Hindawi
Security and Communication Networks
Volume 2023, Article ID 1525998, 16 pages
https://doi.org/10.1155/2023/1525998

Research Article

Enhanced Multiset Consensus Protocol Based on PBFT for Logistics Information Traceability

Linchao Zhang,[1] Lei Hang ,[2] and Dohyeun Kim [1]

[1]*Department of Computer Science, Jeju National University, Jeju, Republic of Korea*
[2]*Business School, Shanghai Normal University Tianhua College, Shanghai 201815, China*

Correspondence should be addressed to Dohyeun Kim; kimdh@jejunu.ac.kr

Received 26 September 2022; Revised 20 October 2022; Accepted 24 November 2022; Published 22 February 2023

Academic Editor: Y. P. Tsang

In the recent years, the global logistics industry has greatly driven the development of the world economy. At the same time, a large amount of data information is generated. Due to the frequent occurrence of logistics information leakage and forgery, it is necessary to find solutions that can accurately trace logistics information and ensure the security and authenticity of logistics information. The birth of blockchain technology has transformed the logistics industry from quantitative change to qualitative change. The technical characteristics of blockchain technology, such as distributed storage ideas, decentralization, immutability, and complex encryption consensus algorithm, endow it with a wide range of application prospects in the logistics industry. This paper proposes an enhanced multiset consensus algorithm based on PBFT (practical Byzantine fault tolerance) for logistics information traceability and storage on the logistics blockchain. The application of the proposed multi-set consensus algorithm in the topology structure composed of multiple sets can improve the consensus efficiency of logistics information in the blockchain. We improve consensus capability and transaction speed, avoid redundant consensus message packets occupying a large bandwidth, and efficiently process logistics information generated at any time. We ensure the traceability of logistics information and achieve efficient and accurate traceability, and the efficiency and security of the proposed algorithm are analyzed. This paper aims to solve the problems of traceability, trustworthiness, and efficient processing of blockchain applications in logistics information to operate the logistics network efficiently. This paper compares the proposed algorithm with the PBFT-related expansion algorithm regarding bandwidth occupation, delay, and throughput. The results show that the MPBFT consensus algorithm significantly improves the efficiency of the logistics blockchain network.

1. Introduction

With the rapid development of e-commerce, the logistics industry and various Internet technologies have become more in-depth, and the logistics industry has become more intelligent. However, many problems in the logistics industry, such as the authenticity of data information and logistics traceability, still need to be solved urgently. Currently, there is a lot of fake data in the logistics industry; at the same time, with the gradual maturity of big data technology, logistics data information has become increasingly important national confidential information. Due to the characteristics of the logistics industry involving multiple regions, it is very suitable for applying decentralized blockchain technology. The data in the blockchain are encrypted and stored in a distributed form and cannot be tampered with. Therefore, the logistics industry is exploring how to use blockchain technology better to protect logistics information's privacy, prevent information from being tampered with, and improve traceability.

This paper aims to study the application of blockchain technology in logistics information traceability scenarios. "Logistics information" refers to the corresponding data generated by each link of logistics circulation. To efficiently trace these data, this paper designs a logistics information traceability architecture based on a multiset consensus algorithm, which stores all logistics information indirectly on the blockchain to build a decentralized logistics blockchain.

According to the characteristics of logistics informatization, combined with the research status and application progress of blockchain technology in the logistics industry in domestic and foreign literature, a multiset consensus algorithm is proposed to improve the efficiency of logistics blockchain information traceability and information authenticity and introduced the blockchain logistics-related solutions, as shown in Table 1.

How to ensure the security of logistics data is a crucial issue. On the one hand, establishing a traceable system to monitor the information on items is ideal for both managers and buyers. On the other hand, managing and sharing logistics data are a challenging job. Handling large amounts of data efficiently and keeping it secure are critical. It is inappropriate and impractical to disclose all information about an item, as doing so may reveal the user's private information. It is also not advisable to rely entirely on information from the online world, as there may be some malicious attackers. Therefore, researchers must figure out the right way to manage logistics data. Traditional methods, including data sampling [4, 5], preprocessing [6], statistical analysis [7], and data mining [8], have insufficient security and low efficiency for big data. As mentioned above, as emerging application technology, blockchain is essentially a database with attractive properties such as anticounterfeiting, traceability, and transparency [9]. Based on these characteristics, using blockchain as a new foundation for logistics scenarios and its decentralization will benefit the industry's development. However, there are several key issues related to the physical world. One of the problems is that the network nodes on the blockchain are difficult to bind with physical entities, which we call the traceability problem. Generally speaking, only trusted nodes on the blockchain can be protected, and information fraud cannot be controlled at the level of a physical collection of information, so no matter how reliable the blockchain is, it cannot be used and managed. This article provides logistics traceability and data security protection at the information flow level. Previous studies combining blockchain and logistics scenarios mainly focused on designing different system levels, and a few focused on data reliability before investing in the blockchain. Other work is based on public key infrastructure (PKI) to achieve identity authentication [10]. For the supply of goods, the digital identity issued by the certificate authority (CA) is used to realize the binding of physical goods and network nodes on the blockchain. However, PKI is highly dependent on CAs responsible for the issues and management of private and public key pairs. It is essentially a centralized system and has the disadvantage of being centralized. On the other hand, even with the credentials of the data source and leveraging the blockchain to build the system, it still has problems due to the sheer volume of data. Logistics datasets are usually large scale in practical applications, so an efficient consensus algorithm is required to transmit all information to each node for users to query. This places great demands on the efficiency of the blockchain network.

The main content of this paper is to carry out an in-depth research expansion and feasibility study on the article [11] proposed earlier. Section 1 analyzes the problems in the logistics industry and the significance of logistics information traceability and expounds on the main research content of this paper. Section 2 introduces the background and related research of logistics information traceability studied in this paper. In addition, related consensus algorithms are also introduced. Section 3 is the logistics traceability system architecture based on a multiset consensus algorithm. Section 4 mainly introduces some improvements to the PBFT algorithm, proposes an efficient consensus algorithm for the logistics industry under MPBFT, avoids redundant consensus message packets, and proposes solutions to describe the basic model algorithm. The algorithm improves the PBFT consensus efficiency, reduces the network bandwidth of the logistics system, and further expounds the algorithm of the overall implementation process. In Section 5, the bandwidth test and performance analysis of the proposed MPBFT algorithm are carried out. Section 6 summarizes the work of this paper, puts forward some shortcomings, and puts forward some suggestions for the research and improvement of this paper.

2. Related Work

This section mainly studies the application of blockchain technology in logistics information traceability scenarios. Firstly, the background knowledge of logistics information traceability is briefly introduced. Then, the existing consensus algorithm-related technologies of blockchain are introduced, and the application of blockchain technology in logistics information traceability and the research status at home and abroad are analyzed.

2.1. Background. With the rapid development of the modern logistics industry, the logistics industry generates a large amount of data information, among which problems such as leakage of user cargo information have threatened people's life safety and should not be underestimated [12]. In real life, the user's cargo information can generally be divided into two parts, one is the user's private information and the other is the information generated by the logistics in the process of cargo transportation.

At present, the following security threats still exist in the traceability of logistics information:

(1) It is not easy to guarantee the confidentiality and authenticity of logistics information. The logistics industry generally only pays attention to protecting consumers' private information, such as name, phone number, and address, and does not take protective measures for the logistics information generated during the transportation process, which is prone to generate a large amount of false information.

(2) User authentication is difficult to achieve. In logistics and distribution, it is often sent to the receiving point. There is no shortage of malicious customers

Table 1: Comparison of logistics traceability solutions.

Literature	Important method	Solution
TPLI: A Traceable Privacy-Preserving Logistics Information Scheme via Blockchain [1]	Development prototype verification traceability	Implementing a prototype system using hyperledger
Blockchain-Based Supply Chain Traceability: Token Recipes Model Manufacturing Processes [2]	Proposing the token to calculate the gas cost	A linear relationship between gas cost and the product is proposed to define the relationship between ingredients and products
Research on Traceability Algorithm of Logistics Service Transaction Based on Blockchain [3]	Traceability algorithm of logistics service transaction	Through the proposed algorithm, the end-to-end traceability service of logistics service is realized

who might fake their identities to pick up packages that do not belong to them.

Faced with the above security threats, the research on logistics information traceability is highly significant. Solutions are urgently needed to ensure the traceability of logistics information's efficiency, authenticity, and security.

2.2. Consensus Algorithm. Reference [13] first proposed the problem of how nodes participating in transaction maintenance in distributed ledgers reach consensus. This problem mainly needs to be verified from the following three aspects: whether it has termination, whether it is consistent, whether it is valid, and whether it can. The method to achieve the above three aspects is called the consensus algorithm. The blockchain logistics traceability system and the consensus algorithm mechanism ensure that the information is decentralized and cannot be easily changed. Consensus refers to making most nodes (at least 51%) in the entire network trust that the data are reliable. At present, the mainstream consensus mechanism includes the proof of work (PoW), proof of interest (PoS), proof of entrusted Interest (DPOS), and Byzantine consensus (PBFT), as shown in Table 2.

2.2.1. Proof of Work. POW (proof of work) [14] is the consensus algorithm adopted in the Bitcoin system proposed by Nakamoto and Bitcoin. The consensus algorithm ensures the consistency of transaction records by introducing the method of computing power competition to make nodes reach a consensus. However, one of the biggest shortcomings of the POW algorithm is the serious waste of computing power.

2.2.2. Proof of Stake. A node holds many tokens, the node will want the currency to be stable, and the node will strive to maintain the system's normal operation and ensure the currency's stability. Therefore, the algorithm of accounting node selection to reach consensus can be designed by using the information related to the token held by the node, also known as the proof of stake algorithm (POS). The advantage of the POS consensus algorithm is that it does not need to consume computing resources. Still, the disadvantage is that the nodes holding the largest number of tokens tend to monopolize the nodes, which deviates from the core idea of blockchain.

2.2.3. Delegated Proof of Stake. The DPOS algorithm is similar to the board system. To avoid centralizing some nodes with excessive rights, the algorithm uses "witnesses" to supervise the behavior of nodes [15]. This algorithm is equivalent to partial decentralization, and the core lies in the representative select. The DPOS algorithm eliminates the time transaction need to wait for verification by untrusted nodes, thereby increasing consensus speed.

2.2.4. Practical Byzantine Fault Tolerance. The PBFT algorithm [16, 17] is divided into five stages: request, preparation, preparation, submission, and reply. PBFT algorithm has the consistency protocol needed to reach a consensus. It includes the checkpoint protocol used to restrain nodes and the view-switching protocol needed to switch views after discovering faulty nodes. In a consistency protocol, the system broadcasts messages to other secondary nodes through the primary node and reaches a consensus by returning the message consistency. The checkpoint protocol saves memory by periodically cleaning up contracted transaction data. The view switchover protocol is used to initiate a view switchover to reselect the primary node when the primary node fails. PBFT algorithm has high consensus efficiency and does not require much computing power maintenance.

2.3. Research Status of Consensus Algorithm and Application in Blockchain Traceability. The development of the logistics express industry should be reflected in the huge volume of parcels brought by the current e-commerce industry and should pay attention to the transformation of the logistics industry from quantitative change to qualitative change. The development of the worldwide logistics industry is increasingly moving towards a high-quality stage [18], in which blockchain technology is expected to endow the logistics industry with more intelligence and innovation by relying on its many unprecedented advantages. Many domestic and foreign researchers are discussing the application of blockchain technology in logistics information traceability scenarios, hoping to promote the logistics industry to another development climax.

Reference [19] uses side chain technology to design a supply chain traceability system. The blockchain platform used in this system is Ethereum. The literature focuses on analyzing the design of smart contracts, using smart contracts to manage goods and realizing logistics information's traceability. Reference [20] utilizes the Ethereum blockchain and smart contracts to efficiently execute commercial transactions for soybean tracking and traceability throughout the agricultural supply chain. The scheme utilizes smart contracts to manage and control all participants' interactive supply chain ecosystem transactions. Reference [21] developed a food traceability prototype system using blockchain technology for food safety issues. This document proposes a management architecture for on-chain and off-chain data through which traceability systems can alleviate the data explosion problem of IoT blockchains. Reference [22] proposed a blockchain-based drug traceability framework to prevent drug fraud, and logistics traceability was carried out from the start of drug production, including distribution.

Reference [23] applies blockchain to intelligent pallet management in logistics, proposes a Palletaa architecture, and studies the synergy of integrating alliance blockchain and IoT. A corresponding layered architecture is proposed to construct the system deployment in the industry. The position-inventory-routing problem of the pallet pool is formulated to efficiently manage pallet usage in the logistics industry. Reference [24] discusses the situation of BTCS

Table 2: Comparison of four consensus algorithms.

Consensus algorithm	POW	POS	DPOS	PBFT
Degree of centralization	Decentralization	Decentralization	Partially decentralized	Partially decentralized
Whether to resist Byzantine nodes	Y	Y	Y	Y
Is permission required	N	N	N	Y
System scale (number of nodes)	Unlimited	Unlimited	Unlimited	Limited
Node dynamic changes	Support	Support	Support	No support
Applicable scene	Public chain	Public chain	Public chain	Alliance chain
Consistency	Weak consistency	Weak consistency	Weak consistency	Consistency

from the perspective of industrial application and market competition. First, it expounds on the potential of BTCS and the gap between the ideal and reality of BCTS. Secondly, it discusses the applicability of BTSC, mainly through the game theory model, to the market. The situation was discussed. Reference [25] summarizes the application of blockchain in traceability. First, it analyzes the problems that need to be solved in the traceability scenario, then expounds on the related technologies of blockchain, and points out how these technologies can be applied to the traceability scenario. Literature [26–28] is based on the current social conditions where smartphones are popularized; the anticounterfeiting system is designed using smartphones as nodes of the blockchain. Users only need to install software on their mobile phones to achieve the purpose of traceability and anticounterfeiting.

The encrypted logistics information blockchain must be transmitted to the blockchain in the logistics information blockchain. Due to the decentralized nature of the blockchain, all transactions are negotiated by all nodes on the chain. “The consensus algorithm realizes the process.” When the transaction information is uploaded to the chain, the nodes complete the consensus of the transaction data. The data can be added to the blockchain database [29]. References [30–33] all use the consensus algorithm of blockchain technology to realize the traceability query of logistics information. The following focuses on analyzing the research status of the PBFT consensus algorithm.

The consensus problem is the core problem of the blockchain network. The PBFT consensus algorithm can accommodate both faulty nodes and malicious nodes. The number of all nodes must be at least $3f + 1$ (f is the number of malicious/faulty nonresponding nodes), which can ensure security and liveness in an asynchronous system. At the same time, it solves the problem that the original Byzantine fault tolerance (BFT) algorithm is inefficient, but PBFT has high latency in the case of network instability. Reference [34] considers many nodes in the blockchain when for the consensus speed problem, the consensus is carried out in a partially decentralized way. That is, a certain number of “central” nodes are selected. For the selection method of such nodes, the literature uses an improved K-medoids clustering algorithm. However, this method is still too expensive in the consensus process. The literature [35] combines the Gossip protocol to allow half of the nodes in the system to do evil. The blockchain system proposed in this literature has certain scalability, allowing foreign nodes to join the system anytime. However, this paper’s model for node selection is not accurate enough, and the time for the system to enter normal operation still needs to be shortened. Reference [36] considers the issue of client latency and proposes an EZ. But consensus algorithm there is no master node, but instead it enables each replica to order requests received from clients. Reference [37] applies blockchain technology to the logistics service supply chain information platform and uses the blockchain network’s DBFT (authorized Byzantine fault tolerance) consensus algorithm. In this consensus algorithm, not all nodes on the chain can reach a consensus. Instead, the nodes eligible for agreement are selected by voting on the nodes. The higher the stake a node has, the greater the probability of becoming a consensus node. The literature [38, 39] also focuses on the source of the product. In the blockchain traceability system, the NPBFT consensus algorithm is used, and the communication overhead required for consensus is reduced, thereby improving traceability efficiency.

It can be seen from the above that blockchain technology can be applied to the traceability requirements of logistics information in various scenarios. Among blockchain-related technologies, the consensus algorithm is an important technology. In the next sections, we will analyze the research status of consensus algorithm technologies in blockchain traceability and focus on exploring the research status of the PBFT consensus algorithm.

3. Design of Logistics Information Traceability System Based on Multiset Consensus Algorithm

Aiming at the proposed logistics information traceability problem, this section designs a logistics information traceability system based on a multiset consensus algorithm in combination with blockchain-related technologies. We carry out the overall design of the system. After analyzing the system’s requirements, the overall architecture of the system is created, and the system’s network is deployed.

3.1. Overall System Design. The design goal of the system in this paper is to solve the problem that the information in each link of the goods circulation is not updated in time, and the transportation process is difficult to trace. Due to the huge amount of logistics information data and real-time changes, an efficient consensus algorithm is needed to ensure that the data can be uploaded to the chain quickly and without errors. This system plans to use an improved PBFT consensus algorithm (MPBFT), which can reduce the time it takes for transactions to reach consensus, thereby improving consensus efficiency.

This paper uses the supply chain solution provided by IBM [40] to build a blockchain traceability system suitable for logistics. The supply chain solution provides basic logistics equipment management, storage and query and provides cloud services required for gateway service integration, which greatly improves the scalability of required functions. According to the IBM product business model [41], the proposed system is layered, and the logistics traceability system is jointly constructed using IBM-related services and the proposed MPBFT consensus algorithm. The overall architecture of this system is based on the five-layer architecture of the blockchain, adding a storage layer, adding a data analysis platform to the contract layer, and merging the network layer and the consensus layer. The system's overall architecture is shown in Figure 1, divided into the application, contract, storage, network, consensus, and data layer from top to bottom.

(1) Application layer: the application layer is the entrance to the realization of system functions and is mainly responsible for the interaction of system users, including the registration, login, and query operations of administrators and ordinary users.

(2) Contract layer: execute-related smart contracts in the application layer, such as sign-off and delivery reminders

(3) Storage layer: the layer is divided into off-chain and on-chain databases. The off-chain database is a traditional distributed database, mainly deployed in the cloud, and stores the hash data corresponding to the original data of logistics information. The on-chain database is mainly used to serve data information generated by nodes in the blockchain.

(4) Network and consensus layer: the logistics information data files are transmitted through the transmission network formed by 5G, NBIoT, and intelligent gateway, and the block data is sent to the P2P point-to-point network.

(5) Data layer: the data layer is divided into logistics information and block data. The logistics information data is actual transportation route data. Address information and form chain structure blocks through technologies such as digital signatures and Merkle trees.

According to the overall system architecture of Figure 1, the network structure of this system mainly includes cloud, blockchain network, and application. The system network deployment is shown in Figure 2. The raw data of logistics information is stored in a distributed database in the cloud. Different roles access through the MSP access control system of Hyperledger Fabric and access the application data in the blockchain network after CA authentication. Each user terminal can quickly and efficiently query the required information, and the Blockchain ensures the authenticity and validity of the data.

4. Multiset PBFT (MPBFT)

This paper uses a consensus algorithm of PBFT type in the logistics information traceability system based on blockchain. Since multiple nodes are involved in the logistics and transportation process, a large amount of logistics information will be generated, which requires the blockchain to have high performance and bandwidth requirements. Therefore, because of the low efficiency of blockchain logistics traceability and the pain point of the redundant large amount of data, this paper designs a multiset PBFT algorithm, which improves the consensus efficiency of the logistics system, improves the consensus efficiency of transaction information on the chain, and reduces the communication overhead of the system.

4.1. Basic Model of Algorithm. As shown in Figure 3, this paper defines each computer as a separate node. In the MPBFT algorithm, the nodes are divided into different node sets according to the architectural requirements (region, environment, communication), the node sets are connected, and the nodes in each node set adopt the PBFT algorithm consensus. The set of nodes relates to the consensus recorder. In this paper, it is assumed that each node set has four nodes. First, the consensus information is broadcast to the consensus recorder through the master node, and the consensus recorder will send a new consensus message to the nodes in each set. After all, nodes complete the consensus; each node will write the consensus data. After the data have passed the node consensus, the master node will verify and respond to the client, which proves that the transaction consensus is completed.

This paper applies the above consensus model to the blockchain logistics industry, and nodes are divided into different sets according to different geographical locations. Nodes with the same geographical location are divided into the same node set, and nodes in any node set can be used as master nodes to broadcast consensus. To reduce the number of communications, we avoid redundant consensus message packets. The MPBFT algorithm is used to verify the consensus message package between nodes, and there is no need to repeatedly forward the consensus message package across regions. Assume that M set networks are set up according to different cities in the logistics blockchain network N:

(1) Set the number of a set in the blockchain node network to M and denote the set as S_i, So$\{Si \in \{S, S_2, \ldots, S_M\}\}$

(2) Set the malicious nodes among the N nodes of the cluster as f, so the cluster N must satisfy the following:

$$N \geq \frac{3f + 1}{M}. \tag{1}$$

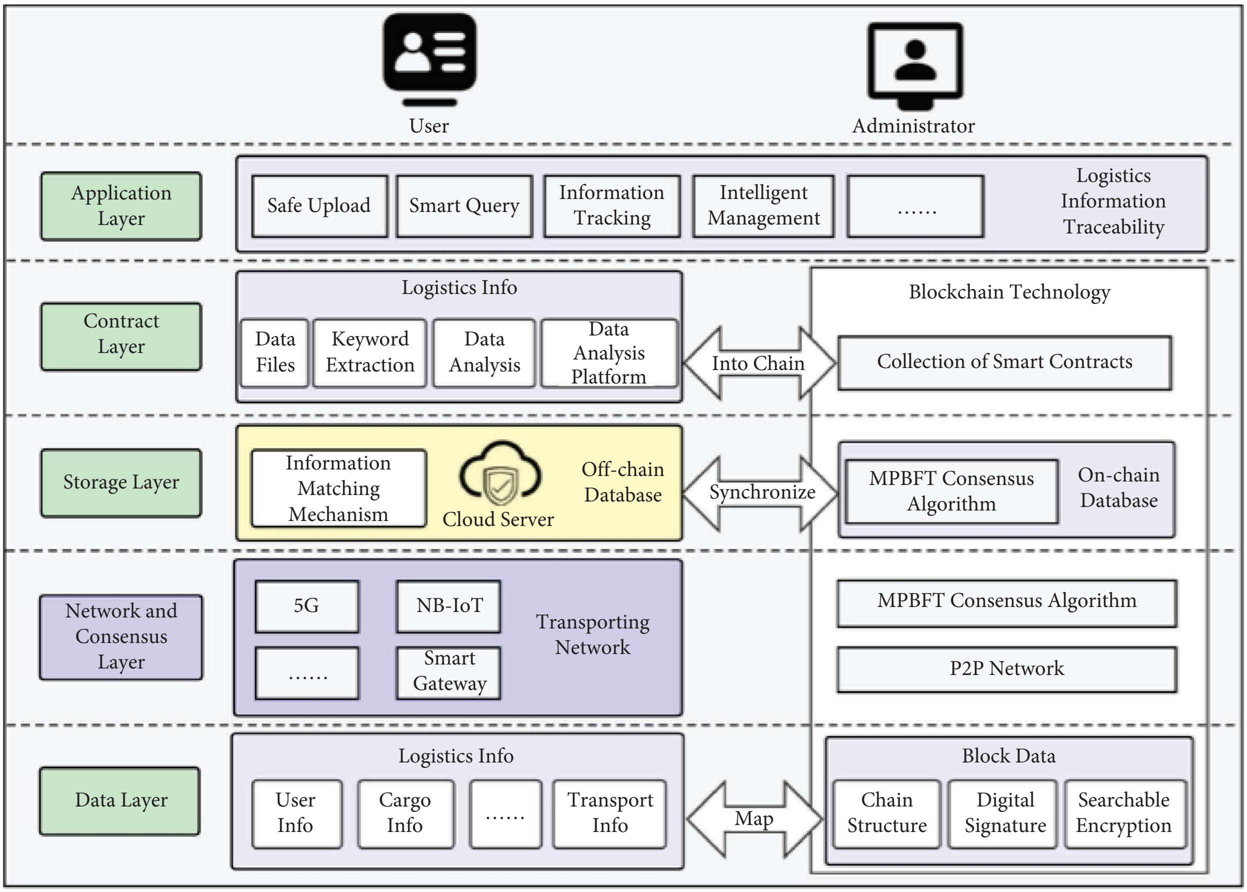


Figure 1: Overall architecture diagram of logistics information traceability system.

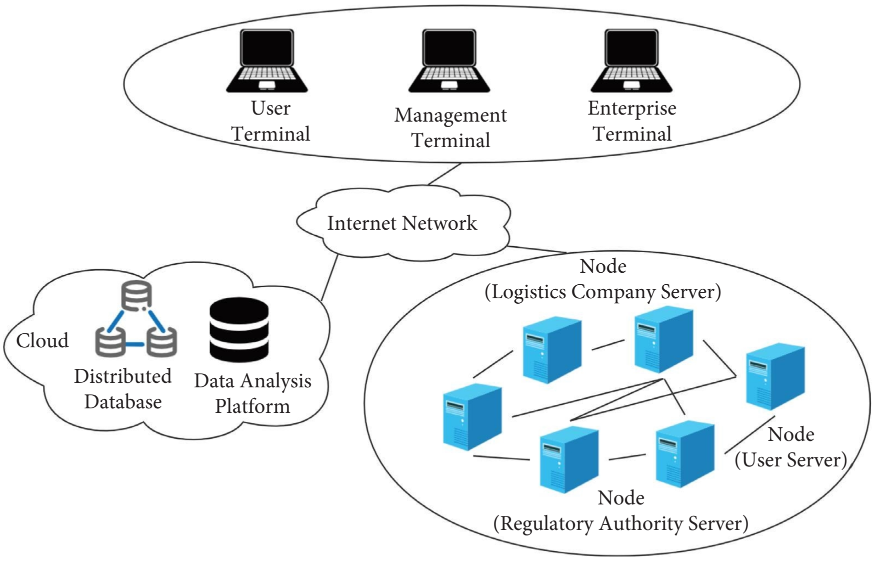


Figure 2: Network deployment diagram of logistics information traceability system.

4.2. MPBFT Algorithm Design. The basic PBFT consensus algorithm requires high-frequency communication to ensure that the consensus message packet can reach all nodes. As the number of nodes increases, the bandwidth and consensus delay will greatly increase. In the event of a faulty node, the communication connection will be lost, and there will also be malicious nodes. In this paper, the forwarding and packet structure optimization of the PBFT algorithm is carried out to improve consensus efficiency and avoid redundant consensus message packets occupying a large

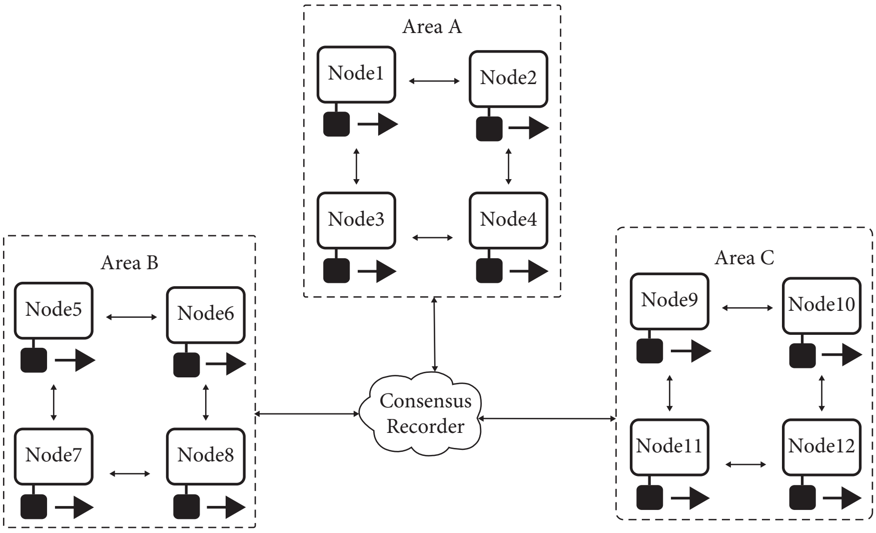


FIGURE 3: Group division diagram for logistics information traceability.

amount of bandwidth. First, the master node obtains a copy of the node set the record in the network through the consensus recorder. The copy in the consensus recorder needs to be registered with the global node during network construction, and each node will synchronize the copy of the consensus recorder. The MSP management mechanism of the alliance chain is introduced here. Jointly manage the number of nodes and state information in the consensus recorder. The master node sends the transaction packet that needs to be broadcast to the consensus recorder, the consensus recorder sends a new consensus notification to each node, and each node executes the consensus process of PBFT.

As shown in Figure 4, consensus blocks in this paper mainly include the following:

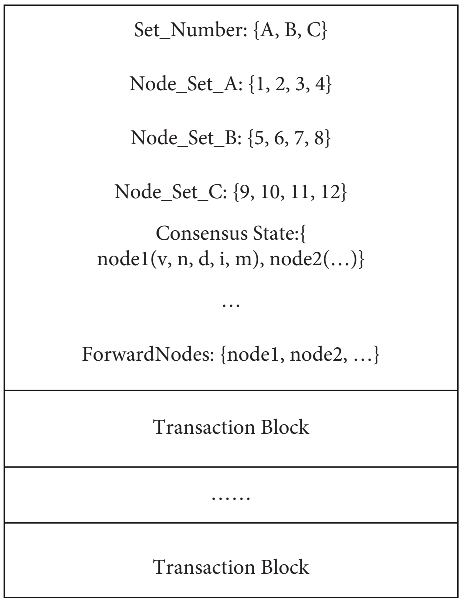
(1) The field Set_Number in the consensus recorder contains all the sets M in the network, the node names in each set, and the consensus state of the nodes.

(2) The Node_Set_Name field represents the node name in the set and records the order between adjacent nodes.

(3) The consensus state field indicates the consensus state of each node, including the verification data (v, n, d, i, m) of each node in the consensus stage, and the consensus recorder compares and confirms the response message packets of each node to determine whether the consensus is successful.

(4) The forward nodes field is used for offline consensus forwarding by the adjacent nodes of the faulty node when a communication failure occurs on the faulty node (introduced below).

(5) We set the timing of each node to synchronize the copy with the consensus recorder, and each node will save the copy to confirm whether the consensus data of the consensus stage is consistent with the consensus data of other nodes and meet the consensus condition of $2f + 1$.


Set_Number: {A, B, C}

Node_Set_A: {1, 2, 3, 4}

Node_Set_B: {5, 6, 7, 8}

Node_Set_C: {9, 10, 11, 12}

Consensus State:{
node1(v, n, d, i, m), node2(…)}

…

ForwardNodes: {node1, node2, …}

Transaction Block

……

Transaction Block

FIGURE 4: Consensus recorder model design.

The introduction of the consensus recorder in this paper greatly reduces the communication frequency between nodes and the communication bandwidth; each node only needs to obtain the consensus state of each node through the consensus recorder. We have modified the PBFT algorithm process to reduce the number of communications and offline forwarding when a faulty node occurs.

4.3. MPBFT Algorithm Flow

4.3.1. Role Division

(i) Client node is responsible for sending transaction requests

(ii) Primary node is responsible for packaging transactions into blocks and block consensus. There is only one primary node in each round of consensus

(iii) Replica node is responsible for block consensus, there are multiple replica nodes in each round of the consensus process, and the processing process of each replica node is similar

Among them, both primary and replica nodes belong to consensus nodes.

4.3.2. Algorithm Flow. As shown in Figure 5, the basic process of the PBFT algorithm mainly includes the following four steps:

(i) The client sends a request to the master node

(ii) The master node broadcasts the request to other nodes, and the nodes execute the three-phase consensus process of the PBFT algorithm

(iii) After the node processes the three-stage process, it returns a message to the client

(iv) After the client receives the same message from $f+1$ nodes, the consensus has been completed correctly

The three core stages of the algorithm are the preprepare stage, the prepare stage, and the commit stage. C in the figure represents the client, 0, 1, 2, and 3 represent the number of the nodes, of which 0 is the primary node, and the three marked with an X represents a possible fault node or a malicious node. The behavior shown here is for other nodes. The whole process is rough as follows.

First, the client initiates a request to the master node 0 $\ll \mathrm{REQUEST}, o, t, c \gg$ where t is the timestamp, o represents the operation, c is the client, the master node receives the client request, and the master node sends the consensus recorder. When the request packet is sent, the consensus recorder will send a preprepare message to other nodes. Other nodes will receive the preprepare message and start the core three-phase consensus process.

(1) Preprepare Stage. After the replica node receives the preprepare message, there are two choices, one is to accept and the other is not to accept. A typical case of not accepting is if a replica node receives a preprepare message $\ll \mathrm{PRE\_PREPARE}, v, n, d >, m >$, where v represents the view number, v represents the sequence number (the primary node receives the client each request on the side is marked with a number), d represents the message digest, and m represents the raw message data. The v and n in the message have appeared in the message received before, but they d, and m are inconsistent with the previous message, and the request will be rejected now. The rejection logic is that the master will not send two messages with the same v and n but different d and m.

The replica node receives the preprepare message and performs the following message verification:

(i) The signature of the message m is valid, and the message digest d matches the message m: $d = \mathrm{hash}(m)$

(ii) The node is currently in view v

(iii) The node currently has no other preprepare messages on the same (view , sequence n). That is, there is no other m' and corresponding d', $d' = \mathrm{hash}(m')$.

(2) Prepare Phase. After the current node agrees to the request, it will send a prepare message $\langle \mathrm{PREPARE}, v, n, d, i \rangle$ to the consensus state of the consensus recorder and record the message in the log, where i is used to represent the identity of the current node. At the same time, not only one node is going through this process. There may be n nodes that are also going through this process. Therefore, the consensus recorder will record the prepare messages sent by different nodes. The current node synchronizes the copy of the consensus recorder. The current node i verifies whether the data v, n, and d of these prepared messages and the prepared messages sent by itself are consistent. After the verification is passed, the current node i sets prepared (m, v, n) to true and sent to the consensus recorder. Prepared (m, v, n) represents the consensus node believing that the message m in (v, n) whether the prepare phase has been completed. The preparation phase has been completed if the consensus recorder replica receives prepare messages from more than $2f$ other nodes within a certain time frame. Finally, the consensus node i sends the commit message and enters the commit stage.

(3) Commit Stage. The current node i receives $2f$ commit messages $\langle \mathrm{COMMIT}, v, n, d, i \rangle$ from the copy of the consensus recorder and inserts the messages into the log ($2f+1$ including its own), verifying after these commit messages are consistent with the three data of v, n, and d in the commit message sent by themselves, the consensus node sets committed-local (m, v, n) to true, committed-local(m, v, n). On behalf of the consensus node, it is determined that the message least $2f+1$ nodes have agreed with the message m in the entire system, and this ensures that at least $f+1$ nonfaulty nodes have reached a consensus on the message m. Then, the node will execute the request and write the data.

After processing, the node will return the message $\ll \mathrm{REPLY}, v, t, c, i, r \gg$ to the client. When the client collects $f+1$ messages, the consensus is completed, which is the whole process of the PBFT algorithm.

4.3.3. Communication Disconnection. Suppose there is a communication failure between the node and the consensus recorder to ensure that all nodes are fully connected. In that case, messages need to be forwarded to the faulty node. This process occurs between nodes in the node set. The normal nodes adjacent to the faulty node forward the copy of the consensus recorder.

As shown in Figure 6, the consensus process of adjacent nodes is as follows:

(i) The consensus recorder sends a PBFT message to {node 0, node 1, node 2, node 3} and finds that {node 1, node 3} is not in the connection list (faulty node), then sets the forward nodes field of the PBFT message msg to {node 1, node 3}, and forwards it to

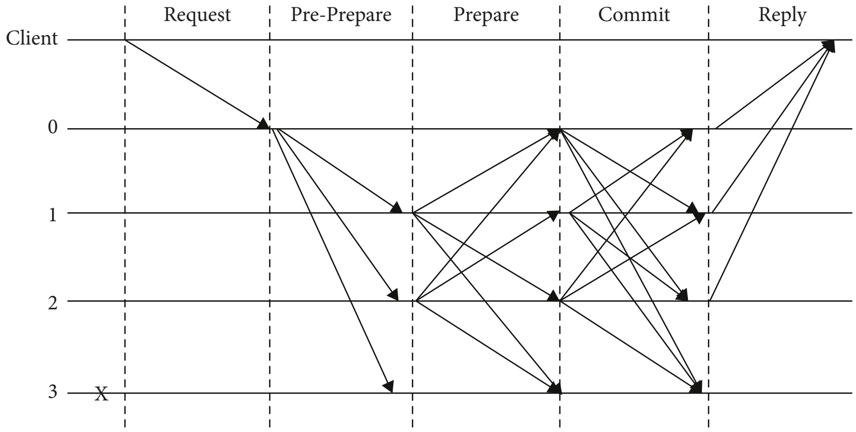


Figure 5: MPBFT consensus algorithm process.

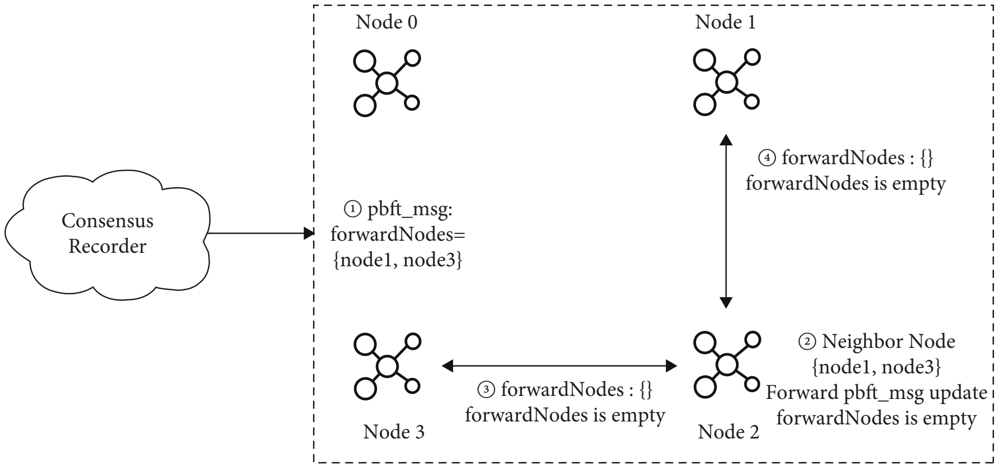


Figure 6: Communication forwarding model of disconnected neighbor nodes in the network.

node 0 and node 2; according to the principle of the proximity of nodes, this paper uses four nodes in the node set as an example.

(ii) If node 2 determines that the forward nodes field is not empty after receiving the PBFT message from the consensus recorder, it traverses the neighbor node list {node 1, node 3} and removes the neighbor nodes from forward nodes.

(iii) Node 2 forwards the updated PBFT message msg to node1 and node 3.

(iv) After node1 and node3 receive the msg, they judge that the forward nodes field is empty, we consider that the message has reached all nodes and do not continue to forward the PBFT message.

As shown in Figure 7, the improved PBFT message forwarding strategy adds the forward nodes field to the PBFT message packet in the copy of the consensus recorder to record the disconnected node information. After receiving the PBFT message packet, other nodes forward the message to the reachable nodes (adjacent nodes) recorded by forward nodes, ensuring that PBFT message packets can reach all nodes as much as possible, reducing redundant PBFT messages in the network and improving network efficiency. Since the message packet of the faulty node is forwarded and received by the adjacent node, the authenticity of the forwarding process of the adjacent node is recorded by the consensus recorder. If the node state of the consensus recorder is $\ll \text{PRE\_PREPARE}, v, n, d >, m >$, the values of v, n, d, and m are verified, and if they are not the same, it is judged as a malicious node.

4.4. Analysis of Communication Times. This section will compare the number of communications between the MDPBFT algorithm and the classic PBFT algorithm, the COMBFT algorithm [42], and the SPBFT algorithm [43] in the consensus process to verify that the MBFT algorithm has a relatively small number of communications.

Assuming that the number of nodes in the network is N, in the first stage of communication in the basic PBFT algorithm, the client first sends a request to the master node, and then the master node broadcasts the request to the rest of the nodes. The number of communications in the first stage is $(N-1)$ times. In the second stage, the remaining nodes need to respond to the pre-preparation message sent by the master node, broadcast consensus to each other, and

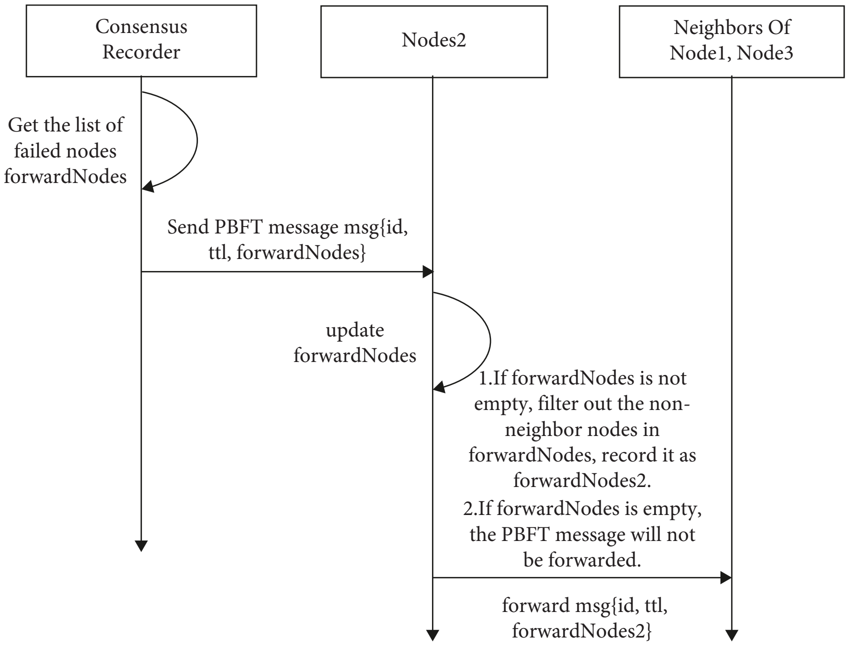

Consensus Recorder
Nodes2
Neighbors Of Node1, Node3
Get the list of failed nodes forwardNodes
Send PBFT message msg{id, ttl, forwardNodes}
update forwardNodes
1.If forwardNodes is not empty, filter out the non-neighbor nodes in forwardNodes, record it as forwardNodes2.
2.If forwardNodes is empty, the PBFT message will not be forwarded.
forward msg{id, ttl, forwardNodes2}

Figure 7: The communication forwarding process of the network disconnecting adjacent nodes.

the number of communications is $(N-1)^2$. All nodes in the third-stage system perform consensus feedback on the second-stage messages. Send an acknowledgment message to the master node, and the number of communications at this stage is $N(N-1)$. Based on the above analysis, it can be concluded that the communication times T_1 for the classical PBFT consensus algorithm to complete a consensus is as follows:

$$T_1 = 2N(N-1). \tag{2}$$

Unlike PBFT, in the COMBFT consensus algorithm, the client sends the message to all nodes when sending a request message. This is the first stage, and the number of communications is N. This mechanism divides the consensus protocol into normal and abnormal cases (with or without malicious nodes). In the normal case, the consensus process is designed into three stages. In the second stage, the master node sends messages to the slave nodes, and the number of communications is $N-1$. In the third stages, all nodes send reply messages to the client, and the number of transmissions is N, and then the number of contacts in the normal case is as follows:

$$T_2 = 3N(N-1). \tag{3}$$

The conflict-ordering mechanism (COM) needs to be implemented in abnormal cases. The SUSPECT phase is added to the consistency protocol. Each slave node sends SUSPECT messages to others except in this phase, and the master node sends messages to the client. The number of communications is $N*(N-1)+1$, and the number of contacts in the abnormal case is as follows:

$$T_3 = N*(N+2). \tag{4}$$

In the SPBFT consensus algorithm, although three stages of consensus are still required, it uses data encoding to assign different data requests to different nodes, so there is no need to broadcast messages to all other nodes in the preparation and confirmation stages. In the algorithm, the requested data is divided into multiple subdata, assuming that the number is also K. In the preliminary stage, the number of communications is still $(N-1)$. In the preparation phase, each node, except the master node only broadcasts the message to the other two nodes related to the news, and the number of communications in this phase is $2(N-1)$ times. In the confirmation phase, each node broadcasts the message to at least the other two nodes, and the number of communications in this phase is at least $2N$ times. Therefore, the number of contacts of the SNCPBFT algorithm is as follows:

$$T_4 = (5N-3)*K. \tag{5}$$

In the MPBFT algorithm, during a round of the consensus process, the node mainly communicates with the consensus recorder. In the first stage, the master node sends a message to the consensus recorder. The number of communications is 1. In the second stage, the consensus recorder sends $(N-1)$ messages. Nodes broadcast consensus messages and carry consensus copies. The third-stage node submits the consensus record to the consensus recorder and verifies whether the consensus of the remaining nodes satisfies the condition greater than $(2f-1)$, where the number of communications $(N-1)$. If the condition is met, each node submits the consensus confirmation result and returns the consensus recorder result $(N-1)$. The consensus recorder returns the result to the master node with a communication count of 1. To sum up,

$$T_5 = 2+3(N-1). \tag{6}$$

We compare T with the above four communication times because when $N \geq 12$ and the node scale is large, there are $T_5 < T_1$, $T_5 < T_2$, $T_5 < T_3$, and $T_5 < T_4$. So overall, the communication times of this scheme are less.

4.5. Security Analysis

4.5.1. The Multiset Architecture Proposed in This Paper Is Safe and Effective

Proof. In the PBFT algorithm, the PRE-PREPARE stage is the stage of requesting sequence number assignment, which is used to ensure that the request has been assigned sequence numbers in the current view. When multiple clients send request information to the nodes in the set, the nodes in the set will assign serial numbers and execute requests in sequence according to the preassigned serial numbers. Due to various attacks, this paper designs the nodes under the multiarea logistics information for identity verification and guarantees the nodes through a third-party identity verification agency. That is, each node that joins the network must be authenticated by a third-party agency. Nodes can be uniquely defined through hash value, key, and DNS name computer mechanism to prevent attacks such as Sybil attack. This method relies on verification by other institutions and will lose some of the node's anonymity. In the preparation stage, the request with the previous serial number will be executed first, and the node initiating the request will broadcasts the preparation message $\langle \text{PREPARE}, m, v, n, t, i \rangle$ through the consensus recorder. This stage is a preliminary consensus; in the confirmation stage, each node gives feedback on the messages broadcast by the consensus recorder. That is, each node sends a confirmation message $\langle \text{PREPARE}, m, v, n, t, i \rangle$ to the sending node through the consensus recorder. During this process, the consensus recorder will record and compare the return information of each node. If there is a malicious node, a node log will be recorded. Through the above process, the consensus validity of the multiset architecture can be guaranteed.

4.5.2. The Security and Data Validity of the Consensus Recorder

Proof. The consensus recorder is in the network and is not controlled by any node. The independence of the consensus recorder is guaranteed by updating the public smart contract. The execution of events (requests, confirmations, verifications) is automatically triggered by smart contracts (when set conditions are met). When a node sends a request, the smart contract of the consensus recorder generates a new consensus log, and the feedback of each node in the confirmation stage is recorded in the log. The smart contract of the consensus recorder automatically verifies the validity of the consensus of the node. It is enough to ensure that the feedback confirmation message satisfies $2f + 1$ benign nodes.

4.5.3. The Messages Responded to by the Node Set in This Paper Are Unique

Proof. There is no direct communication between node sets, and messages need to be transmitted through the consensus recorder. When a node in a set broadcasts a message, the consensus recorder records and relays the broadcast message, waiting for other nodes to feedback confirmation messages; feedback messages between sets are Recorded and verified in the consensus recorder, blocking data forgery communication between collection nodes. When the verification of nodes in a certain set is inconsistent, the malicious node is quickly located and recorded in the consensus recorder. When the verification of all nodes in a certain set is inconsistent with that of other set nodes, $2f + 1$ benign sets are also applicable.

5. Performance Analysis

This section compares the MPBFT algorithm proposed in this paper with the PBFT algorithm, SPBFT algorithm, and COMPBFT algorithm [42] regarding network bandwidth consumption and network delay. The experimental results illustrate the communication of the algorithm in this paperless overhead. The experiment's configuration in this paper is a Windows 10 system with an i7-4558U processor and 8 G.B. memory. The experimental simulation is carried out through MATLAB R2017a, and the number of nodes is 50, 100, 150, and 200. The size of the transmitted data is the same.

5.1. Network Bandwidth. Since the block size is constant, the required network bandwidth increases with the number of nodes. Different algorithms occupy different bandwidths. Algorithms with less communication time can occupy less bandwidth and thus consume less energy. The changes in network bandwidth and the number of nodes for the four algorithms are shown in the following figure. PBFT and SPBFT algorithms have three-stage message broadcasts, COMBFT has four groups of broadcasts, and MPBFT reduces the number of broadcasts and geographic distance through consensus repeaters. Therefore, the bandwidth occupied by the network is smaller than the other three algorithms.

5.2. Network Delay. Network latency is an important indicator reflecting the running speed of the blockchain system. The lower the network latency, the higher the system efficiency, the faster the consensus, and the higher the transaction efficiency. The relationship between the network delay and the number of nodes of the four algorithms is shown in Figure 8. It can be seen from the figure that with the increase in the number of nodes, the network delay is prolonged to a certain extent, and the network delay of the PBFT algorithm increases the most, which also explains the reason for improving the PBFT algorithm. As can be seen from the figure in this paper, the proposed MPBFT algorithm has a low delay and is relatively stable, which also

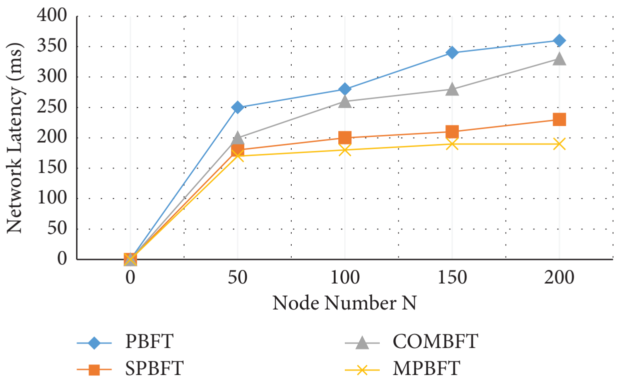


Figure 8: Comparison of network delays of four algorithms with different numbers of nodes.

shows that the scheme in this paper is more suitable for systems with a large number of nodes.

5.3. Throughput. Throughput is an important indicator reflecting the transaction rate of the blockchain system. The higher the throughput, the more the transactions per unit of time and the higher the consensus efficiency. The relationship between the throughput ratio of the four algorithms and the nodes is shown in Figure 9. It can be seen from the figure that as the number of nodes increases, the network throughput of the global environment will gradually decrease. When there are 200 nodes, the throughput of MPBFT is high. On the other three nodes, the MPBFT decline rate reflects that the multiset architecture can effectively improve the throughput of the multinode case. This also shows that the MPBFT consensus algorithm is relatively stable.

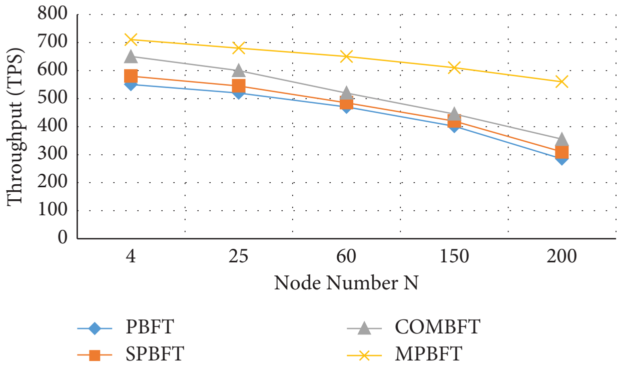


Figure 9: Throughput comparison of four algorithms with different numbers of nodes.

5.4. Algorithm Comparison. It can be seen from Figures 8–10 that PBFT, SPBFT, COMBFT, and MPBFT have low latency, high bandwidth, and high throughput in the 200-node MPBFT network. As shown in Table 3, the performance of the MPBFT consensus process is better than traditional algorithms (PBFT, SPBFT, and COMBFT) is more prominent. MPBFT consists of 3-stage communication and consensus recorder, which can reduce the consensus delay to a greater extent. The channel capacity is determined by the internal structure of the block. Only the block serial number needs to be agreed upon between the node sets. After the consensus is completed, the data in the block serial number is synchronized. Therefore, more block serial number data can be carried out during the consensus process. Scalability of consensus algorithms in the process of adding nodes, PBFT, SPBFT, and COMBFT all directly add nodes globally, while MPBFT adds nodes to the set, and when the consensus set reaches its peak, it expands the new set. When many nodes are added, the MPBFT global consensus performance is less affected. In terms of applicability, PBFT, SPBFT, and COMBFT are not suitable for long-distance geographic location consensus, and the global consensus network delay is greatly affected by communication. MPBFT adopts a multiset architecture and deploys consensus sets based on geographic locations, which greatly reduces communication delays.

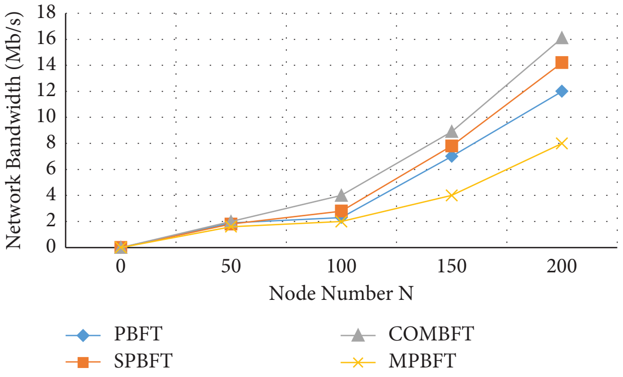


Figure 10: Comparison of network bandwidth of four algorithms with different numbers of nodes.

Table 3: The proposed algorithm is compared with the traditional algorithm.

Consensus algorithm	Throughput	Latency	Channel capacity	Scalability	Adaptability
PBFT	Low	3-stage medium latency	Low	Add nodes performance drops	Low latency short distance
SPBFT	Low	3-stage medium latency	Low	Add nodes performance drops	Low latency short distance
COMBFT	Medium	4 stages high latency	Medium	Add nodes performance drops	High fault tolerance short distance
MPBFT	High	3 stages and consensus recorder, low latency	High	Add nodes performance stable	High latency long distance

6. Conclusion

This paper first introduces the theory and technology involved in the blockchain logistics information traceability system. According to the characteristics of the logistics industry, a more efficient multiset consensus algorithm MDBFT is proposed, and the basic model and the entire implementation process of the algorithm are described. The feasibility of the algorithm is demonstrated by analyzing the communication times of the MDBFT algorithm. Through the comparative experiment and simulation of network bandwidth and network delay, it is verified that the algorithm can improve the consensus efficiency of logistics information, thereby ensuring the information's authenticity and traceability efficiency.

This paper proposes a logistics information traceability system based on blockchain. By improving consensus efficiency, an efficient traceability process is completed. With the development of blockchain technology, the system still needs to be perfected in practical application. This paper only improves the efficiency of logistics information traceability from consensus efficiency. However, there are still many factors that affect the efficiency of traceability, and further research is needed. This paper only considers the geographic information generated by the general logistics and circulation links and studies the traceability of general transportation route information to ensure the authenticity of the information traceability. In the future, the model can be further optimized. For complex logistics links, such as cold chain transportation, cross-border transportation, and other complex scenarios, various information such as temperature, humidity, and weather can be added.

Data Availability

The PBFT consensus time data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by the Energy Cloud R&D Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Science, ICT (grant no. 2019M3F2A1073387), and Institute for Information & Communications Technology Promotion (IITP) (grant no. 2022-0-00980; Cooperative Intelligence Framework of Scene Perception for Autonomous IoT Device).

References

[1] X. Lin, P. Jing, C. Yu, and X. Feng, "TPLI: a traceable privacy-preserving logistics information scheme via blockchain," in *Proceedings of the 2021 International Conference on Networking And Network Applications (NaNA)*, pp. 345–350, IEEE, Lijiang City, China, October 2021.

[2] M. Westerkamp, F. Victor, and A. Küpper, "Blockchain-based supply chain traceability: token recipes model manufacturing processes," in *Proceedings of the 2018 IEEE International Conference on Internet Of Things (iThings) and IEEE Green Computing And Communications (GreenCom) and IEEE Cyber, Physical And Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pp. 1595–1602, IEEE, Halifax, NS, Canada, July 2018.

[3] S. Liang, M. Li, and W. Li, "Research on traceability algorithm of logistics service transaction based on blockchain," in *Proceedings of the 2019 18th International Symposium on Distributed Computing And Applications For Business Engineering And Science (DCABES)*, pp. 186–189, IEEE, Wuhan, China, November 2019.

[4] M. Schwarz, M. Lipp, D. Moghimi et al., "Cross-Privilege-Boundary Data Sampling," 2019, https://arxiv.org/abs/1905.05726.

[5] S. Wu, M. A. Rizoiu, and L. Xie, "Variation across scales: measurement fidelity under twitter data sampling," in *Proceedings of the international AAAI conference on web and social media*, vol. 14, pp. 715–725, California, CA, USA, May 2020.

[6] S. Ramírez-Gallego, B. Krawczyk, S. García, M. Woźniak, and F. Herrera, "A survey on data preprocessing for data stream mining: current status and future directions," *Neurocomputing*, vol. 239, no. C, pp. 39–57, 2017.

[7] P. Ranganathan, C. S. Pramesh, and R. Aggarwal, "Common pitfalls in statistical analysis: logistic regression," *Perspectives in clinical research*, vol. 8, no. 3, pp. 148–151, 2017.

[8] S. A. Alasadi and W. S. Bhaya, "Review of data preprocessing techniques in data mining," *Journal of Engineering and Applied Sciences*, vol. 12, no. 16, pp. 4102–4107, 2017.

[9] M. Nofer, P. Gomber, O. Hinz, and D. Schiereck, "Blockchain," *Business & Information Systems Engineering*, vol. 59, no. 3, pp. 183–187, 2017.

[10] L. Xu, L. Chen, Z. Gao, Y. Chang, E. Iakovou, and W. Shi, "Binding the physical and cyber worlds: a blockchain approach for cargo supply chain security enhancement," in *Proceedings of the 2018 IEEE International Symposium on Technologies For Homeland Security (HST)*, pp. 1–5, IEEE, Woburn, MA, USA, October 2018.

[11] L. Zhang, L. Hang, and D. Kim, "Design of logistics information traceability system based on blockchain," in *Proceedings of the Korea Information Processing Society Conference*, pp. 244–247, Seoul Republic of Korea, December 2022.

[12] L. Barreto, A. Amaral, and T. Pereira, "Industry 4.0 implications in logistics: an overview," *Procedia Manufacturing*, vol. 13, pp. 1245–1252, 2017.

[13] M. Pease, R. Shostak, and L. Lamport, "Reaching agreement in the presence of faults," *Journal of the ACM*, vol. 27, no. 2, pp. 228–234, 1980.

[14] S. Nakamoto and A. Bitcoin, "A peer-to-peer electronic cash system," *Bitcoin*, vol. 4, no. 2, 2008.

[15] D. Larimer, "Delegated proof-of-stake white paper," *IEEE Access*, vol. 7, pp. 10–1109, 2014.

[16] M. Castro and B. Liskov, "Practical byzantine fault tolerance," *OsDI*, vol. 99, pp. 173–186, 1999.

[17] M. Castro and B. Liskov, "Practical Byzantine fault tolerance and proactive recovery," *ACM Transactions on Computer Systems*, vol. 20, no. 4, pp. 398–461, 2002.

[18] J. Borràs, A. Moreno, and A. Valls, "Intelligent tourism recommender systems: a survey," *Expert Systems with Applications*, vol. 41, no. 16, pp. 7370–7389, 2014.

[19] H. T. Nguyen, T. Almenningen, M. Havig et al., “Learning to rank for personalised fashion recommender systems via implicit feedback,” in *Mining Intelligence and Knowledge Exploration*,Springer, Berlin, Germany, 2014.

[20] L. Marin, M. Pawlowski, and A. Jara, “Optimized ECC implementation for secure communication between heterogeneous IoT devices,” *Sensors*, vol. 15, no. 9, pp. 21478–21499, 2015.

[21] Z. Dong, J. Chen, Y. Chen, and R. Shao, “Food traceability system based on blockchain,” in *Proceedings of the 2020 International Conference On Aviation Safety And Information Technology*, pp. 571–576, Weihai, China, October 2020.

[22] X. Liu, A. V. Barenji, Z. Li, B. Montreuil, and G. Q. Huang, “Blockchain-based smart tracking and tracing platform for drug supply chain,” *Computers & Industrial Engineering*, vol. 161, Article ID 107669, 2021.

[23] C. H. Wu, Y. P. Tsang, C. K. M. Lee, and W. K. Ching, “A blockchain-IoT platform for the smart pallet pooling management,” *Sensors*, vol. 21, no. 18, p. 6310, 2021.

[24] S. Ni, X. Bai, Y. Liang, Z. Pang, and L. Li, “Blockchain-based traceability system for supply chain: potentials, gaps, applicability and adoption game,” *Enterprise Information Systems*, vol. 16, no. 12, Article ID 2086021, 2022.

[25] J. M. Song, J. Sung, and T. Park, “Applications of blockchain to improve supply chain traceability,” *Procedia Computer Science*, vol. 162, pp. 119–122, 2019.

[26] A. S. Omar and O. Basir, “Smart phone anti-counterfeiting system using a decentralized identity management framework,” in *Proceedings of the 2019 IEEE Canadian Conference of Electrical And Computer Engineering (CCECE)*, pp. 1–5, IEEE, Edmonton, AB, Canada, May 2019.

[27] M. Rajesh, “Anti-counterfeiting and traceability mechanism based on blockchain,” *Recent Trends in Intensive Computing*, vol. 39, p. 134, 2021.

[28] N. C. K. Yiu, “Toward blockchain-enabled supply chain anti-counterfeiting and traceability,” *Future Internet*, vol. 13, no. 4, p. 86, 2021.

[29] T. McConaghy, R. Marques, A. Müller et al., “Bigchaindb: A Scalable Blockchain Database,” *White paper BigchainDB*, Springer, Berlin, Germany, 2016.

[30] W. Yu and S. Huang, “Traceability of food safety based on block chain and RFID technology,” in *Proceedings of the 2018 11th International Symposium on Computational Intelligence and Design (ISCID)*, vol. 1, pp. 339–342, IEEE, Hangzhou, China, December 2018.

[31] Z. Xie, H. Kong, and B. Wang, “Dual-Chain Blockchain in Agricultural E-Commerce Information Traceability Considering the Viniar Algorithm,” *Scientific Programming*, vol. 2022, Article ID 2604216, 2022.

[32] C. Xu, K. Chen, M. Zuo, H. Liu, and Y. Wu, “Urban fruit quality traceability model based on smart contract for Internet of Things,” *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 9369074, 10 pages, 2021.

[33] Z. Wang, T. Wang, H. Hu, J. Gong, X. Ren, and Q. Xiao, “Blockchain-based framework for improving supply chain traceability and information sharing in precast construction,” *Automation in Construction*, vol. 111, Article ID 103063, 2020.

[34] Y. Chen and F. Liu, “A Multi-Blockchain System Application Based on Improved PBFT Consensus Mechanism for Online Rumor Co-governance,” 2022, https://assets.researchsquare.com/files/rs-2188736/v1_covered.pdf?c=1667892603.

[35] S. Coretti, A. Kiayias, C. Moore, and A. Russell, “The Generals’ Scuttlebutt: Byzantine-Resilient Gossip Protocols,” in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, Los Angeles CA USA, November 2022.

[36] B. Arun, S. Peluso, and B. Ravindran, “ezBFT: decentralizing Byzantine fault-tolerant state machine replication,” in *Proceedings of the 2019 IEEE 39th International Conference On Distributed Computing Systems (ICDCS)*, pp. 565–577, IEEE, Dallas, Texas, USA, July 2019.

[37] A. Litke, D. Anagnostopoulos, and T. Varvarigou, “Blockchains for supply chain management: architectural elements and challenges towards a global scale deployment,” *Logistics*, vol. 3, no. 1, p. 5, 2019.

[38] S. Brotsis and N. Kolokotronis, “Blockchain-Enabled digital forensics for the IoT: challenges, features, and current frameworks,” in *Proceedings of the 2022 IEEE International Conference on Cyber Security And Resilience (CSR)*, pp. 131–137, IEEE, Rhodes, Greece, July2022.

[39] Z. Tian, M. Li, M. Qiu, Y. Sun, and S. Su, “Block-DEF: a secure digital evidence framework using blockchain,” *Information Sciences*, vol. 491, pp. 151–165, 2019.

[40] “Internet of Things architecture,” 2022, https://www.ibm.com/cloud/architecture/architectures/iotArchitecture/reference-architecture.

[41] D. A. Horn and G. Bone, “Developing a business model for product environmental stewardship within IBM,” in *Proceedings of the IEEE International Symposium on Sustainable Systems And Technology (ISSST 2010)*, p. 1, IEEE Computer Society, Arlington, VA, USA, May 2010.

[42] Y. Rong, W. Wu, and Z. Chen, “Combft: conflicting-order-match based byzantine fault tolerance protocol with high efficiency and robustness,” in *Proceedings of the 48th International Conference On Parallel Processing*, pp. 1–10, Kyoto Japan, August 2019.

[43] B. Choi, J. Y. Sohn, D. J. Han, and J. Moon, “Scalable network-coded PBFT consensus algorithm,” in *Proceedings of the 2019 IEEE International Symposium on Information Theory (ISIT)*, pp. 857–861, IEEE, Paris, France, July 2019.