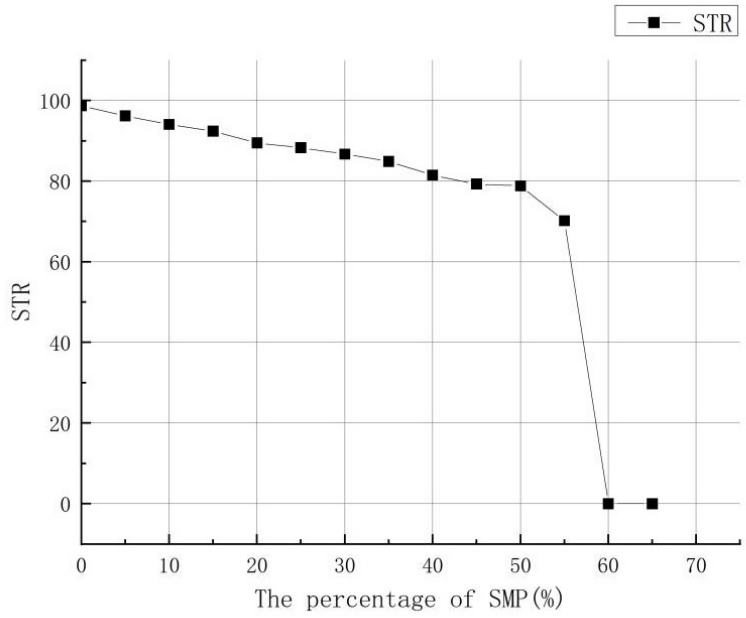

Figure 5. STR changes with SMP nodes.

4.3. Transaction Throughput

Transaction throughput is the standard for operational efficiency. The WtCPBFT algorithm can improve throughput and reduce transaction latency, and the probability of Byzantine nodes appearing in WtCPBFT gradually decreases with the time of system operation. This is mainly because the blacklist mechanism is used, which can also greatly reduce the delay. Therefore, at the system level, the computational cost decreases with a reduction in transaction delay; the processing time is shortened; and the overall throughput is improved. This is optimized compared with the relatively stable mechanism of PBFT.

Figure 6 shows that in the early stage of system operation, because the WtCPBFT algorithm increased the Sort process compared with the PBFT algorithm, the throughput was low, approximately 350 TPS. It took a certain time from the start of the operation to the stability of the system, and owing to the existence of the blacklist mechanism, the throughput of the WtCPBFT algorithm system rapidly increased after eliminating malicious nodes, reaching approximately 680 TPS in approximately 120 min. However, because the PBFT algorithm has no blacklist mechanism, the throughput was stable between 350 TPS and 380 TPS.

4.4. Fault Tolerance Performance

The performance limit of the PBFT consensus algorithm is mainly due to the proportion of Byzantine nodes. The requirements for Byzantine nodes are no more than 1/3 of the entire network. The fault tolerance rate of the WtCPBFT is more than 33%. Due to the existence of the blacklist mechanism, with the long-term operation of the system, the number of Byzantine nodes in all nodes is greatly reduced or even eliminated, and the global credit value of malicious nodes becomes lower and lower, so the role of malicious nodes in the selection of the primary node is minimal. At the same time, the number of malicious nodes that can be tolerated by WtCPBFT increases as the system enters a virtuous cycle, and transaction throughput detection can indirectly reflect the impact on system performance when the malicious nodes are appropriately increased.

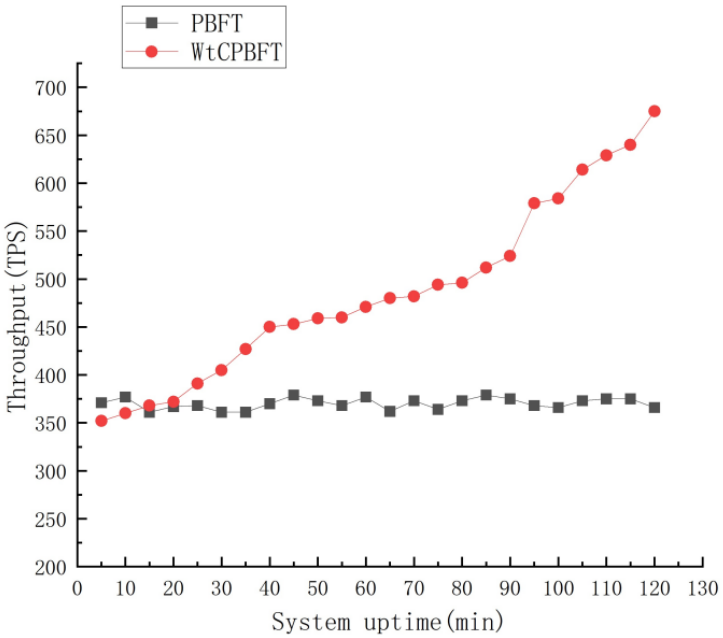

Figure 6. Comparison of PBFT and WtCPBFT throughput with the system running time.

Figure 7 shows that the blacklist mechanism gradually decreased the proportion of malicious nodes in the system with the system running time, and the system throughput gradually increased owing to its influence. The above analysis and comparison demonstrate that the WtCPBFT consensus mechanism has lower communication overhead, higher transaction throughput, and fault tolerance under the same non-essential factors.

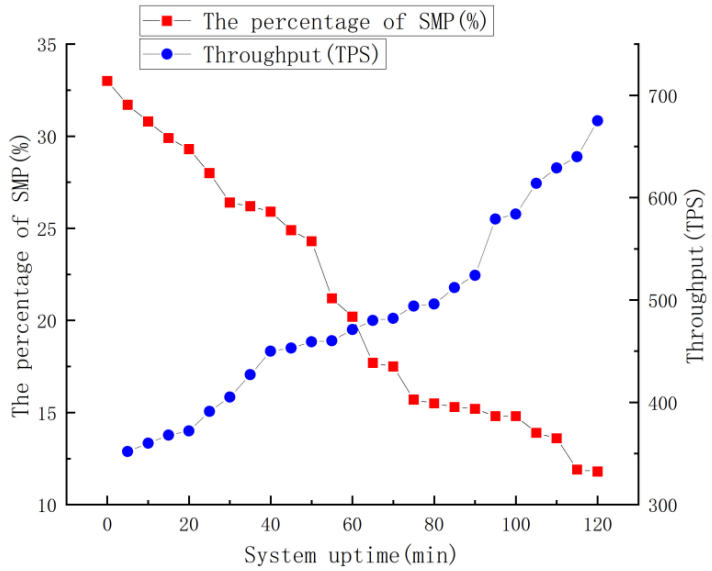

Figure 7. Relationship between the proportion of malicious nodes in WtCPBFT and throughput.

4.5. Correlation of Proposed Research in Food Supply Chain

The proposed research and case study already establish a relationship between food supply chain and blockchain technology by highlighting the benefits of using blockchain for anti-counterfeiting and traceability applications in supply chains. In the Case Study section, the research work provides information on how the proposed WtCPBFT algorithm was implemented and used in the anti-counterfeiting traceability system of a real-life rice

company. This research work also includes information on the specific parameters and messages used in the food industry for the proposed model, such as data related to the origin, quality, and safety of food products. Additionally, the proposed research discusses how the proposed model can be used to track the movement of food products along the supply chain and ensure that they meet certain standards and regulations.

Overall, Table 4 provides a summary of the main points covered in Section 3 of the research paper and how they relate to the food supply chain. It also helps to highlight the practical application of the proposed WtCPBFT algorithm in the real-life example of the rice company and how it can improve the work efficiency of anti-counterfeiting traceability in food supply chains.

Table 4. Proposed methodology of the research paper relationship to the food supply chain real-life example.

Section 3—Consensus Algorithm Based on Contributor Weight Proof	Relationship to Food Supply Chain	Real-Life Example
Section 3.1—Primary node Selection Method and New Node Protection Mechanism	Selecting primary node to ensure authenticity and traceability	Selection of primary node in the anti-counterfeiting traceability of rice supply chain
Section 3.2—Weightage to contributors	Assigning weight to different contributors in the supply chain	Assigning weight to farmers, manufacturers, distributors, and retailers in the chain
Section 3.3.1—Concept of the Algorithm	Incorporating credit evaluation system to select the primary node	Credit value generated by rewards for successful release of blocks
Section 3.3.2—Algorithm Flow	Optimizing the consensus process for the food supply chain	Reducing computing cost, improving stability and fault tolerance
Section 3.3.3—Blacklist Mechanism	Punishing malicious behavior and increasing normal node transactions	Generating a consensus blacklist locally to identify malicious nodes

5. Conclusions

Blockchain technology has a demand for applications in the anti-counterfeiting traceability of food supply chains. Research shows that the number of data and the number of nodes affect the efficiency of the consensus process, including system delay, storage, and query speed. This study combines the PBFT algorithm idea, uses the credit evaluation system to select the primary node, forms the WtC mechanism, improves the PBFT algorithm, and forms a new WtCPBFT algorithm. In addition, the consensus algorithm proposes a blacklist mechanism. Each block producer generates a consensus blacklist locally to punish the dishonest behavior of some block producers.

Using the credit evaluation system to select the primary node solves the drawback of primary node selection in the PBFT algorithm remaining unchanged or in a certain order. The concept of WtC is introduced and added to the PBFT algorithm system. By simulating and collecting data on the workflow of a real anti-counterfeiting traceability platform for a food company and comparing the PBFT algorithm and the improved WtCPBFT algorithm in the system, it can be concluded that WtC can assist the original PBFT algorithm in distinguishing between Byzantine nodes and non-Byzantine nodes. At the same time, the credit value generated by the rewards for nodes participating in the successful release of blocks can effectively improve the efficiency of selecting the primary node and reduce the actual computing cost. The addition of the blacklist mechanism increases the frequency of normal node transactions in the system, and the higher the global credit value is, the higher the node contribution is. All types of malicious nodes can be effectively identified by the global credit model and given a lower global credit value. The system starts a virtuous cycle, and the possibility of malicious nodes being selected as primary nodes is very low. The tolerance to Byzantine nodes of the consensus algorithm is increased. Because the probability of the Byzantine node being selected as the primary node decreases dynamically,

the number of malicious nodes that the whole system can tolerate increases. The WtCPBFT algorithm can provide a practical application solution for improving the work efficiency of anti-counterfeiting traceability in food supply chains, efficiently selecting primary nodes, and punishing malicious nodes.

Author Contributions: Conceptualization, Writing—original draft, J.T.; Supervision, S.B.G. Reviewing and editing, A.S.R., T.J. and N.A.; Validation, J.T.; Presenting of new approach or methodology, J.T. and S.B.G.; Formal analysis and Investigation J.T.; Resources, S.B.G. and T.J.; Software, J.T. and S.B.G.; Writing—Review and Editing, J.T., T.J., S.B.G., N.A. and M.P. All authors have read and agreed to the published version of the manuscript.

Funding: There was no funding available for this research.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not Applicable.

Data Availability Statement: Data will be made available upon appropriate request from the corresponding author.

Acknowledgments: Thanks to Xiangnian Food Co., Ltd., who provided experimental support. The enterprise has achieved a complete industrial chain control of food safety traceability from the field to the dining table by controlling every step of wheat cultivation, acquisition, storage, processing, and sales.

Conflicts of Interest: There is no personal nor financial conflict of interest for this research.

Abbreviations

The following abbreviations are used in this manuscript:

PBFT	Practical Byzantine fault tolerance
WtC	Weightage to contributors
TPS	Transactions per second
WtCPBFT	Weight to contributors practical Byzantine fault tolerance
IR4.0	Industrial Revolution 4.0
SG-PBFT	Score Grouping-PBFT
PPLNS	Pay Per Last N Share
PPS	Pay Per Share
ASICs	Application-specific integrated circuits
CPU	Central Processing Unit
GPU	Graphics Processing Unit
PoW	Proof of work
PoS	Proof of stake
DPoS	Delegated proof of stake
IBFT	Istanbul Byzantine fault tolerance
POL	Proof of luck
PoB	Proof of burning
NSTP	Number of successful transactions in the PBFT test/200 transactions
STR	Successful transaction rate
SMS	Simple malicious nodes
SMR	Dishonest recommended nodes
CM	Coordinated malicious nodes
SMP	Strategic malicious nodes

References

1. Rehman Khan, S.A.; Yu, Z.; Sarwat, S.; Godil, D.I.; Amin, S.; Shujaat, S. The role of block chain technology in circular economy practices to improve organisational performance. *Int. J. Logist. Res. Appl.* **2022**, *25*, 605–622. [CrossRef]
2. Hunt, K.; Narayanan, A.; Zhuang, J. Blockchain in humanitarian operations management: A review of research and practice. *Socio-Econ. Plan. Sci.* **2022**, *80*, 101175. [CrossRef]
3. Shokri, A.; Shokri, A.; White, D.; Gelski, R.; Goldberg, Y.; Harrison, S.; Rashidi, T.H. EnviroCoin: A Holistic, Blockchain Empowered, Consensus-Based Carbon Saving Unit Ecosystem. *Sustainability* **2022**, *14*, 6979. [CrossRef]

4. Moudoud, H.; Cherkaoui, S.; Khoukhi, L. An IoT blockchain architecture using oracles and smart contracts: The use-case of a food supply chain. In Proceedings of the 2019 IEEE 30th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Istanbul, Turkey, 8–11 September 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–6.
5. Hajiaghayi, M.T.; Kowalski, D.R.; Olkowski, J. Improved communication complexity of fault-tolerant consensus. In Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, 20–24 June 2022; pp. 488–501.
6. Trinh, M.H.; Van Vu, D.; Van Tran, Q.; Ahn, H.S. Matrix-Scaled Consensus. In Proceedings of the 2022 IEEE 61st Conference on Decision and Control (CDC), Cancún, Mexico, 6–9 December 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 346–351.
7. Korkmaz, K.; Bruneau-Queyreix, J.; Mokhtar, S.B.; Réveillère, L. ALDER: Unlocking blockchain performance by multiplexing consensus protocols. In Proceedings of the 2022 IEEE 21st International Symposium on Network Computing and Applications (NCA), Boston, MA, USA, 14–16 December 2022; IEEE: Piscataway, NJ, USA, 2022; Volume 21, pp. 9–18.
8. Manolache, M.A.; Manolache, S.; Tapus, N. Decision making using the blockchain proof of authority consensus. *Procedia Comput. Sci.* **2022**, *199*, 580–588. [CrossRef]
9. Rajawat, A.S.; Goyal, S.B.; Bedi, P.; Simoff, S.; Jan, T.; Prasad, M. Smart Scalable ML-Blockchain Framework for Large-Scale Clinical Information Sharing. *Appl. Sci.* **2022**, *12*, 795. [CrossRef]
10. Yang, K.; Li, C.; Jing, X.; Zhu, Z.; Wang, Y.; Ma, H.; Zhang, Y. Energy dispatch optimization of islanded multi-microgrids based on symbiotic organisms search and improved multi-agent consensus algorithm. *Energy* **2022**, *239*, 122105. [CrossRef]
11. Xiong, H.; Chen, M.; Wu, C.; Zhao, Y.; Yi, W. Research on progress of blockchain consensus algorithm: A review on recent progress of blockchain consensus algorithms. *Future Internet* **2022**, *14*, 47. [CrossRef]
12. Xu, G.; Bai, H.; Xing, J.; Luo, T.; Xiong, N.N.; Cheng, X.; Liu, S.; Zheng, X. SG-PBFT: A secure and highly efficient distributed blockchain PBFT consensus algorithm for intelligent Internet of vehicles. *J. Parallel Distrib. Comput.* **2022**, *164*, 1–11. [CrossRef]
13. Mazzoni, M.; Corradi, A.; Di Nicola, V. Performance evaluation of permissioned blockchains for financial applications: The ConsenSys Quorum case study. *Blockchain Res. Appl.* **2022**, *3*, 100026. [CrossRef]
14. Zhang, G.; Pan, F.; Dang'ana, M.; Mao, Y.; Motepalli, S.; Zhang, S.; Jacobsen, H.A. Reaching consensus in the byzantine empire: A comprehensive review of bft consensus algorithms. *arXiv* **2022**, arXiv:2204.03181.
15. Xu, X.; Wang, C.; Zhou, P. GVRP considered oil-gas recovery in refined oil distribution: From an environmental perspective. *Int. J. Prod. Econ.* **2021**, *235*, 108078. [CrossRef]
16. Lv, Z.; Chen, D.; Lou, R.; Song, H. Industrial Security Solution for Virtual Reality. *IEEE Internet Things J.* **2021**, *8*, 6273–6281. [CrossRef]
17. Wang, X.; Feng, H.; Chen, T.; Zhao, S.; Zhang, J.; Zhang, X. Gas sensor technologies and mathematical modelling for quality sensing in fruit and vegetable cold chains: A review. *Trends Food Sci. Technol.* **2021**, *110*, 483–492. [CrossRef]
18. Yu, Y.; Liu, A.; Dhawan, G.; Mei, H.; Zhang, W.; Izawa, K.; Soloshonok, V.A.; Han, J. Fluorine-containing pharmaceuticals approved by the FDA in 2020: Synthesis and biological activity. *Chin. Chem. Lett.* **2021**, *32*, 3342–3354. [CrossRef]
19. Li, Q.K.; Lin, H.; Tan, X.; Du, S. H∞ Consensus for Multiagent-Based Supply Chain Systems Under Switching Topology and Uncertain Demands. *IEEE Trans. Syst. Man. Cybern. Syst.* **2020**, *50*, 4905–4918. [CrossRef]
20. Xu, J.; Yang, Z.; Wang, Z.; Li, J.; Zhang, X. Flexible sensing enabled packaging performance optimization system (FS-PPOS) for lamb loss reduction control in E-commerce supply chain. *Food Control* **2023**, *145*, 109394. [CrossRef]
21. Unhelkar, B.; Joshi, S.; Sharma, M.; Prakash, S.; Mani, A.K.; Prasad, M. Enhancing supply chain performance using RFID technology and decision support systems in the industry 4.0–A systematic literature review. *Int. J. Inf. Manag. Data Insights* **2022**, *2*, 100084. [CrossRef]
22. Wang, C.; Tan, X.; Yao, C.; Gu, F.; Shi, F.; Cao, H. Trusted Blockchain-Driven IoT Security Consensus Mechanism. *Sustainability* **2022**, *14*, 5200. [CrossRef]
23. Zhao, C.; Zhang, S.; Wang, T.; Liew, S.C. Bodyless Block Propagation: TPS Fully Scalable Blockchain with Pre-Validation. *arXiv* **2022**, arXiv:2204.08769.
24. Li, Z.; Wang, W.; Guo, J.; Zhu, Y.; Han, L.; Wu, Q. Blockchain-Empowered Dynamic Spectrum Management for Space-Air-Ground Integrated Network. *Chin. J. Electron.* **2022**, *31*, 456–466. [CrossRef]
25. Tellew, J.; Kuo, T.T. CertificateChain: Decentralized healthcare training certificate management system using blockchain and smart contracts. *JAMIA Open* **2022**, *5*, ooac019. [CrossRef] [PubMed]
26. Yang, L.; Zou, Y.; Xu, M.; Xu, Y.; Yu, D.; Cheng, X. Distributed consensus for blockchains in internet-of-things networks. *Tsinghua Sci. Technol.* **2022**, *27*, 817–831. [CrossRef]
27. Chen, X.; Zhao, S.; Qi, J.; Jiang, J.; Song, H.; Wang, C.; On Li, T.; Hubert Chan, T.; Zhang, F.; Luo, X.; et al. Efficient and DoS-resistant consensus for permissioned blockchains. *ACM Sigmetrics Perform. Eval. Rev.* **2022**, *49*, 61–62. [CrossRef]
28. Yang, W.; Garg, S.; Huang, Z.; Kang, B. A hybrid consensus algorithm for master–slave blockchain in a multidomain conversation system. *Expert Syst. Appl.* **2022**, *204*, 117300. [CrossRef]
29. Jain, A.; Arora, S.; Damle, S.; Gujar, S. Tiramisu: Layering consensus protocols for scalable and secure blockchains. In Proceedings of the 2022 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Shanghai, China, 2–5 May 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–3.
30. Nguyen, D.C.; Hosseinalipour, S.; Love, D.J.; Pathirana, P.N.; Brinton, C.G. Latency optimization for blockchain-empowered federated learning in multi-server edge computing. *IEEE J. Sel. Areas Commun.* **2022**, *40*, 3373–3390. [CrossRef]

31. Ma, X.; Wu, H.; Xu, D.; Wolter, K. CBlockSim: A Modular High-Performance Blockchain Simulator. In Proceedings of the 2022 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), Shanghai, China, 2–5 May 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–5.
32. Keddar, M.; Doumbia, M.L.; Belmokhtar, K.; Krachai, M.D. Enhanced reactive power sharing and voltage restoration based on adaptive virtual impedance and consensus algorithm. *Energies* **2022**, *15*, 3480. [CrossRef]
33. Tritt, A.; Abda, I.N.; Dahdah, N. Review of MIS-C Clinical Protocols and Diagnostic Pathways: Towards a Consensus Algorithm. *CJC Pediatr. Congenit. Heart Dis.* **2022**, *1*, 86–93. [CrossRef]
34. Wang, Q.; Li, R.; Wang, Q.; Chen, S.; Xiang, Y. Exploring unfairness on proof of authority: Order manipulation attacks and remedies. In Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security, New York, NY, USA, 30 May–3 July 2022; pp. 123–137.
35. Wang, H.; Tan, W.; Wu, J.; Liu, P. OPBFT: Optimized Practical Byzantine Fault Tolerant Consensus Mechanism Model. *AI and Analytics for Public Health: Proceedings of the 2020 INFORMS International Conference on Service Science*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 123–135.
36. Gu, S.; Pan, W.; Chung, T.; Huang, X. Blockchain-based model for intelligent supply chain production and distribution. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 7503017. [CrossRef]
37. Lu, B.; Guo, Z.; Zhong, K.; Osire, T.; Sun, Y.; Jiang, L. State of the art in CRISPR/Cas system-based signal conversion and amplification applied in the field of food analysis. *Trends Food Sci. Technol.* **2023**, *135*, 174–189. [CrossRef]
38. Yan, L.; Yin-He, S.; Qian, Y.; Zhi-Yu, S.; Chun-Zi, W.; Zi-Yun, L. Method of Reaching Consensus on Probability of Food Safety Based on the Integration of Finite Credible Data on Block Chain. *IEEE Access* **2021**, *9*, 123764–123776. [CrossRef]
39. Xu, J.; Ma, R.; Stankovski, S.; Liu, X.; Zhang, X. Intelligent Dynamic Quality Prediction of Chilled Chicken with Integrated IoT Flexible Sensing and Knowledge Rules Extraction. *Foods* **2022**, *11*, 836. [CrossRef] [PubMed]
40. Li, G.; Wang, J.; Li, D.; Liu, S.; Yin, J.; Lai, Z.; Yang, G. A Hg(II)-specific probe for imaging application in living systems and quantitative analysis in environmental/food samples. *Chin. Chem. Lett.* **2021**, *32*, 1527–1531. [CrossRef]
41. Joshi, S.; Sharma, M.; Ekren, B.Y.; Kazancoglu, Y.; Luthra, S.; Prasad, M. Assessing Supply Chain Innovations for Building Resilient Food Supply Chains: An Emerging Economy Perspective. *Sustainability* **2023**, *15*, 4924. [CrossRef]
42. Mishra, A.K.; Tripathy, A.K.; Obaidat, M.S.; Tan, Z.; Prasad, M.; Sadoun, B.; Puthal, D. A Chain Topology for Efficient Monitoring of Food Grain Storage using Smart Sensors. In Proceedings of the 15th International Joint Conference on e-Business and Telecommunications (ICETE 2018), Porto, Portugal, 26–28 July 2018; Volume 1, pp. 89–98.
43. Lee, H.; Yeon, C. Blockchain-based traceability for anti-counterfeit in cross-border e-commerce transactions. *Sustainability* **2021**, *13*, 11057. [CrossRef]
44. Kim, H.M.; Laskowski, M. Toward an ontology-driven blockchain design for supply-chain provenance. *Intell. Syst. Account. Financ. Manag.* **2018**, *25*, 18–27. [CrossRef]
45. Abeyratne, S.A.; Monfared, R.P. Blockchain ready manufacturing supply chain using distributed ledger. *Int. J. Res. Eng. Technol.* **2016**, *5*, 1–10.
46. ul Abadin, Z.; Syed, M. A Pattern for Proof of Work Consensus Algorithm in Blockchain. In Proceedings of the 26th European Conference on Pattern Languages of Programs, Graz, Austria, 7–11 July 2021; pp. 1–6.
47. Russell, P.; Brown, P.N. The Philos Trust Algorithm: Preventing Exploitation of Distributed Trust. In Proceedings of the 2022 IEEE International Conference on Blockchain (Blockchain), Espoo, Finland, 22–25 August 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 45–52.
48. Dua, K. Implementation of an efficient, portable and platform-agnostic cryptocurrency mining algorithm for Internet of Things devices. *arXiv* **2022**, arXiv:2205.01646.
49. Singh, A.; Kumar, G.; Saha, R.; Conti, M.; Alazab, M.; Thomas, R. A survey and taxonomy of consensus protocols for blockchains. *J. Syst. Archit.* **2022**, *127*, 102503. [CrossRef]
50. Wen, X.J.; Chen, Y.Z.; Fan, X.C.; Zhang, W.; Yi, Z.Z.; Fang, J.B. Blockchain consensus mechanism based on quantum zero-knowledge proof. *Opt. Laser Technol.* **2022**, *147*, 107693. [CrossRef]
51. Zheng, X.; Feng, W.; Huang, M.; Feng, S. Optimization of PBFT algorithm based on improved C4. 5. *Math. Probl. Eng.* **2021**, *2021*, 1–7.
52. Tian, J.; Hou, M.; Bian, H.; Li, J. Variable surrogate model-based particle swarm optimization for high-dimensional expensive problems. *Complex Intell. Syst.* **2022**. [CrossRef]