electronics

MDPI

Article

A Blockchain-Based Framework for Rural Property Rights Transactions

Caijian Hua [1,2,*,†], Sichao Wu [1,†], Yan Zhang [1,2], Kun Luo [1], Miaomiao Li [1] and Jiaguo Fu [1]

[1] School of Computer Science and Engineering, Sichuan University of Science and Engineering, Yibin 644000, China; luokun306@163.com (K.L.); fujiaguo172540@163.com (J.F.)
[2] Key Laboratory of Higher Education of Sichuan Province for Enterprise Informationalization and Internet of Things, Sichuan University of Science and Engineering, Yibin 644000, China
* Correspondence: hwacj@suse.edu.cn
† These authors contributed equally to this work.

Abstract: Currently, there are several issues in China's rural property rights trading, including poor credit reporting, low transaction efficiency, non-standardized transaction contracts, and high costs for rights protection. However, blockchain technology, known for its convenience, security, and traceability, is highly compatible with the requirements of standardization, convenience, and agility in rural property rights trading. Therefore, we propose a system framework for rural property rights trading based on the Hyperledger Fabric consortium blockchain. This framework significantly enhances transaction efficiency, improves the security of rural property rights trading, effectively resolves contract disputes in property rights transactions, and promotes rural revitalization and development. Furthermore, we design the information data structure on the blockchain, present the information flow, and develop smart contracts that utilize automation capabilities to automatically summarize, analyze, and issue alerts based on on-chain data, enabling risk assessment for both buyers and sellers. Additionally, we introduce an improved PBFT consensus algorithm called CA-PBFT, which is based on the PBFT consensus protocol and integrates a scoring model and simplified consensus protocol. CA-PBFT efficiently reduces transaction response latency and improves information throughput. Finally, we develop a prototype system for rural property rights trading and perform a performance comparison and analysis of CA-PBFT against other PBFT-based consensus algorithms in this prototype system. Experimental results demonstrate the feasibility of the prototype system framework for rural property rights trading.

Keywords: rural property rights transaction; blockchain; HyperLedger fabric; CA-PBFT

Citation: Hua, C.; Wu, S.; Zhang, Y.; Luo, K.; Li, M.; Fu, J. A Blockchain-Based Framework for Rural Property Rights Transactions. *Electronics* **2023**, *12*, 4334. https://doi.org/10.3390/electronics12204334

Academic Editors: Satyabrata Aich, Kamalakanta Muduli and Sushanta Tripathy

Received: 15 August 2023
Revised: 30 September 2023
Accepted: 16 October 2023
Published: 19 October 2023

1. Introduction

In recent years, significant transformations and innovations have unfolded in the realm of rural property rights in China. The Central Committee of the Communist Party of China and the State Council have issued the *Opinions on Fully Advancing Rural Revitalization and Accelerating Agricultural and Rural Modernization*, which explicitly emphasize the strengthening of rural property rights transfer transactions and the construction of management information network platforms to provide comprehensive transaction services. Simultaneously, various local governments have actively responded to national policies, striving to enhance the digital management of agriculture and rural areas, thereby promoting the process of rural modernization.

In this context, blockchain technology has gradually emerged as an innovative tool to address issues pertaining to rural property rights transactions. The distinctive features of blockchain technology, such as security, transparency, and tamper resistance, position it as a potent choice for building digital platforms for rural property rights transactions. For instance, in 2020, Suqian County in Suzhou achieved the first *cloud signing* of rural land

transfer contracts nationwide through the application of blockchain technology. Additionally, Kunshan in Jiangsu and Shaoguan in Guangdong have also completed the transfer of idle residential property rights based on blockchain technology, marking successful use cases of blockchain in rural property rights.

However, traditional rural property rights transaction markets still confront numerous challenges, including information asymmetry, inadequate security, low efficiency, lack of standardized transaction procedures, and relatively small market scales. These issues impede the development of modern agriculture in rural areas. Therefore, concerted efforts are required from governments and various stakeholders to strengthen management and regulation, refine relevant laws and regulations, optimize transaction standards and procedures, reinforce land transfer systems, and expand market scales. These actions are essential to promote the health and sustainability of rural property rights transaction markets.

Digital platforms for rural property rights transactions, as solutions based on internet technologies and information tools, provide online transaction platforms that offer convenient services such as property rights inquiries, evaluations, listings, and transactions for farmers. Simultaneously, they offer more opportunities and information for investors and capital markets, effectively addressing issues of information asymmetry and transaction complexity prevalent in traditional rural property rights transaction markets. The emergence of these platforms has fostered a more transparent and open transaction environment, offering increased information and protection for both parties involved while also reducing transaction costs and time.

Despite the conveniences introduced by digital platforms for rural property rights transactions, they are not without risks and security concerns. Firstly, there is a critical issue of information security risk, as these platforms deal with the personal and property information of farmers. If platform security is not adequately safeguarded, it may lead to information leakage or exploitation by malicious actors. Secondly, there is the risk of network attacks, such as hacker attacks and virus threats, which could result in losses for the platform and its users if not addressed promptly. Lastly, transaction risks must also be considered, as instances of false information or fraudulent behavior may harm the rights and interests of users.

To address these challenges, rural property rights transaction platforms must establish secure, transparent, and reliable data recording systems. In this regard, blockchain technology has garnered significant attention. Blockchain is regarded as an emerging information and communication technology (ICT) within the context of *Industry 4.0*. Its core attributes include decentralization, cryptographic algorithms ensuring data security, and smart contracts for the automated execution of agreements. Consensus algorithms guarantee the authenticity and security of transaction records within the blockchain network, all without relying on centralized institutions. Consequently, blockchain technology offers innovative solutions for constructing rural property rights transaction systems.

Nonetheless, there remains a dearth of in-depth research and exploration of the potential of blockchain technology in the domain of rural property rights transactions. This study aims to bridge this gap by proposing a conceptual framework for rural property rights transactions based on blockchain technology and optimizing the Practical Byzantine Fault Tolerance (PBFT) consensus algorithm within the context of rural property rights transactions. The contributions of our research encompass the following aspects:

1. We propose a blockchain-based conceptual framework for rural property rights trading, endowed with data tamper-proof capabilities, enhancing transparency and trustworthiness in the rural property rights trading platform. This effectively elevates the security of rural property rights transactions and provides a solution to contract disputes;
2. We design smart contracts that aggregate, analyze, and alert on-chain data, enabling functions such as transaction anomaly detection, auditing, and supervision. These smart contracts also support rural property rights confirmation, identity verification, and permission control;

3. In the context of rural property rights trading, we introduce an efficient consensus mechanism, CA-PBFT. This mechanism maintains security and stability while enhancing the completion rate of transaction requests and information throughput, all the while reducing consensus and transaction latency;
4. Leveraging the Hyperledger Fabric framework in a laboratory environment, we successfully develop a prototype system for rural property rights trading.

Through these research endeavors, we aspire to offer fresh perspectives and methodologies for the digitization and modernization of rural property rights transactions, thereby contributing to the sustainable development of rural areas in China and facilitating the successful implementation of the rural revitalization strategy. Furthermore, we acknowledge that digitizing rural property rights transactions involves structural and contextual factors that necessitate collaborative efforts from government bodies and diverse sectors of society to realize the digitalization and modernization goals of rural areas.

2. Related Research

In this section, we discuss the popular methods adopted in the property rights transaction system framework and the improved algorithm of the CA-PBFT consensus algorithm. Additionally, we provide an overview of relevant research on the practical applications of blockchain.

2.1. Property Rights Transaction Platform

In the past, innovation in property rights transaction platforms mainly focused on the integration of big data technology and cloud computing technology. Additionally, some research efforts have been made to develop integrated systems for property rights transactions. For instance, Li Zhaoxing [1] proposed a web-based intellectual property management system that effectively enhances the efficiency of intellectual property management. Wang Bo [2] addressed the challenges of scattered information, diverse property types, and difficult supervision in rural comprehensive property rights transactions by introducing a Hadoop-based rural comprehensive property rights transaction system.

However, previous research primarily focused on proposing various solutions, lacking consideration of how to ensure the security and reliability of user transaction data. Typically, the acquired data were stored in centralized systems, fully trusting a single data management entity (e.g., the government). However, property rights transactions involve multiple participants: (1) the government; (2) sellers; and (3) buyers. Finding a reliable data management entity can be challenging. For instance, the public may not trust the government's data, leading to disputes and conflicts between buyers and sellers. Moreover, there has been limited research on how to ensure the authenticity and validity of sellers' property rights. To address these issues, the development of a transparent, tamper-resistant, and traceable property rights transaction data monitoring and recording system is necessary.

2.2. Blockchain and Its Application in Property Rights Transactions

The history of blockchain can be divided into three stages: 1.0, 2.0, and 3.0+ [3]. *Bitcoin* [4] is an example of blockchain 1.0, representing a form of programmable currency related to financial applications. Using the example application *Ethereum* [5], smart contracts ushered in the era of blockchain 2.0. Smart contracts are described as computerized transaction protocols that execute the terms of a contract, enabling the blockchain to act as a self-executing platform for transactions [6]. Smart contracts existed before blockchain, but it was not until blockchain technology supported immutable smart contract content that they gained attention and became trustworthy. Programming languages such as Solidity in Ethereum made it possible to encode collective data processing protocols into smart contracts. Once successfully implemented, smart contracts are tamper-proof and can automatically execute once predefined parameters are met. Therefore, smart contracts enable the blockchain to function as a transaction processing platform rather than just a database. Since then, blockchain has garnered widespread attention across various do-

mains, with many companies attempting to develop blockchain-based solutions. In the 3.0+ era, customized enterprise blockchain technologies have received significant attention, with Hyperledger Fabric (HLF) being a representative project. HLF is a popular permissioned blockchain platform with vast commercial application prospects [7]. Due to privacy considerations, HLF has been identified as the optimal blockchain solution for building applications [8]. Additionally, as a private and permissioned architecture, HLF is more energy-efficient, easier to construct, and boasts good transaction throughput (the ability to process an increasing number of transactions simultaneously) compared to public blockchain systems [9]. It serves as a technology that accumulates data and compiles them into multiple blockchains [10], focusing on secure data exchange among multiple distributed applications [11]. Consequently, the HLF architecture is widely utilized in property rights transaction applications. Regarding the application of blockchain in property rights transactions, Duane et al. [12] designed the overall structure and process specification of smart contracts and formulated standardized specifications for intellectual property smart contracts. The transaction scenario addresses the problem of the current chaotic structure of smart contracts in intellectual property transactions, facilitating collaborative development among scholars in various fields. Sladic Goran et al. [13] proposed a system architecture for protecting and trading real estate. The system tracks transactions in the Land Information System (LIS) in an immutable and tamper-proof manner to enhance system security, thereby improving transaction speed, efficiency, and data integrity. Rodrigo M et al. [14] chose Hyperledger Fabric as the most suitable blockchain platform and implemented a prototype of a blockchain solution for the real estate industry based on a real estate transaction use case model. Addressing the issue of double fraud in real estate transactions, Mashatan et al. [15] proposed a prototype solution based on Hyperledger Fabric and Sails. They demonstrated the results of agent-based modeling and simulation, validating the inherent transparency of the proposed design and providing optimal allocation for buyers and sellers. Yadav et al. [16] proposed a scalable and innovative property/land registration framework based on blockchain technology utilizing the Interplanetary File System (IPFS), a peer-to-peer (P2P) distributed network. It enables efficient, decentralized, and transparent data sharing and storage [17]. Ahmad et al. [18] presented a system capable of recording real estate transactions on a private blockchain using smart contracts. The immutability of the blockchain ledger and transactions provides a secure space for real estate business. Song et al. [19] introduced a new blockchain consensus mechanism based on participant contribution. The proposed consensus mechanism, called Proof of Contribution (PoC), quantifies user behavior and operations in blockchain applications into contribution values computed through algorithms. In each consensus round, the node with the highest contribution value gains the right to generate the next new block. Zhao et al. [20] proposed a blockchain-based digital rights management scheme (BMDRM), realizing distributed digital rights management and authorization systems through the introduction of non-fungible tokens (NFTs) and smart contracts. Hasan et al. [21] proposed a DRM-distributed media transaction framework based on digital watermarking and a scalable blockchain model. Yan et al. [22] employed an alliance chain and the practical Byzantine fault tolerance (PBFT) consensus mechanism to support smart contracts and applications, proposing a blockchain-based trading platform for agricultural water rights. Chen et al. [23] introduced a blockchain-based anti-counterfeit and traceable NBA digital trading card management system, utilizing blockchain technology to safeguard digital trading cards and incorporating unique digital copyrights. This approach reduces reliance on external regulatory bodies, enhancing the system's ability to combat counterfeit cards and ensuring the security of the digital trading card market. Thien H. T. et al. [24] discussed the technical challenges of the metaverse and then emphasized how blockchain can help address these challenges. Additionally, they explored the impact of blockchain on key-enabling technologies in the metaverse, including the Internet of Things, digital twins, multi-sensory and immersive applications, artificial intelligence, and big data. They also introduced some major projects to showcase the role of blockchain in metaverse applications

and services. Finally, they presented some promising directions to drive further research innovations and developments in the use of blockchain in the metaverse in the future. Wang X.J. et al. [25] proposed blockchain intelligence as a key to integrate blockchain and ML, combining the advantages of both to drive the rapid development of the Internet of Vehicles (IoV). They discussed general frameworks, issues, requirements, and advantages for the implementation of blockchain intelligence in the IoV. In terms of implementing blockchain intelligence in the IoV, driven by its advantages, they summarized solutions in four aspects, including reliable interactions, network security and data privacy, a trustworthy environment, and scalability. Peng J. et al. [26] systematically provided an overview of various applications of blockchain technology in the waste management industry and further discussed related challenges and opportunities by considering the promising prospect of combining blockchain technology with IoT, artificial intelligence (AI), and life cycle assessment (LCA). They also offered insights into emerging applications of blockchain in waste management and clarified research paths in the context of blockchain, digitalized waste management, and the circular economy. Li K. P. et al. [27] introduced the major blockchain platforms currently used in food supply chains and conducted a comprehensive analysis, exploring the benefits and challenges of blockchain technology in the food industry. They demonstrated that blockchain enables unprecedented visibility at each step of the food supply chain, helps increase transaction transparency, food safety, and quality, and reduces food fraud and waste. Milad B. et al. [28] identified notable publications, conferences, significant authors, nations, organizations, and funding organizations. Their research indicates that the primary research topics include blockchain in the construction industry, supply chain management, smart contracts, sustainability, building information modeling (BIM), the Internet of Things (IoT), and energy efficiency. Additionally, they suggested potential research areas, such as the use of blockchain in the circular economy, risk management, smart villages, and infrastructure construction projects.

In the realm of property rights transactions, it is evident that researchers have made significant contributions to the development of smart contracts and the standardization of property transactions. Furthermore, various frameworks and solutions have been proposed to ensure data security and privacy. Typically, these studies have concentrated their efforts within specific domains. Nevertheless, the distinguishing feature of blockchain technology lies in its remarkable versatility across a multitude of domains, thereby harboring the potential for cross-disciplinary applications.

Moreover, these endeavors have uniformly adopted the mature blockchain platform known as Hyperledger Fabric, a platform that has already undergone rigorous validation in numerous practical applications. This ensures the feasibility of our technological foundation and its practical deployability in real-world scenarios.

2.3. Consensus Algorithms

Consensus algorithms are a fundamental element of blockchain technology and have become a hot topic in distributed systems research in recent years [29]. Among them, the PBFT consensus algorithm, as a commonly used Byzantine fault-tolerant algorithm, has been widely applied in numerous blockchain platforms. However, the PBFT consensus algorithm still faces some challenges, such as performance degradation in scenarios with high network latency and a large number of nodes. To address these issues, a series of improved algorithms based on PBFT have emerged in recent years. Yao Sou et al. [30] proposed a more efficient consensus algorithm called SV-PBFT (Shapley Value-PBFT), which simplifies the consensus process, reduces communication overhead, and enhances consensus efficiency. In Riyad et al. [31], to address the challenges of high communication overhead and low algorithm efficiency, a practical Byzantine fault tolerance algorithm (S-PBFT) was proposed, tailored to the characteristics of consortium blockchains. This algorithm introduces a node-scoring mechanism, classifying nodes into consensus nodes, candidate nodes, and early-stage nodes. To enhance the reliability of consensus nodes, nodes adapt dynamically based on individual behavior. Notable improvements have been made: the

election process for controller nodes has been modified, utilizing node scores and behaviors as election criteria, thereby enhancing the stability of the algorithm. Jiangyao et al. [32] designed an efficient Byzantine fault-tolerant consensus mechanism (DE-BFT) for energy blockchains. It improves node election and main chain consensus. DE-BFT enhances the security and randomness of node selection using health scores and verifiable random functions. The protocol for efficient data consistency interactions reduces communication complexity between nodes from exponential to constant levels. Wei Liu et al. [33] proposed the QoS-aware Trust Practical Byzantine Fault Tolerance (QTPBFT) algorithm, which effectively achieves consensus, significantly reduces resource consumption , and enhances consensus efficiency. It incorporates a QoS-aware trust service global evaluation mechanism. WeiWei Fan et al. [34] introduced an optimized consensus algorithm that utilizes a scoring mechanism to select nodes participating in consensus, reducing communication overhead in the network. The algorithm also improves the consistency protocol for PBFT when there are no Byzantine nodes and introduces an upgrade mechanism to dynamically update the set of nodes participating in the consensus to ensure that the optimized consistency protocol is executed most of the time. Xianan Liu et al. [35] presented an optimized Credit Byzantine Fault Tolerance (CBFT) algorithm based on grouping and credit grading to optimize the node structure in large-scale consortium blockchains. Meng Wutong et al. [36] proposed a non-interactive and verifiable random endorsement node optimization scheme. Using the "endorsement–sort–verification" model in the Hyperledger Fabric consensus, it introduced an endorsement node candidate set and used a verifiable random function for random selection, achieving non-interactive and verifiable random endorsement. Ren Xiyu et al. [37] introduced the Trust-based Practical Byzantine Fault Tolerance (T-PBFT) algorithm based on a trust evaluation model. It employed clustering to group consensus nodes, reducing communication complexity. Nodes were then selected as main nodes in consensus groups based on trust evaluations of their historical behavior, ensuring the consistency of data information stored by consensus nodes in the system. Zekun Liu et al. [38] proposed the DT-PBFT algorithm based on dynamic mechanisms and credit scoring. It introduced dynamic joining or exiting mechanisms, enabling nodes to freely join or exit the cluster as needed. Additionally, it added a credit-scoring mechanism and divided nodes into layers based on trust levels. The use of a punishment mechanism reduced the likelihood of continuous malicious behavior by nodes and ensured the selection of the best main node from the backup main node layer, significantly improving consensus efficiency. Wang Zhong et al. [39] presented a secure and efficient blockchain-distributed consensus algorithm, ST-PBFT (Shard Transaction Practical Byzantine Fault Tolerance), applied to the IP transaction scenario.

In summary, blockchain technology, coupled with the PBFT consensus algorithm, offers comprehensive technical support for property rights transactions, ensuring transaction efficiency, security, transparency, and traceability. When both buyers and sellers engage in a transaction, the blockchain system can record and synchronize transaction data, guaranteeing data integrity, preventing loss or tampering, and rendering the data accessible to all participants within the same blockchain network. In the event of disputes, all stakeholders, including buyers, sellers, and government entities, can rely on the data stored on the blockchain for accountability.

Therefore, within the context of rural property rights transactions, the integration of blockchain technology establishes a framework for a blockchain-based rural property rights trading system. Leveraging the inherent advantages of decentralization, data immutability, transparency, and trustworthiness inherent in blockchain technology, this system assumes a pivotal role in authenticating rural property rights, facilitating identity verification, and storing transaction data. Consequently, it enhances the security and convenience of rural property rights transactions while addressing issues related to information asymmetry between transacting parties.

Furthermore, the introduction of an enhanced PBFT consensus algorithm, incorporating dynamic mechanisms and a dual-scoring model encompassing credit and activation,

optimizes the consensus process. This leads to improved consensus efficiency, enhanced information throughput, and a substantial reduction in transaction response times.

3. Framework for a Blockchain-Based Rural Property Rights Trading System

In this section, we present a blockchain-based framework designed for rural property rights trading. The framework offers an extensive examination of the architecture of the blockchain system, the involved participants, and the flow of information within the blockchain. Furthermore, the paper outlines the development of pertinent smart contracts aimed at streamlining the rural property rights trading system framework.

3.1. Overall Conceptual Framework

The conceptual framework, as proposed, is depicted in Figure 1, comprising four key components: the blockchain network, buyer users, seller users, and government institutions. Government institutions play a supervisory role in monitoring the transactional engagements of both buyers and sellers. Transaction records stemming from these interactions are gathered and subsequently uploaded onto the decentralized blockchain network, as opposed to relying on a centralized database or server. Within this blockchain system, data are made accessible to all network nodes and maintain immutability. Consequently, participants can place their trust in the data and utilize on-chain information to validate compliance.

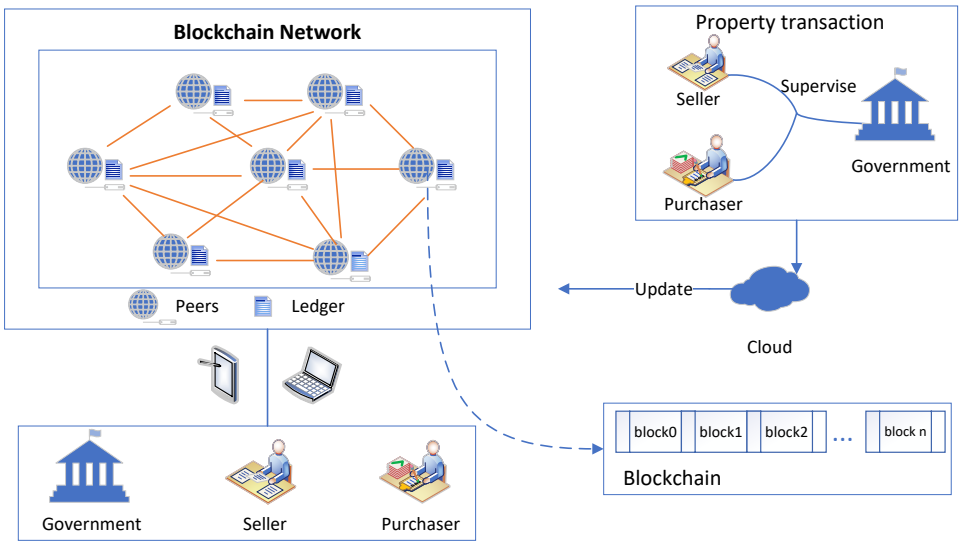

Figure 1. Overall Conceptual Framework.

3.2. Hyperledger Fabric

In this research, we have adopted Hyperledger Fabric as the fundamental architecture for our system, an open-source blockchain platform recognized for its modularity and scalability, designed particularly for permissioned blockchain setups. The choice of Hyperledger Fabric is underpinned by its array of merits. To begin with, it effectively meets the consensus demands of blockchain within a distributed context, guaranteeing data security and reliability. In contrast to permissionless blockchain systems, Hyperledger Fabric excels in preserving the confidentiality of transaction participants and demonstrates substantial benefits in terms of energy efficiency and implementation simplicity.

Hyperledger Fabric, categorized as a consortium blockchain, effectively implements authorization mechanisms through Member Service Providers (MSP) and Fabric-Certificate Authority (Fabric-CA). MSP, a modular component within Hyperledger Fabric, offers identity verification for all entities involved in the blockchain network. Conversely, Fabric-CA oversees member certificates, overseeing member qualifications such as registration, addition, and revocation. In our blockchain-based rural property rights transaction framework,

all participants are mandated to submit registration requests through clients to become part of the blockchain network and obtain two types of certificates (E-Cert and T-Cert).

Moreover, Hyperledger Fabric (HLF) demonstrates characteristics of openness and high customizability, affording us the flexibility to design customized solutions that align with the specific needs of rural property rights transactions. It also incorporates meticulous permission control mechanisms, ensuring that only authorized participants can access and engage in transactions. Addressing the demands of large-scale rural property rights transactions, HLF places particular emphasis on performance and scalability, supporting distributed smart contract execution to enhance transaction efficiency. For property transactions involving sensitive information, HLF provides robust privacy protection mechanisms, disclosing private data only when necessary.

Nonetheless, it is crucial to acknowledge that HLF does come with certain limitations. It exhibits a relatively steep learning curve and complexity, which may necessitate a significant investment of time and effort for proficiency and deployment. Additionally, since HLF operates as a consortium blockchain framework, establishing and maintaining trust relationships among participants may require additional resources and patience.

In conclusion, our decision to adopt Hyperledger Fabric as the blockchain framework for our study is based on its exceptional performance in meeting our research needs. However, when selecting a blockchain framework, a comprehensive evaluation of various factors must be conducted, with the choice tailored to the specific context. In summary, HLF provides us with a robust and adaptable foundation for our research, driving innovation and advancement in the field of rural property rights transactions.

3.3. Identity Management and Access Control

In the context of our system framework, identity management and access control stand as critical components to ensure that only authorized participants can engage with blockchain networks. This paper elucidates the workflow of identity management and access control within this framework, outlining the steps involved in a manner reminiscent of the writing style found in Nature journal articles.

1. Identity registration:
   New participants are required to undergo identity registration as the first step in gaining access to the system. They must provide personal identity information and relevant credentials to substantiate their identity. This information is submitted to the system and subsequently subjected to verification.
2. Transaction contracts: Smart contracts can be written and executed to define the terms and conditions of transactions, automating the transaction process and ensuring its security and reliability. The contract can specify the rights of both buyers and sellers, prices, and payment methods, and automatically release funds or transfer property based on predefined conditions;
3. Identity verification:
   Upon submission of information, the system initiates identity verification procedures. This may encompass the validation of identity documents, credit reports, or other pertinent documents. Once identity verification is successfully completed, participants are granted access permissions to the system.
4. Issuance of identity certificates:
   Following successful identity verification, the system issues identity certificates to participants. These certificates contain the participant's identity information and access privileges.
5. Access control:
   Once participants possess identity certificates, they can attempt to access the system. The system examines the certificates to ensure that the holder possesses the requisite permissions for the requested operations. If permissions are insufficient, access is denied.
6. Smart contract execution:

Access control extends to the execution of smart contracts on the blockchain network. Only participants with appropriate permissions are allowed to execute specific smart contracts.

7. Auditing and monitoring:
   The system necessitates real-time monitoring of participant activities to ensure that they do not exceed their authorized scope. Simultaneously, audit logs record all accesses and transactions for auditing and tracking purposes.

It is imperative to note that Hyperledger Fabric offers robust identity management and access control capabilities through pluggable identity verification and access control modules, allowing for flexible definition and management of participant permissions. These mechanisms contribute significantly to ensuring the security and compliance of the system, mitigating unauthorized access and operations.

3.4. On-Chain Transaction Information Flow

In a distributed environment, blockchain relies on a consensus mechanism to validate data. In this research, we have adopted Hyperledger Fabric (HLF) as the underlying architecture. HLF is an open-source system known for its modularity and scalability, specifically designed for the deployment and operation of permissioned blockchains. It effectively addresses privacy concerns of transaction participants and offers superior energy efficiency and implementation ease when compared to permissionless blockchain architectures [40].

Fabric falls under the category of consortium blockchains and implements two crucial components for authorization: Member Service Providers (MSP) and Fabric-Certificate Authority (Fabric-CA). MSP, which is an integral modular part of Hyperledger Fabric (HLF), serves the purpose of providing identities to all participating peers within the blockchain network. On the other hand, Fabric-CA manages membership by handling member certificates, which include functions such as registration, addition, and revocation. In the framework of the blockchain-based rural property rights trading system, all participants are required to submit registration requests through the client to gain access to the blockchain network. Subsequently, participants are provided with two types of certificates, namely E-Cert and T-Cert.

In the context of blockchain transactions, data from both buyers and sellers are submitted to the client, which in turn forwards the signed transaction to endorsing nodes. These endorsing nodes conduct simulations to validate the transaction proposal. The endorsing nodes generate the transaction result, comprising response values, read sets, and write sets, if the following conditions are met: (1) the correct information format; (2) not previously submitted; (3) a valid signature using MSP; and (4) the submitter possesses the appropriate authorization to execute the proposed operation on the channel. Channels serve the purpose of partitioning the state of the blockchain network. Once the client accumulates a sufficient number of responses from the endorsing nodes, the transaction, along with the endorsing transaction proposal responses, is submitted to the ordering nodes, referred to as ordering service nodes. The orderer node is responsible for creating transaction blocks and sequentially distributing them to all users. Initially, the orderer node distributes the block to all peers connected to it. Within the channel, each node verifies every transaction in the block to ensure that signatures, endorsements, and endorsement policies align with the specified requirements. Invalid transactions are retained within the block but are marked as invalid, and the ledger's state remains unaltered. This meticulous process ensures the efficient processing of transactions on the blockchain while upholding the consistency of the entire blockchain.

3.5. Smart Contracts

Smart contracts [41,42], also known as chain codes, are program codes that offer a series of advantages such as high security, strong reliability, fairness, and efficiency [43]. They can automatically execute specific tasks in a blockchain network, including the execution of complex logic and data recording [44]. Once deployed on the blockchain, the smart

contract's code is permanently stored and accessible for execution by nodes in the network. This feature makes smart contracts a core functionality of blockchain, providing automation and programmability for various application scenarios.

Smart contracts in blockchain can achieve automation through the following ways:

- Condition triggering;
- Automated execution;
- Time triggering;
- Event listening;
- External invocation.

These automated functions empower smart contracts to execute automated business procedures and tasks on the blockchain, thereby enhancing efficiency, mitigating human errors, and guaranteeing the reliability and uniformity of contracts. In our study, we have formulated five distinct categories of smart contracts tailored for rural property rights: property rights certifications, transaction contracts, identity verification, access control, transaction auditing and supervision, and transaction anomaly alerts, as depicted in Figure 2.

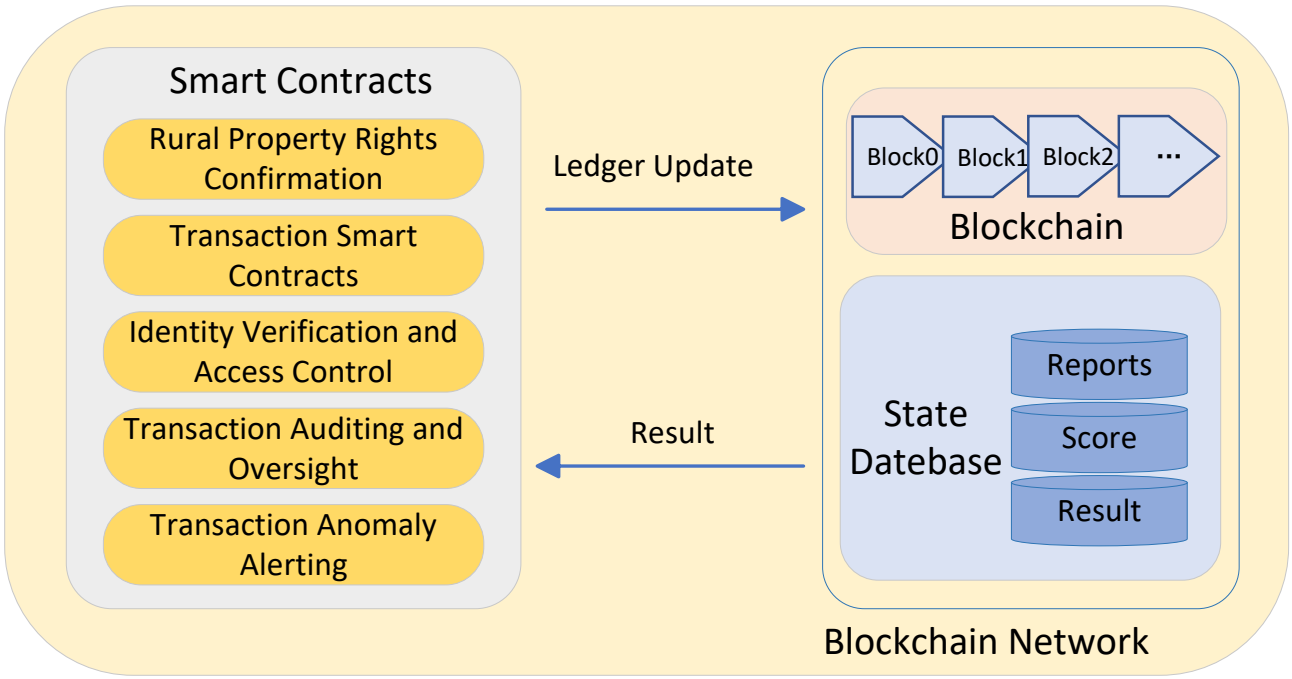

Figure 2. Smart Contracts for Rural Property Rights Transactions on the Blockchain.

1. Rural property rights certification:
   Smart contracts are employed to record and certify rural property rights information, storing ownership information for farmland, houses, or other assets on the blockchain in a manner that ensures the authenticity and traceability of property rights.
   - Design principle:
     This contract is designed to record and certify rural property rights information, securely storing ownership details on the blockchain to ensure the legitimacy and security of property rights.
   - Problem solved:
     Storing property rights information on an immutable blockchain resolves trust issues, ensuring the legality and security of property rights.

2. Transaction contracts:
   Smart contracts are used to compose and execute transaction contracts, specifying transaction conditions and terms, automating the transaction process to ensure its security and reliability. These contracts define the interests of both buyer and seller, prices, payment methods, and more, releasing funds or transferring ownership automatically based on conditions.
   - Design principle:
     This contract is designed to compose and execute transaction contracts, defining transaction conditions and terms and automating the transaction process to ensure its security and reliability.
   - Problem solved:
     Automatic execution of transaction contracts enhances transparency and reduces the risk of contract disputes.
3. Identity verification and access control:
   Smart contracts integrate identity verification mechanisms to ensure the legitimacy and compliance of participants. They validate participant identities and enforce permissions and access control, allowing only qualified participants to engage in property transaction operations.
   - Design principle:
     This contract integrates identity verification mechanisms, ensuring participant identities are legitimate and compliant with predefined criteria, allowing only qualified participants to engage in property transaction operations.
   - Problem solved:
     Identity verification and access control ensure that only legitimate and authorized participants conduct transactions, enhancing transaction security.
4. Transaction auditing and regulation:
   Smart contracts record all operations and changes in property transactions, providing transparent and traceable transaction records. These serve as effective auditing tools for regulatory authorities, ensuring transaction compliance and regulatory requirements.
   - Design principle:
     This contract records all operations and changes in property transactions, providing transparent and traceable transaction records, serving as auditing tools for regulatory authorities.
   - Problem solved:
     Providing auditable transaction records ensures transaction compliance and assists regulatory authorities in fulfilling their oversight responsibilities.
5. Transaction anomaly alerts:
   By setting specific conditions and rules within smart contracts, transaction anomalies can be monitored, triggering alert mechanisms when suspicious transactions, unusual amounts, or contract violations occur. Additionally, a predefined evaluation index and weight calculations for performance data are applied. The results are used as the output for evaluating the contract. Furthermore, a user interface is provided for users to query and view the credibility of each participant, enhancing transparency and mitigating potential fraud risks.
   - Design principle:
     This contract monitors transaction anomalies and triggers alert mechanisms for suspicious behavior, unusual amounts, or contract violations.
   - Problem solved:
     Automatic detection and alerting of anomalies enhance transaction security and reliability while reducing potential fraud risks.

The integration of these smart contracts enhances the safety, efficiency, and transparency of rural property rights transactions in China. Each contract is tailored to address

specific objectives, collectively forming a comprehensive system framework to tackle the challenges faced in rural property transactions in Chinese rural areas.

4. Enhanced PBFT Consensus Algorithm (CA-PBFT) Based on a Dual Rating System of Credit and Activation

In a blockchain network, all nodes are expected to maintain identical copies of the blockchain data to ensure data consistency. To achieve such consistency, a consensus mechanism is required. Different consensus mechanisms have been designed for various systems and purposes, including PoW (Proof of Work) [4], PoS (Proof of Stake) [45], and PBFT [46], among others. Selecting an appropriate consensus algorithm to meet the specific requirements of different blockchain applications is crucial, and relevant recommendations and research can be found in References [47,48]. In this study, for the rural property rights trading framework based on blockchain, we have opted for the PBFT consensus algorithm and conducted optimizations to enhance system performance and security.

4.1. PBFT

PBFT is a widely applied general solution in distributed systems, primarily aimed at ensuring system consistency. The PBFT algorithm has achieved the remarkable feat of reducing the exponential communication complexity associated with Byzantine fault tolerance algorithms to polynomial levels [49], paving the way for the practical application of Byzantine fault tolerance algorithms in engineering. This algorithm effectively addresses the issue of malicious nodes disseminating erroneous information to disrupt the normal operation of a system. In comparison to other consensus algorithms, PBFT offers significant advantages in terms of security, fault tolerance, and communication complexity. For instance, when contrasted with the Proof of Work (PoW) algorithm, PBFT stands out by not requiring extensive computational power and power consumption, making it both environmentally friendly and efficient. Moreover, in comparison to the Proof of Stake (PoS) algorithm, PBFT excels in providing heightened security and fault tolerance, particularly in the context of Byzantine fault tolerance. Additionally, when compared to other Byzantine fault tolerance algorithms, PBFT typically boasts lower communication complexity and superior performance.

1. Security and fault tolerance:
   PBFT is renowned for its outstanding security and fault tolerance capabilities. In applications demanding a high level of security and trustworthiness, such as rural property rights transaction platforms, opting for PBFT helps ensure transaction security and data integrity. PBFT can tolerate up to 1/3 of malicious nodes, a critical feature for thwarting potential attacks and ensuring system stability;
2. Communication complexity:
   PBFT has successfully reduced the communication complexity of Byzantine fault tolerance algorithms from exponential to polynomial levels. This efficiency gain is particularly vital in rural areas where network connectivity may be weak. PBFT's low communication complexity guarantees the efficiency of both transaction and consensus processes, contributing to enhanced transaction speed and efficiency;
3. Consensus protocol:
   PBFT's consensus protocol [50] ensures consistency of state across all nodes, a crucial factor for ensuring transaction accuracy and traceability. In platforms dealing with numerous property transactions and data records, such as rural property rights transaction platforms, a reliable consensus protocol is essential for managing these transactions;
4. Real-world applications and maturity:
   PBFT has been widely adopted and boasts mature implementations and ecosystems in real-world applications. In critical domains such as rural property rights transactions, selecting a validated consensus algorithm is paramount due to its proven reliability and effectiveness across various scenarios.

Despite PBFT's commendable performance in multiple aspects, it does have some drawbacks. One of these relates to the selection of primary nodes, which may result in malicious nodes consecutively becoming primary nodes, leading to the wastage of network resources. Additionally, executing three-round broadcast communications incurs higher network communication overhead. More importantly, PBFT exhibits limited dynamism, meaning nodes cannot dynamically join or exit the cluster, thus lacking flexibility in specific application scenarios.

4.2. CA-PBFT

The CA-PBFT algorithm is an extension of PBFT that introduces dynamic mechanisms, a dual-scoring system based on credit and enthusiasm, and optimizations for protocol consistency. This innovative approach effectively tackles the shortcomings of PBFT, including limited dynamism, inflexibility, and significant communication overhead delays.

4.2.1. Dynamic Mechanism

The fundamental objective of the dynamic mechanism is to enable nodes to join or leave the network as needed, responding to changing requirements and conditions. This mechanism ensures the system's adaptability and resilience by allowing the smooth incorporation of new nodes and automatic reconfiguration of the network when nodes depart. This enhanced flexibility contributes to system stability, responsiveness, and overall efficiency, enabling it to effectively handle evolving network scenarios and fluctuations in node participation dynamics. We draw inspiration from the approach outlined in Reference [38].

(1) Dynamic Node Addition

The process of dynamically adding a node can be summarized as follows:

1. Application phase:
   When a new node is introduced into the active network and intends to join the cluster for subsequent consensus phases, it initiates an AddNode request by dispatching a message s to all existing nodes within the cluster, along with an appended timestamp.
2. New node authentication phase:
   Upon receiving the AddNode request from the newly introduced node, existing nodes broadcast AgreeAdd messages and collect these messages from other nodes. When a node accumulates 2f+1 AgreeAdd messages, it sends an authentication message concurring with the integration of the new node into the cluster. Subsequently, when the new node collects 2f+1 authentication messages, a consensus is reached, granting permission for the new node to be incorporated into the cluster.
3. Data synchronization phase:
   The new node enters the proactive recovery process, sending data synchronization requests and receiving all currently stored information from other nodes to achieve data synchronization.
4. Network integration phase:
   After completing data synchronization, the new node broadcasts the JoinNet request to all nodes within the entire blockchain network, requesting participation in the consensus process of the blockchain network. Upon receiving the JoinNet request from the newly introduced node, all existing nodes simultaneously inform each other of the new node's formal network entry. Concurrently, a reevaluation of the total node count within the network is conducted, and a recalibration of the new view, denoted as v, is initiated.
5. Receipt phase:
   The primary node issues the UpdateNet information to all nodes within the cluster. Upon receiving this message, all consensus nodes execute an update for the total node count, denoted as N, and the view, denoted as v, within the blockchain cluster. This concludes the procedure for adding the new node. Following the successful

completion of view and node count updates, consensus nodes provide feedback to the primary node. Upon receiving 2f+1 such acknowledgments, indicating the successful integration of the new node into the network, a dynamic node addition consensus event is accomplished.

The process of dynamic node inclusion is illustrated in Figure 3, where the newly added node is denoted as New_Replica5.

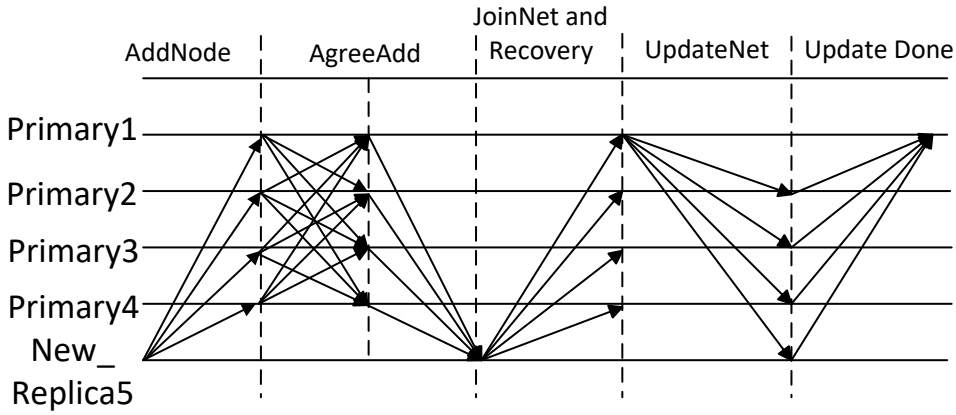

Figure 3. New Node Completes Data Synchronization.

(2) Data Synchronization for Dynamic Node Addition

1. Data request: Upon formal network authentication, the node New_Replica5 initiates data synchronization by requesting all existing nodes provide their current data snapshots;
2. Data response:
Existing nodes respond to these requests by commencing the transmission of their blockchain data to the new node. This dataset may encompass complete block replicas, transaction histories, and other vital network information;
3. Data verification:
Upon receiving the data, New_Replica5 conducts a comprehensive verification process. This verification encompasses checking block continuity, transaction legitimacy, and potential digital signatures;
4. Data completion:
Once all data have been successfully synchronized, New_Replica5 broadcasts a message to notify the network of its attainment of a data version consistent with that of other nodes.

(3) Dynamic Node Departure

The Node Dynamic Exit Process can be summarized as follows:

1. Application phase:
When the node labeled as Del_Replica5 initiates a voluntary exit, it begins broadcasting Del_request messages to other nodes.
2. Authentication message phase:
Upon receiving the Del_request message from the exiting node (Del_Replica5), and assuming that this node is exiting the existing blockchain network, other nodes calculate the new view v and the total node count N after removing the exiting node. They broadcast their agreement to delete the Del_Replica5 node to the rest of the blockchain network. When f *AgreeDel* messages are collected, all nodes consent to and execute the request to remove the node, which includes data synchronization. Additionally, they encapsulate messages containing the updated view v and total node count N after the node's deletion.
3. Exit phase:
After the node Del_Replica5 has exited, the primary node broadcasts an UpdataNet message.

4. Network update:
   Upon receiving the UpdataNet message, all nodes within the network update the total node count N and the view v in the blockchain network. This finalizes the removal process of the mentioned node.

During this process, nodes achieve consensus and synchronization through message broadcasting and mutual confirmation, ensuring the secure departure of nodes from the blockchain network. The procedure for dynamic node exit is illustrated in Figure 4, where Del_Replica5 represents the node initiating the exit request.

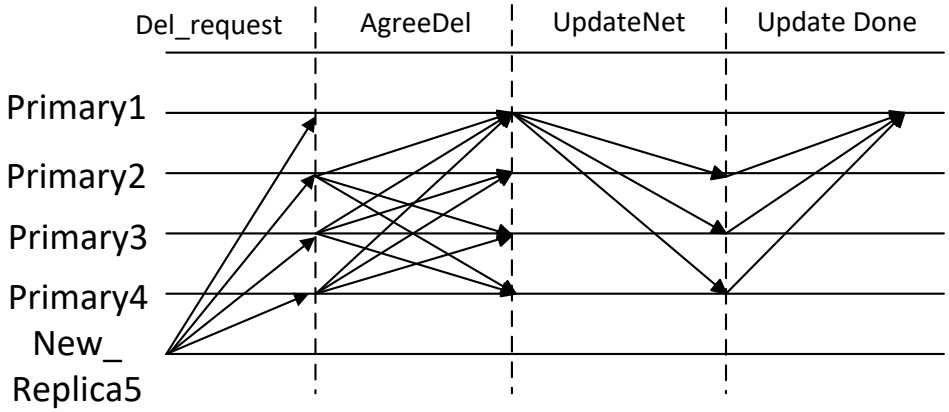

Figure 4. Node Dynamic Exit Process.

(4) Data Synchronization for Dynamic Node Departure

1. Data request: Before Del_Replica5 decides to depart, it may need to ensure that all pending transactions or incomplete data synchronizations have been successfully completed or taken over by other nodes in the network;
2. Data transfer:
   In certain scenarios, Del_Replica5 may possess data that are not present in other nodes. In such cases, it becomes imperative for Del_Replica5 to transfer these unique data to other nodes within the network;
3. Data confirmation:
   Once all data have been effectively transferred, Del_Replica5 broadcasts a message to notify the network that it can safely depart without causing data loss or inconsistencies;
4. Data completion:
   Del_Replica5 formally disconnects from the network after ensuring that its data have been fully synchronized with the network.

4.2.2. The Dual-Scoring Mechanism with Credit and Activation

In the PBFT algorithm, the sequential selection of primary nodes introduces a potential risk of selecting anomalous nodes as primaries, which could compromise the security and stability of the system. However, the enhanced CA-PBFT consensus algorithm introduces a dual-scoring mechanism to assess the status of nodes, categorize consensus nodes in the network, and determine primary nodes accordingly. Nodes with higher credit and enthusiasm levels are considered more secure and stable, reducing the likelihood of initiating view-change protocols. This reinforces the security of primary node selection, enhances consensus efficiency, promotes transactional fairness, and ensures the effectiveness of the consensus.

We propose a dual-score mechanism that incorporates both credit and activation, classifying nodes into two dimensions for scoring: credit and enthusiasm. The scoring ranges for both dimensions are set within [0, 100], with initial values uniformly set at 30. Based on their respective scores, nodes participating in the CA-PBFT consensus are grouped into four categories within the credit-scoring dimension: priority nodes ([100, 75]) are eligible to become primary nodes and actively participate in the consensus process;

candidate nodes ((75, 50]) can only function as sub-nodes within the consensus and are ineligible for primary node roles; regular nodes ((50, 25]) do not partake in the consensus process but receive consensus results; and malicious nodes ((25, 0]) neither participate in consensus nor accept consensus results and are subsequently removed.

In the activation score dimension, nodes are divided into three categories: priority nodes ([75, 100]) are eligible to become primary nodes and actively participate in the consensus process; Candidate nodes ([50, 75)) can only serve as sub-nodes within the consensus and are ineligible for primary node roles; and regular nodes ([0, 50)) do not engage in the consensus process but solely receive consensus results.

Ultimately, the two dimensions are overlaid, resulting in creditworthiness and activation levels falling within the ranges [75, 100] for priority nodes, [50, 75) for candidate nodes, [25, 50) for regular nodes, and [0, 25) for malicious nodes, as illustrated in Figure 5.

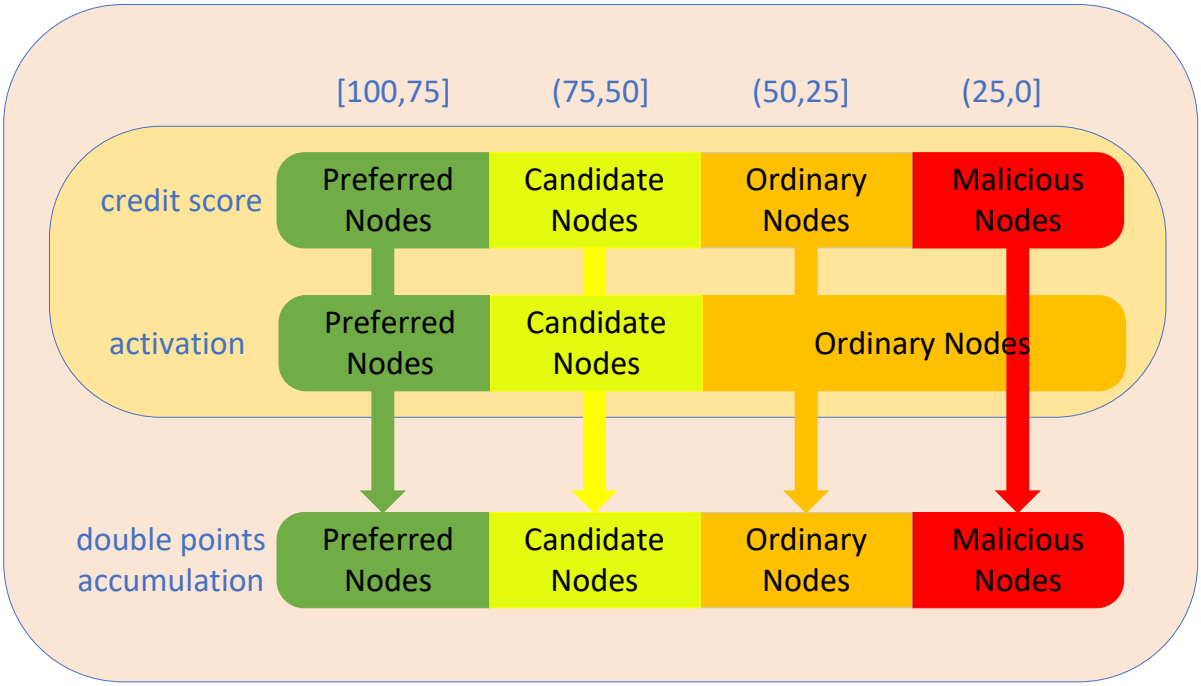

Figure 5. Node Classification Diagram under the Dual-Scoring Mechanism.

(1) Credit Score

For the credit-scoring section, we referred to Reference [35].

Definition 1. *Delay Index refers to the delay in various message response processes, expressed as:*

$$D(i) = [1 - \frac{d_{ij}}{d_{max}}] \quad (1)$$

d_ij represents the delay of the jth transaction of node i; d_max represents the maximum allowable delay for transactions, and if it exceeds the maximum delay, it indicates that the node failed to execute the transaction.

Definition 2. *Transaction completion rate with penalty mechanism refers to the proportion of successful participation in various transactions by nodes after entering the network, expressed as:*

$$T(i) = \frac{100}{n} \sum_{i=1}^{m} \mu_i \quad (2)$$

n represents the total number of transactions in the system, and m denotes the number of transactions completed by node i. μ is the transaction success indicator, where μ equals 1 for successful

transactions and −1 for failed transactions. This formulation takes into account both the positive impact of successfully completed transactions on nodes and the adverse effects of transaction failures on normal operations, thus providing a more accurate differentiation of node credit values.

Definition 3. *The formula for calculating the final credit-scoring of a node is as follows:*

$$C(i) = \frac{1}{2}(xD(i) + yT(i)) \tag{3}$$

x represents the weight of node transaction latency, and y represents the weight of the node's own completed transactions, with x + y = 100. The credit model intuitively reflects the node's performance in consensus. If a node has low latency, a high transaction completion rate, and a good historical credit value, its credibility is high. Conversely, if a node has high latency, a low transaction completion rate, and poor credit value, its credibility is low.

(2) Activation

The activation of node i is calculated by selecting the transaction frequency index T(i), transaction amount index M(i), and distance index to the Active Transaction Region D(i).

Definition 4. *Distance Index from the node to the Active Transaction Region, expressed as:*

$$D = min_{\sqrt{(x_i-x)^2+(y_i-y)^2}} \left| (x, y) \in G \right. \tag{4}$$

Representing the node locations, dividing the Active Transaction Region into grid cells, and denoting each grid cell as cell, the distance between a node and the Active Transaction Region is expressed as distance, where all grid cells represents the entirety of grid cells.

$$D(i) = \frac{max_D - D}{max_D} \tag{5}$$

max_D is the maximum distance to the Active Transaction Region among all nodes.

Definition 5. *The Transaction Frequency Index T(i) and Transaction Amount Index M(i) are expressed as follows:*

$$T(i) = \frac{T}{max_T} \tag{6}$$

$$M(i) = \frac{M}{max_M} \tag{7}$$

T and M, respectively, represent the transaction frequency and transaction amount of the node, while max_T and max_M represent the maximum transaction frequency and maximum transaction amount among all nodes.

Definition 6. *The final activation score of a node is calculated using the following formula:*

$$A(i) = xD_1(i) + yT(i) + zM(i) \tag{8}$$

w represents the weight assigned to the node's proximity to the transaction-active area, x represents the weight assigned to node transaction latency, y represents the weight assigned to the node transaction amount, z represents the weight assigned to the node transaction count, and the constraint w + x + y + z = 100 holds. The node's activation algorithm is tailored to suit the characteristics of the rural property rights trading platform, taking into account parameters such as transaction amount, transaction count, distance from the transaction-active area, and transaction latency. This approach aims to enhance consensus efficiency, whereby a node with low transaction latency, a substantial transaction amount, high transaction count, and proximity to the transaction-active area will exhibit high activation, while others will have lower activation.

4.2.3. Simplified Consensus Protocol

The PBFT algorithm, a consensus protocol extensively employed in distributed systems, entails intricate sorting and pairwise communication among nodes to maintain data consistency. However, in the context of the rural property rights trading market blockchain, the primary objective of consensus is to validate and confirm transaction outcomes, payment results, and similar aspects, obviating the necessity for sorting operations. Consequently, we have streamlined the PBFT algorithm to suit this specific scenario. The streamlined execution process of the consensus protocol is depicted in Figure 6.

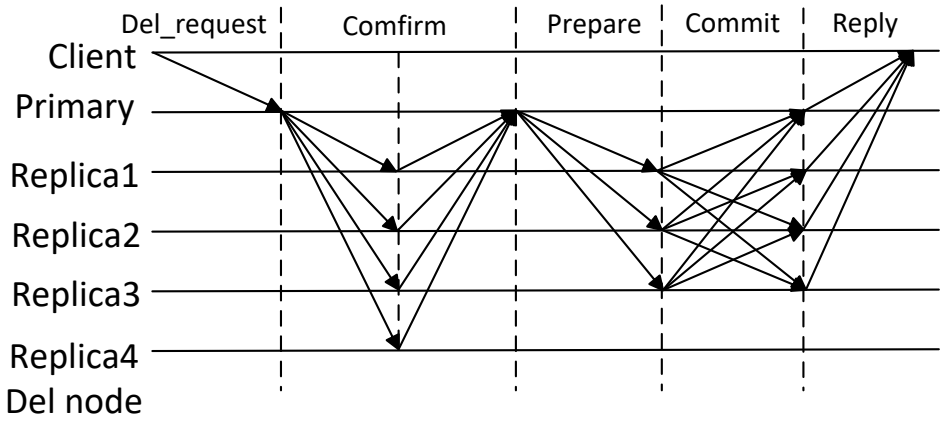

Figure 6. Simplified Consensus Protocol Execution Process.

In a Byzantine fault-tolerant system, where the maximum tolerable number of Byzantine nodes is represented as 'f', achieving consensus necessitates the selection of a minimum of 'R = 3f + 1' nodes to partake in the consensus process [51]. All nodes undergo an activation scoring process, and the top 'R' ranked nodes are designated as consensus nodes. Among these, the node ranked first assumes the role of the primary node, while the others serve as non-consensus nodes.

The consensus nodes execute a simplified consensus protocol, and in the presence of Byzantine nodes, they abort the ongoing simplified process and revert to executing the original PBFT protocol. Subsequently, a reselection of consensus nodes occurs. The consensus process encompasses several phases, including the request phase, pre-preparation phase, preparation phase, commit phase, and feedback phase.

During these phases, the primary node bears the responsibilities of data validation, data broadcasting, feedback collection from nodes, and the generation of the final data. Meanwhile, the non-primary nodes are tasked with receiving data and providing feedback regarding the validation results.

In the event of failures involving either the primary or non-primary nodes, the progress of the consensus process may be disrupted. Additionally, when the count of Byzantine nodes surpasses 'f,' the participation of all nodes within the entire network becomes necessary to ensure the consistency of results.

4.2.4. Upgrading the Consensus Node Set under the Dynamic Scoring Model

Ensuring the security and efficiency of the consensus algorithm hinges on the selection of appropriate consensus nodes. The CA-PBFT algorithm employed in this study utilizes a unique dynamic scoring model to dynamically update and select these consensus nodes. The following provides a detailed description of this process.

- Foundations of the dynamic scoring model
  The dynamic scoring model is a scoring mechanism that comprehensively considers the credibility and proactiveness of nodes. By assessing a node's performance and activity within the network, this model assigns a dynamically changing score to each node. Credibility primarily reflects a node's stability and reliability within the

consensus process, while proactiveness predominantly considers a node's activity and efficiency.

- The Process of upgrading the consensus node set
  1. Data collection:
     The system periodically collects relevant data from each node, including transaction latency, transaction completion rates, transaction amounts, transaction frequency, and the distance between nodes and active transaction areas;
  2. Scoring and classification:
     Based on the collected data and the dynamic scoring model, each node is assigned a score for credibility and proactiveness. Subsequently, nodes are categorized as malicious nodes, regular nodes, candidate nodes, or priority nodes based on these scores;
  3. Role determination:
     Node categorization determines their roles within the consensus. For instance, malicious nodes are excluded from the consensus, while priority nodes may become primary nodes and participate in the consensus process;
  4. Network broadcasting:
     Once the node set is updated, this information is broadcasted through the network to ensure that all nodes are aware of the latest consensus node set.
- Upgrade frequency:
  Currently, our system performs consensus node set upgrades every 24 h. This frequency is based on our initial research and testing and may be adjusted in the future based on actual network performance and requirements.

4.2.5. CA-PBFT Consensus Algorithm Implementation Process

The execution process of the CA-PBFT consensus algorithm unfolds as follows:

1. Initialization of nodes:
   Upon startup, newly joined nodes receive initial scores, and an exit process is initiated for nodes requesting to leave. Existing nodes' credit scores and activation are evaluated based on their historical performance, and nodes are categorized according to their score ranges.
2. Client submits transaction requests:
   When a transaction request is received, the system checks for the current primary node. If no primary node exists, the candidate node with the highest score is selected as the primary node. Priority nodes, if available, are given preference in primary node selection.
3. Primary node executes consensus task:
   The primary node takes charge of executing consensus tasks, which involve tasks such as assigning identifiers and processing request messages. Subsequently, it engages in executing a streamlined consensus protocol. During this phase, the primary node assesses and compares the status of participating nodes based on received feedback and messages.
4. Confirming Byzantine nodes:
   In the feedback phase of the simplified consensus protocol, the primary node verifies the information transmitted by consensus nodes, primarily by checking the consistency of the information's hash values. If the hash values of the information transmitted by all consensus nodes are identical, it indicates the absence of Byzantine nodes' interference in the current network, allowing the consensus process to proceed. The primary node records scores and consensus information, preparing for the next round of consensus. However, if inconsistencies in hash values are detected, signifying the presence of Byzantine nodes, the primary node immediately halts the execution of the simplified consensus protocol to prevent any disruption.
5. Complete PBFT consensus protocol:

If the presence of Byzantine nodes is confirmed, the primary node initiates the execution of the complete PBFT consensus protocol, involving all consensus nodes to ensure the safety and correctness of the consensus process.

6. Update scores after consensus:
   After completing the consensus, all nodes record the data generated during the process and calculate and update the credit score and activation of each node. Nodes with scores falling below the established threshold are excluded from the consensus group and cannot participate in future consensus processes.
7. Return to step one for the next round:
   The system returns to the initial step, awaiting new transaction requests and preparing for the next round of consensus.

The CA-PBFT consensus algorithm enhances the security and reliability of the consensus process through various mechanisms, including the evaluation of node credit scoring, activation, Byzantine node detection, and dynamic node exclusion. Furthermore, it simplifies the consensus protocol, reducing interaction information and thereby enhancing the efficiency and performance of consensus. Over time, specific nodes may emerge as central points of consensus power, indicating their reliability in the market—a desirable outcome.

In a consortium blockchain, centralized entities do not reap substantial benefits and may even face penalties or expulsion from the consortium if they fail to adhere to the rules and maintain system stability. This motivation encourages these entities to uphold the established rules. Figure 7 provides a visual representation of the execution flow of the CA-PBFT consensus algorithm.

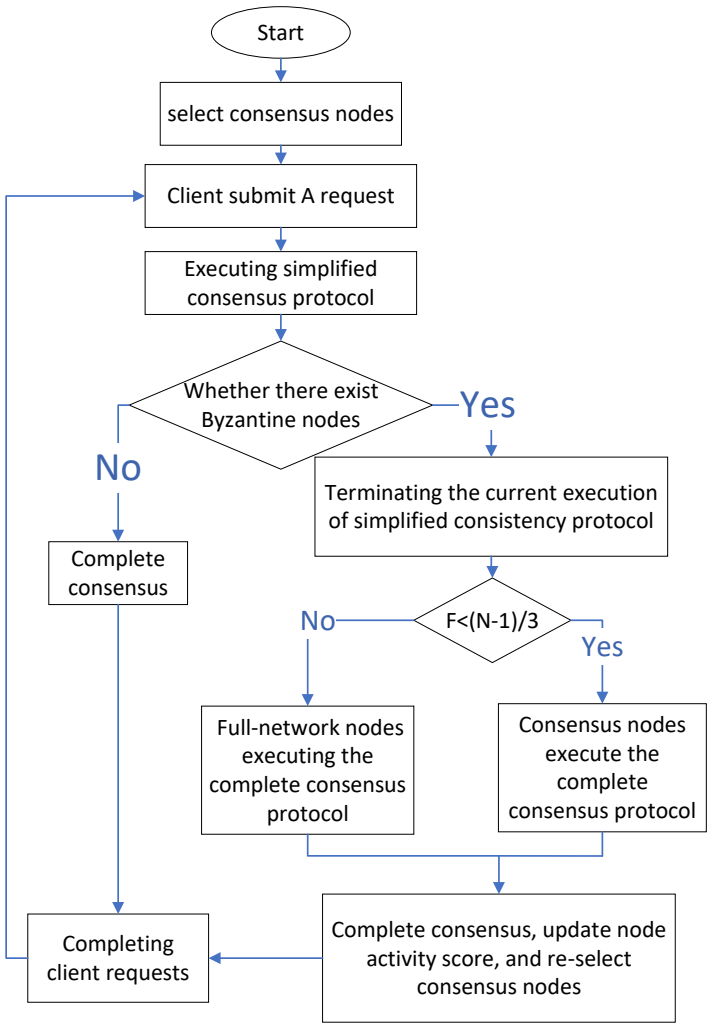

Figure 7. CA-PBFT consensus algorithm process.

4.2.6. Process Summary of the CA-PBFT Algorithm

The CA-PBFT algorithm integrates dynamic joining and exiting mechanisms, a dual-scoring mechanism based on creditworthiness and activation, and a streamlined consensus protocol to enhance the algorithm's performance and scalability. It effectively meets the requirements of a rural property rights transaction framework, including dynamicity, high performance, and security. Additionally, it aligns well with the demands for standardization, convenience, and liveliness in rural property rights transactions.

1. Dynamic mechanism:
   This mechanism's benefits lie in its ability to adapt to ever-changing circumstances, seamlessly incorporate new nodes, and maintain operational continuity. Its core principles emphasize flexibility, adaptability, stability, and efficient consensus, critical for sustaining the effectiveness and reliability of the blockchain network. Whether nodes exit due to failures, maintenance, or other reasons, the algorithm swiftly adjusts to changes and ensures the network's continuous operation. This flexibility and fault-tolerance enable the CA-PBFT algorithm to effectively handle continuously shifting network conditions and node join/exit scenarios. In the real world, network topology and node participation may change frequently, but the CA-PBFT algorithm can adapt flexibly, maintaining a high level of system stability and reliability. In summary, the dynamic exit and join mechanisms provide the CA-PBFT algorithm with increased adaptability and robustness.
2. Dual-scoring mechanism:
   Through the dual-scoring mechanism, which combines credit scoring and activation, nodes contributing significant value to the system are chosen to participate in the consensus process, while malicious nodes are identified and excluded. Nodes that successfully complete consensus tasks or offer valuable contributions receive higher scores, incentivizing active participation in the consensus process and promoting the provision of higher-quality services and contributions. By rewarding beneficial behavior, this mechanism stimulates nodes' enthusiasm and sense of responsibility, thereby enhancing the overall efficiency and stability of the system.
3. Simplified consensus protocol:
   In the CA-PBFT algorithm, a simplified consensus protocol has been specifically designed for scenarios without Byzantine nodes. Through optimizations in the algorithm and communication mechanisms, unnecessary communication frequency and data transmission volume between nodes are reduced, effectively lowering network load and latency. This streamlined consensus protocol, focused on consistency, enhances overall system performance, reduces latency, conserves resources, and reinforces system stability, ultimately delivering efficient performance, a favorable user experience, and adaptability to blockchain systems.

The combination of these three key mechanisms provides the CA-PBFT algorithm with flexibility, efficiency, and stability advantages. These enhancements allow the CA-PBFT algorithm to adapt to the ever-changing network environment and business requirements, improving consensus efficiency, trustworthiness, and security. As a result, it provides a more reliable consensus foundation for the rural property rights trading system framework.

5. Development and Implementation of the Prototype System

The developed prototype system comprises three layers: the user interface layer, the business logic layer, and the data storage layer, as illustrated in Figure 8.

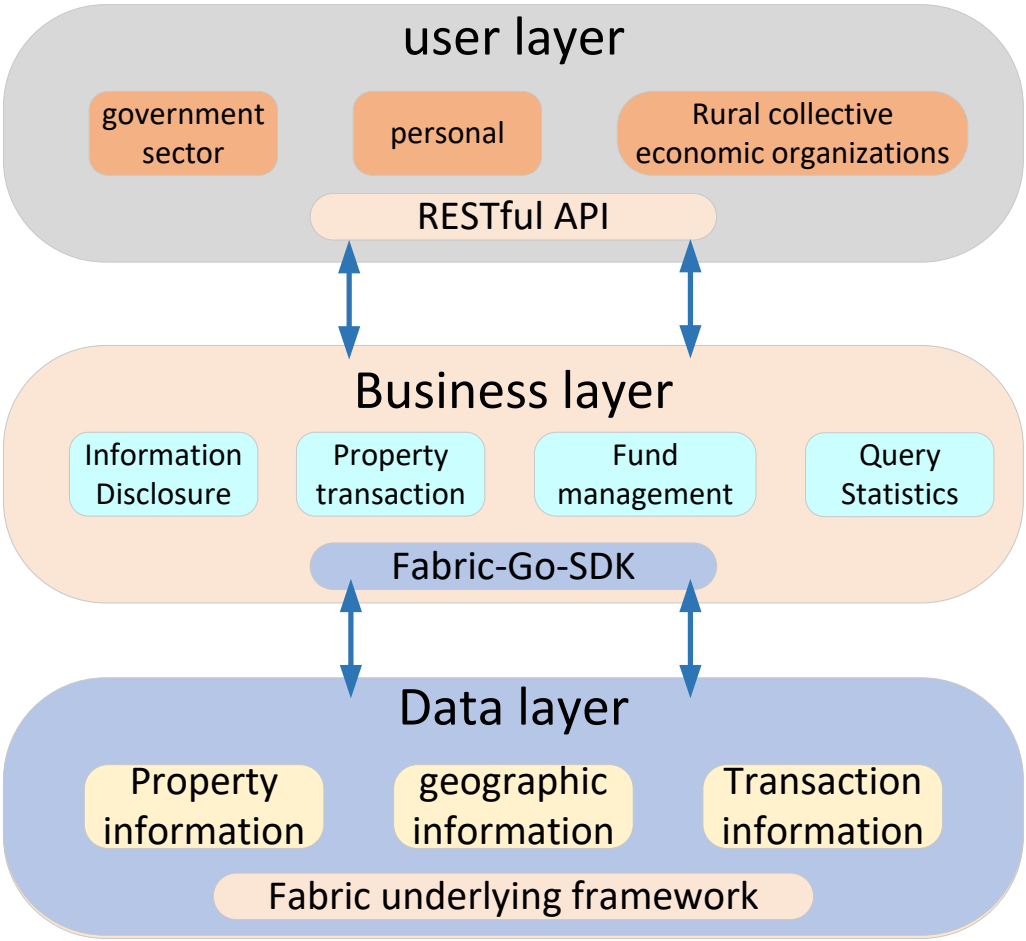

Figure 8. The architecture of a Blockchain-based Rural Property Rights Trading System.

In the architecture of a blockchain-based rural property rights trading system, the blockchain layer plays a pivotal role, as it is responsible for recording and storing all transaction information and constructing an immutable blockchain. Its function lies in ensuring the security, transparency, and traceability of transactions, endowing the entire system with reliable transaction records and property transfer details. We adopted Hyperledger Fabric as the blockchain framework for the rural property rights trading system and designed the following five aspects:

1. Smart contract development:
   On the blockchain, smart contracts are the code that implements business logic. We employed the GO programming language to write smart contracts, defining rules and operations for property rights transactions.
2. Data model design:
   Appropriate data structures were designed to store transaction information, geographical location data, and property-related details. This involves using state databases (such as CouchDB and LevelDB) to store data and defining the structure and relationships of the data.
3. Chaincode deployment:
   Chaincode is the executable instance of smart contracts on the blockchain. We deployed well-written smart contracts to nodes within the blockchain network, enabling them to be invoked and executed.
4. Consensus configuration:
   To facilitate the easy adjustment of consensus algorithm parameters and network node configurations, we configured the consensus mechanism of the blockchain network, ensuring that all nodes in the network achieve consensus on transaction approvals.

5. Transaction verification and block construction:
   When property transactions occur, they are submitted to the blockchain network and subsequently undergo verification and processing. Verification ensures transactions adhere to contract rules before adding them to new blocks.

The core business layer plays a crucial role in the architecture of rural property rights trading systems and is responsible for implementing the system's primary functionalities and key business logic, including property transactions, fund management, information disclosure, and property rights confirmation. To effectively realize these functions, modern technological approaches have been adopted.

During the development of the business logic layer, we have chosen the Spring Boot framework as the foundation. This framework offers a rapid and flexible development approach, enabling the swift construction of stable and efficient systems. We have employed a typical layered architecture methodology to enhance code maintainability and extensibility. Concurrently, the MyBatis framework was used as the data access layer, facilitating the mapping of Java objects to database tables and providing convenient data manipulation interfaces. Within the core service layer, we have defined business logic and functional interfaces to implement the system's essential features. These services receive user requests through the controller layer and delegate them to the appropriate processing services for handling.

Moreover, for interactions with blockchain technology, we have integrated the Fabric Java SDK into the Spring Boot application. This integration equips our Java applications with the capability to interact with the Hyperledger Fabric blockchain network, encompassing network connectivity, smart contract invocation, transaction sending and receiving, data querying, event processing, and more.

In the rural property rights trading system, the implementation of the user interface layer aims to provide users with an intuitive and efficient interactive experience to meet their needs and enhance system performance. We have chosen Vue.js as the frontend framework, utilizing its lightweight and flexible nature to construct interactive and dynamic interfaces. Emphasis was placed on prototype design and visual interface development to ensure rational layout and aesthetics. Employing a layered architecture approach, the interface was divided into independent components, enhancing code modularity and the reuse of interface elements. A data-binding functionality ensures data synchronization and interaction, facilitating real-time updates and responsiveness to user input. Through the built-in HTTP library of the framework, we can engage in API requests with the backend system and blockchain technology to perform key operations, including smart contract invocations and data queries.

After comprehensive testing and performance optimization, we ensured the stability of the user interface functionality and deployed it on reliable servers or cloud platforms to guarantee users an exceptional interactive experience. This established a solid foundation for the smooth operation of the rural property rights trading system.

In terms of the prototype system's utilization, initially, upon different users' logins (taking administrators as an example), the main page of the rural property rights trading prototype system furnishes users with an intuitive dashboard displaying real-time information, such as node statistics, transaction volume, smart contracts, and block height data. Furthermore, the rural property rights system framework offers a case inquiry feature, enabling users to retrieve case information stored within the blockchain by inputting case details. The query results present comprehensive case details, including transaction specifics, hash values, and transaction timelines. This functionality aids supervisory entities in more effectively managing the rural property rights trading market, as illustrated in Figures 9 and 10.

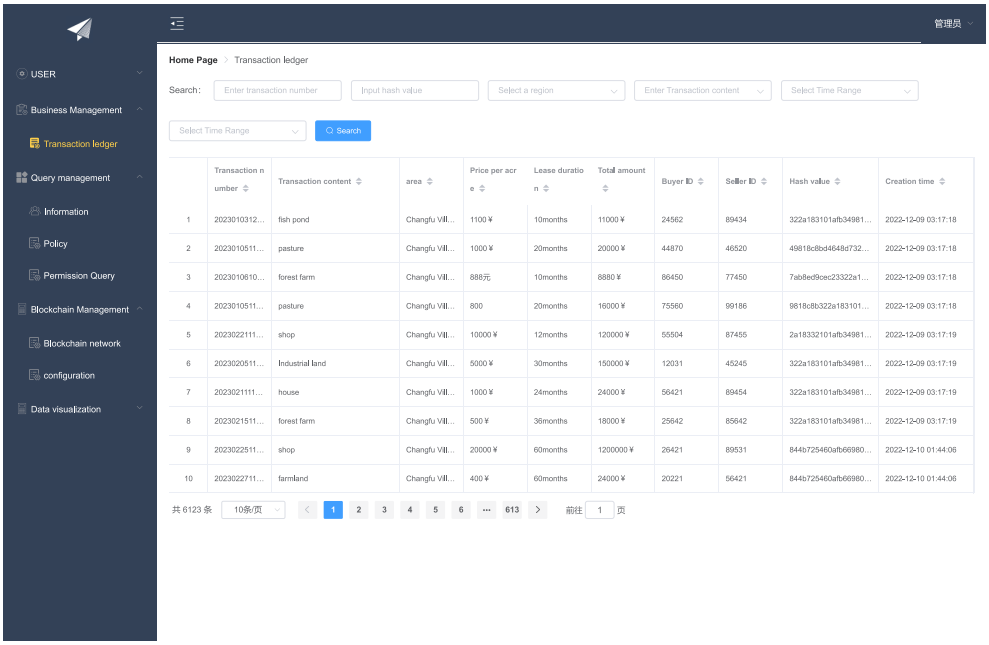

Figure 9. Main Interface of the System.

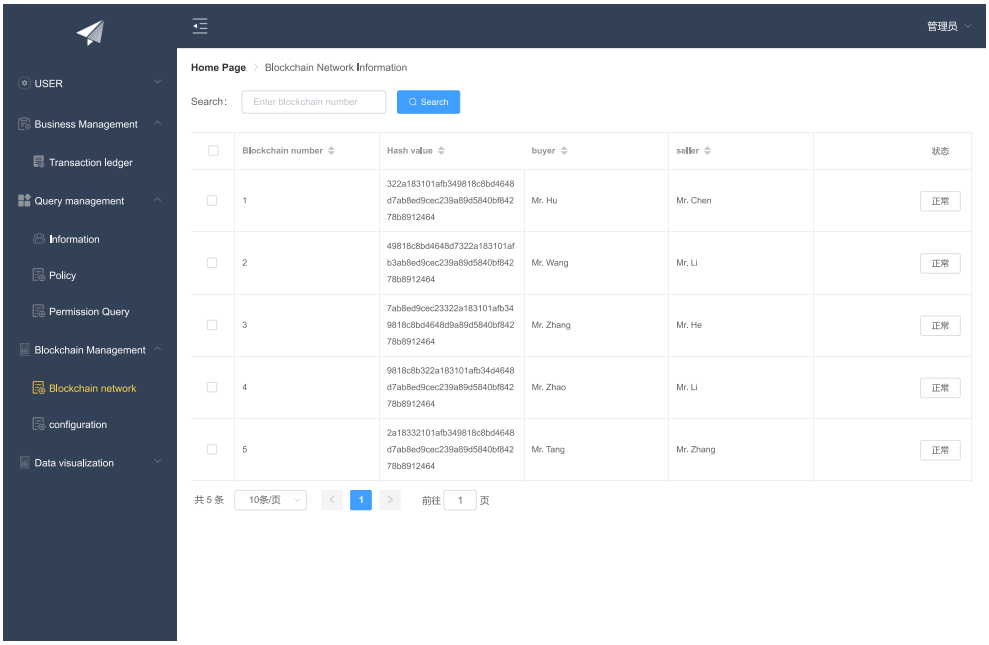

Figure 10. Information on the BlockChain.

6. Experimental Design and Analysis

In this section, a series of experiments were conducted to evaluate the performance and feasibility of the CA-PBFT consensus algorithm for its applicability in rural property rights trading. To accurately measure the algorithm's performance, several commonly used metrics, such as throughput, consensus latency, fault tolerance, and transaction request completion rate, were adopted. These metrics provided a comprehensive assessment of the algorithm's performance under various conditions, validating its applicability. The measurement tool utilized for these experiments was caliper, a widely recognized blockchain platform and consensus algorithm performance testing tool known for providing accurate experimental results and data analysis, which subsequently facilitated algorithm adjustments. These experiments and performance evaluations aimed to provide a thorough understanding of the potential and advantages of the CA-PBFT consensus algo-

rithm in the context of rural property rights trading, serving as a reliable foundation for its practical application.

A comparative analysis of the performance metrics was conducted for the CA-PBFT, PBFT, and the Credit-Based Single-Scoring PBFT Enhancement Algorithm (C-PBFT). It is worth noting that the primary distinction between CA-PBFT and C-PBFT lies in their adopted scoring mechanisms. CA-PBFT incorporates a dual-score mechanism, encompassing credit-scoring and the activation score, while C-PBFT relies solely on a single-score mechanism based on credit. These algorithms were used as control groups to compare the impact of integrating the activation score mechanism on performance under the same credit-scoring mechanism. This comparison aimed to provide insight into the influence of the dual-score mechanism on the performance of the CA-PBFT consensus algorithm and verify its superiority in practical applications. The experimental setup for the CA-PBFT algorithm is outlined in Table 1.

Table 1. Lab Environment.

Software	Version
Deployment Tool	Docker
SDK	Hyperledger Fabric SDK
CPU	Intel Core i7-9750 H 2.60 GHz
Memory	16 GB RAM
Operating System	Centos 7.6
Hyperledger Fabric	2.2
Deployment Tool	Docker
Memory	16 GB RAM
SDK	Hyperledger Fabric SDK

We conducted a series of experiments and conducted a comprehensive evaluation of the CA-PBFT consensus algorithm using the caliper tool. The evaluation included various performance metrics, such as throughput, consensus latency, fault tolerance, and transaction request completion rate. Furthermore, we compared the experimental results with those obtained using the PBFT and C-PBFT algorithms to gain a deeper understanding of CA-PBFT's performance and effectiveness in the context of rural property rights trading. These experimental findings offer valuable insights into the strengths and limitations of the algorithm, serving as a valuable reference for future optimizations and improvements. Ultimately, our goal is to contribute to the field of rural property rights trading by offering a more reliable and efficient consensus solution.

6.1. Throughput Analysis

In blockchain systems, throughput refers to the number of transactions processed by the system within a unit of time, often denoted as TPS (transactions per second). It serves as a critical metric for assessing the system's concurrent processing capacity. The calculation formula is as follows:

$$TPS = \frac{Transactions_{\Delta t}}{\Delta t} \tag{9}$$

$Transactions\Delta t$ represents the number of transactions processed within a time interval Δt. We conducted Experiments 1 and 2 to measure the throughput.

- Experiment 1

Throughout the experiments, we configured PBFT, C-PBFT, and CA-PBFT, each with seven consensus nodes, and systematically augmented the transaction volume using the caliper tool. We meticulously documented the count of completed consensus transactions per second for all three algorithms, and to enhance the validity of our experimental results, we executed multiple repetitions, averaging the transaction volumes at each stage. The aim

was to ensure the robustness and reliability of the outcomes obtained. The findings from Experiment 1 are presented in Figure 11.

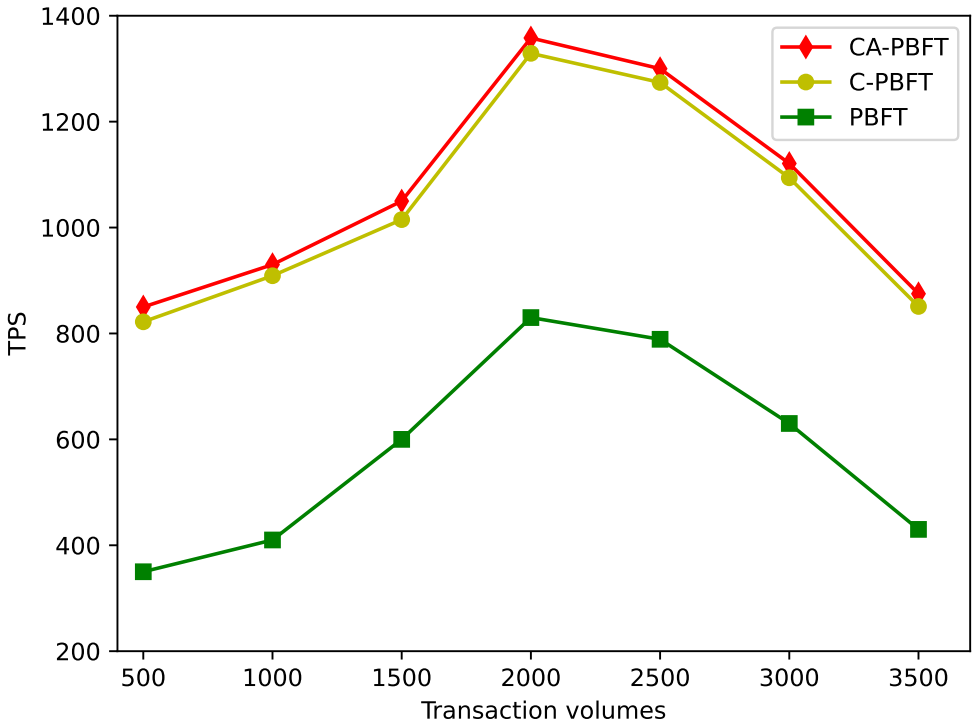

Figure 11. Comparison of Throughput at Various Transaction Volumes.

The incorporation of the dual-score mechanism encompassing credit-scoring and the activation score in CA-PBFT entails the exclusion of nodes with scores below a specified threshold from participating in the consensus process. This exclusion serves to mitigate unnecessary overhead. In contrast to C-PBFT, which relies solely on credit-scoring for node selection, CA-PBFT employs a more stringent and comprehensive approach to screen primary and consensus nodes. Consequently, this simplifies the consensus protocol, leading to reduced communication and processing times between nodes, naturally resulting in higher throughput compared to the PBFT algorithm.

Nevertheless, when the transaction volume exceeds 2000, all three consensus algorithms exhibit a decline in throughput. This can be attributed to two primary factors, as enumerated below.

First, the rise in latency: As transaction volume escalates, the burden of communication between nodes intensifies, culminating in extended message delivery and interaction delays. Although the CA-PBFT algorithm optimizes traditional PBFT by diminishing communication frequency and data transfers, it still grapples with challenges in circumventing delays under heavy load conditions. Consequently, this leads to an increase in consensus response time, thereby limiting further improvements in throughput.

Second, node processing capacity nears its limits: Beyond a certain transaction volume threshold, nodes may approach the limits of their processing capacity. Despite the reduction in communication burden between nodes, the CA-PBFT algorithm still mandates that nodes execute intricate consensus algorithms and smart contracts for transaction processing. In cases where nodes possess limited computational resources, transactions exceeding their processing capabilities are not handled promptly, resulting in a decrease in throughput.

- Experiment 2

To gain a more profound insight into the variations in throughput in the presence of Byzantine nodes during our experiments, we conducted a controlled approach. Specifically, we evenly distributed a quantity of 2000 transaction requests across the three consensus

algorithms while systematically increasing the node count and introducing Byzantine nodes. These experiments were meticulously repeated to ensure reproducibility. The collected data included the average number of completed transactions per second, which was meticulously recorded for analysis, as depicted in Figure 12.

From Figure 12a, it becomes apparent that as the number of nodes in the experiment increases, the throughput of all three algorithms decreases. This phenomenon can be attributed to the heightened communication burden among nodes as their numbers rise, resulting in amplified message transmission and interaction delays. However, concurrently, both CA-PBFT and C-PBFT exhibit higher throughput than PBFT. In the case of 4 to 16 nodes, CA-PBFT demonstrates a throughput approximately 100 to 150 TX/s higher than that of C-PBFT.

Figure 12b reveals that the presence of Byzantine nodes leads to a decrease in throughput for both CA-PBFT and C-PBFT. This decline is primarily attributed to the necessity for primary nodes to terminate the simplified consensus protocol and transition to the backup protocol. This transition requires additional time and consequently impacts the algorithm's throughput [52].

In summary, the CA-PBFT algorithm consistently exhibits higher throughput than the other two algorithms.

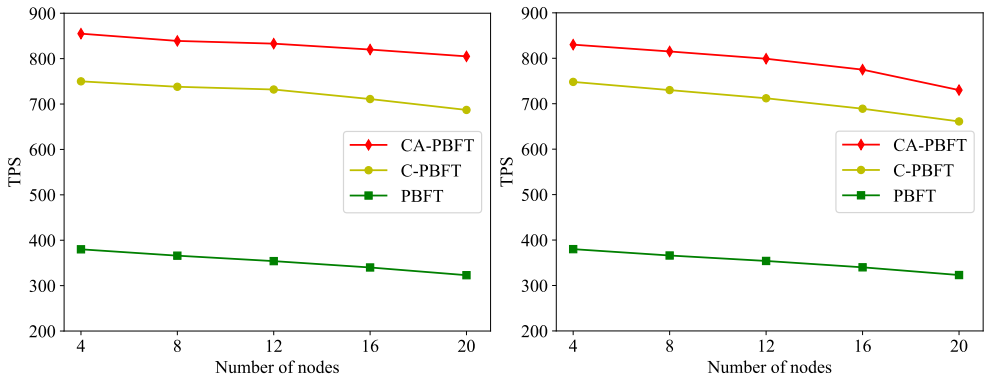

Figure 12. Comparison of Throughput with and without Byzantine Nodes: (**a**) Experimental group without Byzantine nodes; (**b**) Experimental group with Byzantine nodes.

6.2. Consistency Delay Analysis

A consistency latency analysis encompasses the evaluation and investigation of consistency protocols and mechanisms within distributed systems. This process entails the measurement and scrutiny of time delays introduced during data synchronization and replica updates within the system. The primary objective of a consistency latency analysis is to gain insights into the system's performance across various scenarios, including varying workloads and network latencies. Additionally, it seeks to determine the time required to achieve data consistency. Through a latency analysis, it becomes possible to optimize the system's performance and resource utilization, ensuring that the system upholds data consistency while operating efficiently. We employed Equation (10) for the calculation of consistency delay, rendering this metric indispensable for assessing the speed and practical effectiveness of the consensus algorithm in real-world applications.

$$Tcd = Tfinish - Trequest \tag{10}$$

Tcd symbolizes the consistency delay, signifying the duration starting from the client's request (Trequest) to the finalization of block confirmation in the consensus process (Tfinish). In order to capture the fluctuations in consistency delay, we devised Experiments 3 and 4. These experiments involved conducting numerous measurements to derive average values, which were subsequently used for constructing graphical representations.

- Experiment 3

We conducted experiments to record the variations in consistency delay under different node conditions and compared the impact of different block generation times on algorithm performance by adjusting the configuration files. In Figure 13, we present the changes in consistency delay for both algorithms as the number of nodes varies when the block generation times are set to 5 s (Figure 13a), 10 s (Figure 13b), 15 s (Figure 13c), and 20 s (Figure 13d). This experimental design allowed us to gain insights into the effects of different parameters on algorithm performance, providing valuable references for further optimization and improvement.

By observing Figure 13, we can gain a clearer understanding of the performance of the three algorithms under different conditions, enabling a more accurate and comprehensive assessment of the performance of CA-PBFT. Based on the results in Figure 13, we can draw the following conclusions:

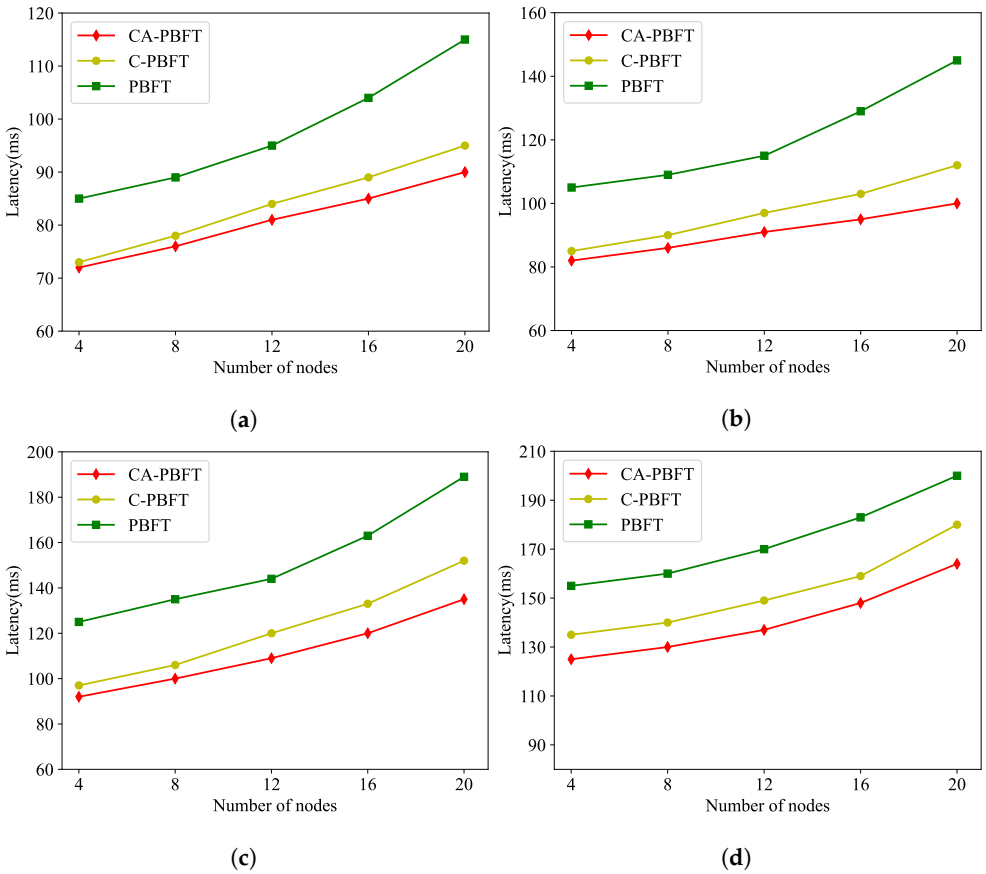

Figure 13. Variation of consensus latency with node count under different block generation times: (**a**) 5 s, (**b**) 10 s, (**c**) 15 s, and (**d**) 20 s.

The increase in the number of nodes results in an elevated communication load among nodes, leading to an extended transaction processing time and consequently causing an overall increase in the system's consensus latency. Compared to PBFT and C-PBFT, our proposed CA-PBFT consensus algorithm exhibits a lower consensus delay. This is attributed to the dynamic joining and exiting mechanism, the introduction of the dual-scoring mechanism based on both credit and activation, and the simplified consensus protocol. These optimizations streamline the consensus process, reducing inter-node communication frequency and data transmission, thereby enhancing the overall efficiency and performance of the system.

Overall, our CA-PBFT consistently outperforms PBFT and C-PBFT in terms of consensus latency across various block generation times, achieving faster consensus and enhancing system performance and efficiency.

- Experiment 4

Under the same block generation time (10 s), we conducted a comparative experimental study on consensus latency for three consensus algorithms in two scenarios: one with Byzantine nodes and the other without. The aim of this study was to investigate the impact of Byzantine nodes on consensus latency. Figure 14 illustrates the variations in consensus latency for these two scenarios.

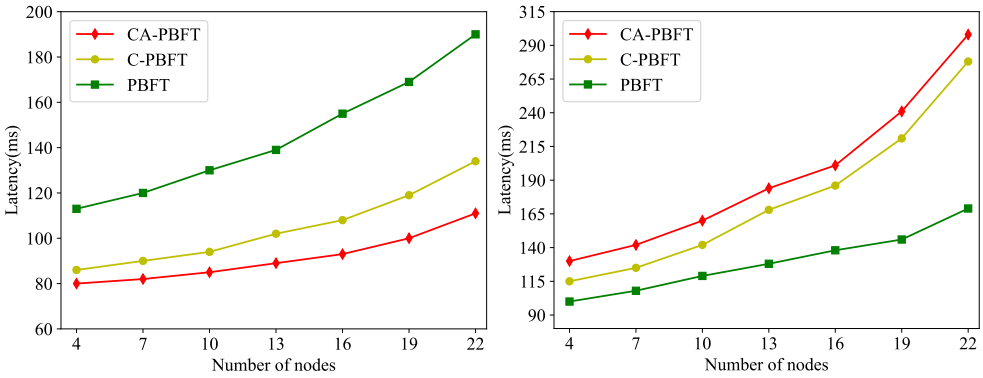

Figure 14. Consensus Latency Variation with Presence of Byzantine Nodes: (**a**) Without Byzantine Nodes; (**b**) With Byzantine Nodes.

When Byzantine nodes are absent, both CA-PBFT and C-PBFT adopt a simplified consensus protocol. As observed from Figure 14a, with an increase in the number of nodes, the consensus latency of the PBFT algorithm rapidly increases. In contrast, the consensus latencies of the CA-PBFT and C-PBFT algorithms are lower than that of PBFT, and as the number of nodes increases, their consensus latencies grow at a slower and relatively stable rate. This advantage can be attributed to our optimization of the consensus process in CA-PBFT and C-PBFT by simplifying their consensus mechanisms. Additionally, due to the lack of an activation node selection process in C-PBFT, resulting in slightly higher inter-node communication overhead compared to CA-PBFT, we can observe from Figure 14a that the consensus latency of the CA-PBFT is slightly lower than that of C-PBFT at each stage.

In the presence of Byzantine nodes, the comparison of consensus latency among CA-PBFT, C-PBFT, and PBFT algorithms shows a significant increase, as depicted in Figure 14b. This increase can be attributed to the necessity of the primary node to terminate the simplified consensus protocol in the presence of Byzantine nodes, leading to a considerable increase in consensus latency.

In conclusion, the CA-PBFT consensus algorithm demonstrates significant advantages when there are no Byzantine nodes, while its performance experiences a slight decrease in the presence of Byzantine nodes due to protocol termination effects. Nonetheless, the CA-PBFT consensus algorithm still performs excellently in most cases and provides a more efficient solution for scenarios without Byzantine nodes.

6.3. Fault-Tolerance Security Analysis

- Experiment 5

Fault-tolerance security analysis holds distinct significance within consensus algorithms, playing a pivotal role in ensuring the stable operation, data security, and consistency of systems in distributed and decentralized environments. Through comprehensive fault tolerance and security analysis, it becomes possible to construct more reliable, secure, and resilient blockchain applications. In Experiment 5, we established a network of 100 nodes,

designating 32 of them as Byzantine nodes, and subsequently conducted 40 rounds of consensus. As depicted in Figure 15, the changes in the count of Byzantine nodes are shown for PBFT, C-PBFT, and CA-PBFT as the consensus rounds progress.

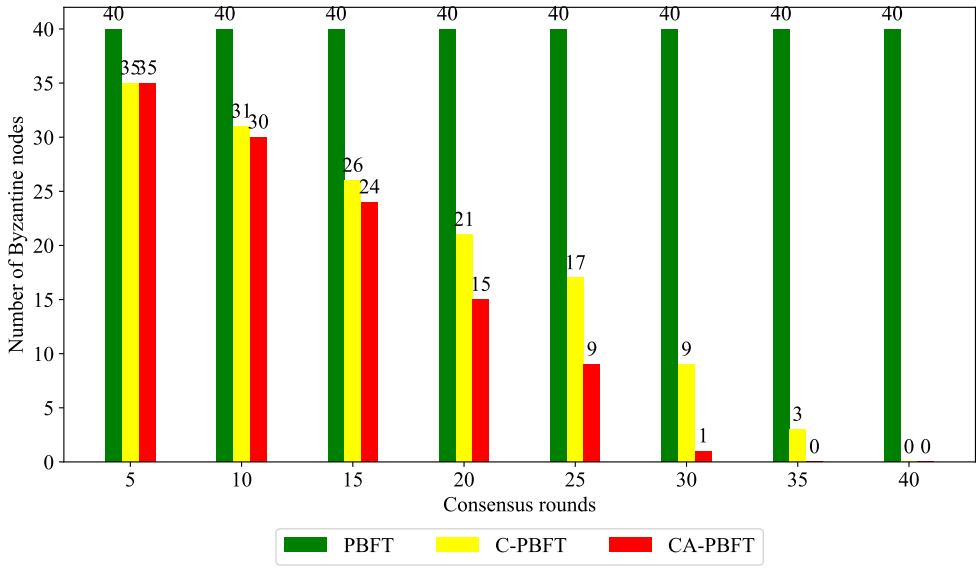

Figure 15. The variation of Byzantine node count with increasing consensus rounds.

As depicted in Figure 15, after 40 rounds of consensus, the PBFT algorithm, which does not employ a rating mechanism, still retains 40 Byzantine nodes. In contrast, both the CA-PBFT and C-PBFT algorithms exhibit a gradual reduction in the number of Byzantine nodes with increasing consensus rounds. Notably, our proposed CA-PBFT consensus algorithm successfully reduces the count of Byzantine nodes to zero after 40 rounds of consensus. This situation illustrates that in the case of long-term system operation, the CA-PBFT and C-PBFT algorithms effectively exclude Byzantine nodes from the consensus group through the rating mechanism. This enhances the reliability of primary nodes, mitigates Byzantine failures and malicious attacks, optimizes algorithm performance, and incentivizes favorable node behavior. Consequently, these mechanisms augment the overall trustworthiness and security of the system.

Compared to C-PBFT, CA-PBFT provides an additional safeguard of positivity-based ratings, resulting in a more comprehensive evaluation of nodes. As a result, it can more precisely and rapidly identify and eliminate malicious nodes, significantly enhancing the reliability of primary nodes and bolstering resistance against Byzantine failures. The CA-PBFT consensus algorithm excels in fault tolerance and security aspects, while also being particularly suited for rural property rights transactions. It offers an effective solution for the secure and stable operation of blockchain in complex network environments.

6.4. Analysis of Transaction Request Valid Completion Rate

- Experiment 6

In Experiment 6, we recorded and compared the transaction completion rates of different algorithms as the number of nodes increased, and the results are shown in Figure 16. From the graph, it can be observed that regardless of the number of nodes, the CA-PBFT algorithm achieves the highest transaction completion rate. This indicates that the CA-PBFT algorithm excels in ensuring a high transaction request valid completion rate, demonstrating superior security and reliability irrespective of network scale. This significantly improves efficiency in rural property rights transactions, enhances user trust, and improves user experience.

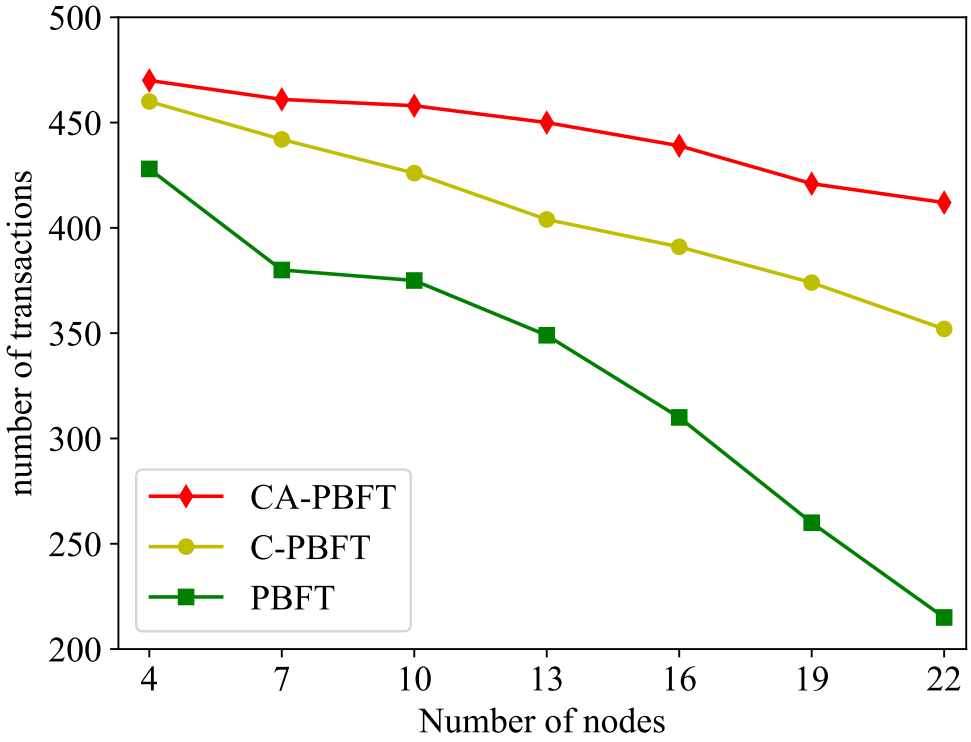

Figure 16. Comparison of completed number of transactions among different algorithms.

In Table 2, we compared the transaction request valid completion rates of different algorithms and observed their changes with the increase in the number of nodes. The results show that as the number of nodes increases, the valid completion rate of the PBFT algorithm exhibits a noticeable decline, while the CA-PBFT algorithm demonstrates a relatively stable overall change with a slight decrease. Compared to C-PBFT, the CA-PBFT algorithm exhibits a slight advantage. This indicates that during the consensus process, the CA-PBFT algorithm displays superior stability, significantly reducing the probability of malicious nodes being selected as primary nodes. Furthermore, the introduction of a mechanism to activate the removal of malicious nodes helps to expel them from the network, thereby enhancing the overall security and stability of the nodes. As a result, each transaction request can achieve consensus smoothly, reducing network resource waste caused by multiple retries and effectively decreasing network overhead.

Table 2. Effective completion rates of transaction request comparison among different algorithms.

The Number of Nodes	PBFT	C-PBFT	CA-PBFT
4	86.8%	92.0%	94.0%
7	78.2%	88.4%	92.2%
10	74.8%	85.2%	91.6%
13	70.2%	80.8%	90.0%
16	62.2%	78.2%	87.8%
19	53.0%	74.8%	84.2%
22	44.8%	70.0%	82.4%

In summary, the CA-PBFT algorithm exhibits a higher rate of successful transaction completion and stability within rural property rights trading platforms. This contributes to enhanced transaction efficiency, improved security, and overall optimization of platform performance. In complex network environments, such robustness and efficiency play a crucial role in bolstering the trustworthiness, fostering positive engagement, and ensuring the security and reliability of rural property rights trading platforms.

6.5. Summary of CA-PBFT

The CA-PBFT algorithm has undergone extensive enhancements to adapt to the rural property rights trading scenario and overcome challenges encountered by PBFT, including issues related to dynamics, inflexibility, and high communication overhead. These improvements include the introduction of dynamic mechanisms, the incorporation of a dual-scoring mechanism based on credit and activation, and optimizations in protocol consistency.

Firstly, the introduction of dynamic mechanisms empowers CA-PBFT to seamlessly handle node dynamics, such as node joinings or exits, without requiring network-wide restarts. This significantly enhances system flexibility and stability, allowing nodes to join or leave during runtime without causing disruptions across the entire network.

Secondly, the advantages of the dual-scoring model over a single scoring model are emphasized, and these improvements are outlined as follows:

1. Comprehensiveness of dual scoring:
    - The dual-scoring model provides a more comprehensive node assessment. The credit score focuses on node stability and reliability, while the activation score emphasizes node activeness and efficiency. The combination ensures that selected nodes are both trustworthy and efficient.
2. Enhanced security:
    - In a single-scoring model, malicious nodes could boost their score through specific actions to participate in consensus. However, in the dual-scoring model, malicious nodes must excel in both dimensions, significantly increasing the difficulty of their selection. Additionally, CA-PBFT directly removes malicious nodes from the consensus process, greatly enhancing system security.
3. Performance improvements:
    - The dual-scoring mechanism considers factors such as node proximity to active transaction areas, transaction latency, transaction amounts, and transaction frequency. This ensures that selected nodes are not only reliable but also most likely to process transactions quickly;
    - Ensuring that primary nodes are efficient accelerates the entire consensus process, enhancing performance.
4. Enhanced adaptability:
    - Tailored for rural property rights trading platforms, the dual-scoring mechanism specifically addresses critical factors in this scenario, such as node proximity to active transaction areas, making it more suitable than a generic single-scoring model.

Furthermore, optimizations in protocol consistency further elevate the algorithm's performance and communication efficiency. By reducing communication overhead and latency, CA-PBFT can process transactions more rapidly, aligning with the requirements of rural property rights trading scenarios.

In conclusion, the CA-PBFT algorithm, through the introduction of dynamic mechanisms, dual scoring, and protocol consistency enhancements, effectively overcomes the limitations of PBFT. It brings multiple advantages to rural property rights trading systems, including improved system flexibility, fault tolerance, reliability, and enhanced transaction processing efficiency through the reduction of communication overhead and latency. In the realm of rural property rights trading, CA-PBFT offers a viable solution for constructing an efficient, secure, and stable consensus mechanism.

7. Conclusions and Prospects

This study centers on the implementation of a rural property rights trading system framework based on blockchain technology, with the overarching aim of improving transaction efficiency, bolstering security, and fostering the healthy growth of China's rural

property rights trading market. Conventional rural property rights transactions grapple with issues such as centralized server-based data manipulation, culminating in data fraud and information imbalances. The absence of dependable information recording exacerbates disputes and conflicts among involved parties. Blockchain technology, characterized by its immutability, transparency, traceability, and self-executing protocols, offers a fresh perspective on rural property rights trading.

By scrutinizing the challenges ingrained in traditional rural property rights trading, we proffer a blueprint for rural property rights trading systems grounded in the Hyperledger Fabric consortium blockchain. This blueprint incorporates an enhanced PBFT consensus algorithm to amplify information throughput and curtail transaction latency. The framework's objectives encompass ameliorating key issues in the rural property rights trading sphere, including deficiencies in credit reporting, subpar transaction efficiency, non-standardized transaction contracts, and soaring equity protection expenses. Capitalizing on blockchain's attributes, the proposed framework ascertains reliable information recording and automated transaction execution, thereby elevating the transparency and credibility of the entire transaction process.

We have amalgamated theoretical and practical research to underpin the implementation of a blockchain-based rural property rights trading system framework. By adopting the enhanced PBFT consensus algorithm, we accentuate the benefits of the dual-scoring mechanism rooted in credit and activation. This empowers the system to adapt dynamically to node entrances and exits, thereby heightening system stability and adaptability.

Furthermore, experimental investigations into consensus latency and transaction request completion rates substantiate the superior performance of the CA-PBFT algorithm across distinct node scenarios, coupled with its commendable resilience to Byzantine faults, ultimately buttressing the dependability and security of participating consensus nodes.

In sum, our research outcomes hold promise in efficaciously tackling issues in the realm of rural property rights trading, offering support for rural revitalization and growth, and fostering a virtuous cycle in China's rural property rights trading landscape. By ushering in blockchain technology and instating trustworthy transaction records and automated execution mechanisms, the rural property rights trading system stands to achieve more streamlined and secure transaction processes, consequently contributing positively to the robust development of the rural economy and social stability.

Author Contributions: Conceptualization, C.H. and S.W.; methodology, C.H. and S.W.; software, Y.Z. and K.L.; validation, M.L., K.L. and J.F.; formal analysis, S.W.; investigation, C.H. and S.W.; resources, C.H. and S.W.; data curation, C.H. and S.W.; writing—original draft preparation, S.W.; writing—review and editing, C.H. and S.W.; visualization, S.W. and M.L.; supervision, C.H.; project administration, C.H. and S.W.; funding acquisition, C.H. and Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Sichuan University of Science and Engineering for Talent Introduction Project under Grant 2017RCL59 and in part by the Key Laboratory of Higher Education of Sichuan Province for Enterprise Informationalization and Internet of Things Plan Project under Grant 2022WYY01.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Li, Z.X. Design and implementation of intellectual property rights exhibition and trade platform. *Electron. Des. Eng.* **2013**, *21*, 9–12.
2. Wang, B.; Li, J.J. Design and Implementation of Rural Comprehensive Property RightsTransaction System. *Geospat. Inf.* **2017**, *15*, 77–79.

3. Jiang, S.; Cao, J.; Zhu, J.; Cao, Y. PolyChain: A generic blockchain as a service platform. In Proceedings of the International Conference on Blockchain and Trustworthy Systems, Guangzhou, China, 5–6 August 2021; pp. 459–472.
4. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. *Decentralized Bus. Rev.* **2008**, 21260.
5. Wood, G. Ethereum: A Secure Decentralised Generalised Transaction Ledger. *Ethereum Proj. Yellow Pap.* **2014**, *151*, 1–32.
6. Wu, H.T.; Zhong, B.T.; Li, H.; Guo, J.D.; Wang, Y.H. On-Site Construction Quality Inspection Using Blockchain and Smart Contracts. *J. Manag. Eng.* **2021**, *37*, 04021065. [CrossRef]
7. Xu, L.; Chen, W.; Li, Z.; Xu, J.; Liu, A.; Zhao, L. Solutions for concurrency conflict problem on Hyperledger Fabric. *World Wide Web* **2021**, *24*, 463–482. [CrossRef]
8. Zhong, B.; Guo, J.; Zhang, L.; Wu, H.T.; Li, H. A blockchain-based framework for on-site construction environmental monitoring: Proof of concept. *Build. Environ.* **2022**, *217*, 109064. [CrossRef]
9. Hang, L.; Kim, D. Optimal blockchain network construction methodology based on analysis of configurable components for enhancing Hyperledger Fabric performance. *Blockchain Res. Appl.* **2021**, *2*, 100009. [CrossRef]
10. Firdaus, A.; Razak, M.F.A.; Feizollah, A.; Hashem, I.A.T.; Hazim, M.; Anuar, N.B. The rise of "blockchain": Bibliometric analysis of blockchain study. *Scientometrics* **2019**, *120*, 1289–1331. [CrossRef]
11. Touloupou, M.; Themistocleous, M.; Iosif, E.; Christodoulou, K. A Systematic Literature Review Toward a Blockchain Benchmarking Framework. *IEEE Access* **2022**, *10*, 70630–70644. [CrossRef]
12. Duan, Z.; Feng, W.; Zhong, W.; Huang, M.; Feng, S. Form Specification of Smart Contract for Intellectual Property Transaction Based on Blockchain. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 3274454. [CrossRef]
13. Sladić, G.; Milosavljević, B.; Nikolić, S.; Sladić, D.; Radulović, A. A Blockchain Solution for Securing Real Property Transactions: A Case Study for Serbia. *ISPRS Int. J.-Geo-Inf.* **2021**, *10*, 35. [CrossRef]
14. Perera, S.; Hijazi, A.A.; Weerasuriya, G.T.; Nanayakkara, S.; Rodrigo, M.N.N. Blockchain-Based Trusted Property Transactions in the Built Environment: Development of an Incubation-Ready Prototype . *Buildings* **2021**, *11*, 560. [CrossRef]
15. Mashatan, A.; Lemieux, V.; Lee, S.H.; Szufel, P.; Roberts, Z. Usurping Double-Ending Fraud in Real Estate Transactions via Blockchain Technology. *J. Database Manag.* **2021**, *32*, 27–48. [CrossRef]
16. Yadav, A.S.; Agrawal, S.; Kushwaha, D.S. Distributed Ledger Technology-based land transaction system with trusted nodes consensus mechanism. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 6414–6424. [CrossRef]
17. Ma, Z.; Zhao, W.; Luo, S.; Wang, L. TrustedBaaS: Blockchain-Enabled Distributed and Higher-Level Trusted Platform. *Comput. Netw.* **2020**, *183*, 107600. [CrossRef]
18. Ahmad, I.; Alqarni, M.A.; Almazroi, A.A. Real Estate Management via a Decentralized Blockchain Platform. *Comput. Mater. Contin.* **2021**, *66*, 1813–1822. [CrossRef]
19. Song, H.; Zhu, N.; Xue, R. Proof-of-Contribution consensus mechanism for blockchain and its application in intellectual property protection. *Inf. Process. Manag.* **2021**, *58*, 102507. [CrossRef]
20. Zhao, L.T.; Zhang, J.W.; Jing, H.R. Blockchain-Enabled Digital Rights Management for Museum-Digital Property Rights. *Intell. Autom. Soft Comput.* **2022**, *34*, 1785–1801. [CrossRef]
21. Hasan, M.A.; Tariq, M.; Chen, Z. A digital rights management system based on a scalable blockchain. *Peer -Peer Netw. Appl.* **2020**, *14*, 2665–2680.
22. Liu, Y.; Shang, C. Application of Blockchain Technology in Agricultural Water Rights Trade Management. *IEEE Access* **2022**, *14*, 7017. [CrossRef]
23. Chen, C.L.; Fang, C.C.; Zhou, M.A. Blockchain-Based Anti-Counterfeit and Traceable NBA Digital Trading Card Management System. *Symmetry* **2022**, *14*, 1827. [CrossRef]
24. Thien, H.T.; Thippa, R.G.; Wang, W.Z. Blockchain for the metaverse: A Review. *ScienceDirect* **2023**, *143*, 401–419.
25. Wang, X.J.; Zhu, H.L.; Guo, L. Blockchain Intelligence for Internet of Vehicles: Challenges and Solutions. *IEEE Commun. Surv. Tutor.* **2023**, *10*, 3305312. [CrossRef]
26. Peng, J.; Zhang, L.; You, S.M. Blockchain technology applications in waste management: Overview, challenges and opportunities. *ScienceDirect* **2023**, *421*, 138466.
27. Li, K.P.; Lee, J.Y.; Amir, G. Blockchain in food supply chains: A literature review and synthesis analysis of platforms, benefits and challenges. *Int. J. Prod. Res.* **2021**, *61*, 3527–3546. [CrossRef]
28. Milad, B.S.; Robert, C.M.; Sina, F.M. Blockchain in the Construction Industry between 2016 and 2022: A Review, Bibliometric, and Network Analysis. *Smart Cities* **2023**, *6*, 819–845.
29. Yuan, Y.; Ni, X.C.; Zeng, S.; Wang, F.Y. Blockchain Consensus Algorithms: The State of the Art and Future Trends. *Acta Autom. Sin.* **2018**, *44*, 2011–2022.
30. Yao, S.; Yingsen, W. SV-PBFT: An Efficient and Stable Blockchain PBFT Improved Consensus Algorithm for Vehicle-to-Vehicle Energy Transactions. *J. Internet Technol.* **2022**, *23*, 1191–1201. [CrossRef]
31. Almakki, R.; AlSuwaidan, L.; Khan, S.; Baig, AR.; Baseer, S.; Singh, M. Fault Tolerance Byzantine Algorithm for Lower Overhead Blockchain. *Secur. Commun. Netw.* **2022**, *2022*, 1855238. [CrossRef]
32. Wu, J.Y.; Liu, Y.; Cai, J.F.; Su, S.H. Blockchain Consensus Mechanism for Distributed Energy Transactions. *Wirel. Commun. Mob. Comput.* **2022**, *44*, 1–13. [CrossRef]
33. Liu, W.; Zhang, X.; Feng, W.; Huang, M.; Xu, Y. Optimization of the PBFT Algorithm Based on QoS-Aware Trust Service Evaluation. *Sensors* **2022**, *22*, 4590. [CrossRef] [PubMed]

34. Fang, W.W.; Wang, Z.Y.; Song, H.L.; Wang, Y.P.; Ding, Y. An optimized PBFT consensus algorithm for blockchain. *J. Beijing Jiaotong Univ.* **2019**, *43*, 58–64.
35. Liu, X.N.; Zhang, R.H.; Liu, C.Z. Improvement of PBFT Consensus Algorithm Based on Grouping and Credit Rating. *Comput. Eng.* **2023**, *36*, 1–13.
36. Meng, W.T.; Zhang, D.W. Optimization Scheme for Hyperledger Fabric Consensus Mechanism. *Acta Autom. Sin.* **2021**, *47*, 1885–1898.
37. Reng, X.Y.; Tong, X.R.; Zhang, W. PBFT Consensus Algorithm Based on Trust Evaluation Mode. *J. Shanxi Univ.* **2023**, *46*, 108–118.
38. Liu, Z.K.; Wang, F.; Jia, H.R. Optimization Scheme of PBFT Algorithm Combining Dynamic Credit Mechanism. *Comput. Eng.* **2023**, *49*, 191–198.
39. Zhong, W.; Feng, W.L.; Huang, M.X.; Feng,SL. ST-PBFT: An Optimized PBFT Consensus Algorithm for Intellectual Property Transaction Scenarios. *Electronic* **2023**, *12*, 325. [CrossRef]
40. Ouyang, L.W.; Wang, S.; Yuan, Y. Smart contracts: Architecture and research progresses. *Acta Autom. Sin.* **2019**, *45*, 445–457.
41. Ding, T.S.; Cheng, S.P. Improved PBFT Consensus Mechanism Based on Credit-Layered Mechanism. *Comput. Syst. Appl.* **2020**, *29*, 255–259.
42. Elli, A.; Artem, B. Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains. *Ithaca Cornell Univ. Libr.* **2018**, *30*, 1–15.
43. Kuzlu, M.; Pipattanasomporn, M.; Gurses, L.; Rahman, S. Performance analysis of a hyperledger fabric blockchain framework: Throughput, latency and scalability. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, Atlanta, GA, USA, 14–17 July 2019; pp. 536–540.
44. Zheng, Z.; Xie, S.; Dai, H.; Chen, W.; Chen, X.; Weng, J. An overview on smart contracts: Challenges, advances and platforms. *Future Gener. Comput. Syst.* **2020**, *105*, 475–491. [CrossRef]
45. Peter, G.; Aggelos, K.; Dionysis, Z. Proof-of-stake sidechains. In Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 19–23 May 2019; pp. 139–156.
46. Lobanov, A.V. Models of Closed Multimachine Computer Systems with Transient-fault-tolerance and Fault-tolerance on the Basis of Replication under Byzantine Faults. *Autom. Remote Control* **2009**, *70*, 328–343. [CrossRef]
47. Lashkari, B.; Musilek, P. A Comprehensive Review of Blockchain Consensus Mechanisms. *IEEE Access* **2021**, *9*, 43620–43652. [CrossRef]
48. Fu, X.M.; Wang, H.; Shi, P. A survey of Blockchain consensus algorithms: Mechanism, design and applications. *Sci. China Inf. Sci.* **2021**, *62*, 121101. [CrossRef]
49. Wang, Q.; Li, H.J.; Ni, X.L. Survey on Blockchain Consensus Algorithms and Application. *J. Front. Comput. Sci. Technol.* **2022**, *16*, 1214–1242.
50. Di, S.P.; Huo, Y.Y.; Yang, Y. Improvement of PBFT algorithm based on consistent hash and random selection. *Comput. Eng. Appl.* **2023**, *14*, 1–11.
51. Chen, S.M.; Wang, B.; Chen, Y.Q. Improved PBFT consensus algorithm based on node grouping reputation model. *Appl. Res. Comput.* **2023**, *11*, 1–7.
52. Xu, J.Y.; Hua, C.J.; Zhang, Y. A blockchain-based framework for supervision of livelihood issues: Proof of concept with optimized consensus. *IEEE Access* **2023**, *11*, 73414–73434. [CrossRef]