electronics

MDPI

Article

Improvement of Practical Byzantine Fault Tolerance Consensus Algorithm Based on DIANA in Intellectual Property Environment Transactions

Jing Wang, Wenlong Feng *, Mengxing Huang, Siling Feng and Dan Du

School of Information and Communication Engineering, Hainan University, Haikou 570228, China; 22220854000028@hainanu.edu.cn (J.W.); huangmx09@163.com (M.H.); fengsiling@hainanu.edu.cn (S.F.); 22220854000035@hainanu.edu.cn (D.D.)
* Correspondence: fwlfwl@163.com

Abstract: In response to the shortcomings of the consensus algorithm for intellectual property transactions, such as high communication overhead, random primary node selection, and prolonged consensus time, a Practical Byzantine Fault Tolerance (PBFT) improvement algorithm based on Divisive Analysis (DIANA) D-PBFT algorithm is proposed. Firstly, the algorithm adopts the hierarchical clustering mechanism of DIANA to cluster nodes based on similarity, enhancing node partition accuracy and reducing the number of participating consensus nodes. Secondly, it designs a reward and punishment system based on node ranking, to achieve consistency between node status and permissions, timely evaluation, and feedback on node behaviours, thereby enhancing node enthusiasm. Then, the election method of the primary node is improved by constructing proxy and alternate nodes and adopting a majority voting strategy to achieve the selection and reliability of the primary node. Finally, the consistency protocol is optimised to perform consensus once within the cluster and once between all primary nodes, to ensure the accuracy of the consensus results. Experimental results demonstrate that the D-PBFT algorithm shows a better performance, in terms of communication complexity, throughput, and latency.

Keywords: blockchain; intellectual property transactions; consensual algorithm; PBFT algorithm; DIANA hierarchical clustering

Citation: Wang, J.; Feng, W.; Huang, M.; Feng, S.; Du, D. Improvement of Practical Byzantine Fault Tolerance Consensus Algorithm Based on DIANA in Intellectual Property Environment Transactions. *Electronics* **2024**, *13*, 1634. https://doi.org/10.3390/electronics13091634

Academic Editor: Fabio Grandi

Received: 16 March 2024
Revised: 5 April 2024
Accepted: 23 April 2024
Published: 24 April 2024

1. Introduction

Intellectual property transactions have become an essential driving force for high-quality economic development and complete and effective transactions can strongly promote the development of science and technology. Intellectual property trading platforms have gone through variations from C/S management information systems, B/S models [1], Web Services, databases [2], SOA [3], XML, and cloud models [4]; however, all of the above traditional intellectual property trading platforms rely on centralised intermediaries, through which negotiations are brokered and contract enforcement is ensured. This centralised model fails to effectively address issues such as platform fraud, lack of trust among multiple parties involved in transactions [5], high transaction costs, and information asymmetry [6].

There is a problem of high costs associated with the methods of intellectual property rights transactions; traditional intellectual property transactions are usually coordinated with the help of intermediaries who charge fees for their services and these fees directly add to the total cost of the transaction. The problems of information asymmetry in intellectual property transactions include the relatively weak regulation and supervision of traditional intellectual property transaction platforms; the lack of uniform standards for prices, standards, and processes of transactions; and the fact that it is often difficult for buyers and sellers to ascertain the authenticity, legitimacy, and fairness of transactions. The difficulty

in intellectual property rights transaction protection lies primarily in the preservation and deposition of evidence, often facing issues such as false evidence and evidence tampering. There are also information technology constraints. In traditional intellectual property transactions, the way that information is exchanged between transaction participants is relatively traditional and it is not easy to transmit and share information quickly. This makes it difficult for transaction participants to understand the details of the transaction, market demand, and popular trends.

Intellectual property transaction is a multi-party scenario and the participants usually involve right holders, inventors, transferees, and intermediaries; how to efficiently coordinate and motivate these different interests is an urgent problem that needs to be solved. Whereas today's rapidly evolving blockchain technology and intellectual property transactions are a good match for each other, the immutability, high transparency [7], and traceability of blockchain offer inherent advantages in establishing trust, facilitating information sharing, and enabling information traceability in intellectual property transactions. Blockchain has emerged as a new computational paradigm and collaborative model for establishing trust at low cost in untrusted environments [8]. The application of blockchain technology in intellectual property transactions can facilitate more secure and transparent transactions. The electronic evidence preservation and transaction collaboration capabilities inherent in blockchain systems provide technical solutions for the protection and transaction of intellectual property. Consensus algorithms, as the core technology of blockchain, aim to ensure the system's consistency, security, and stable operation [9]. PBFT (Practical Byzantine Fault Tolerance) [10] is a consensus algorithm that effectively addresses the Byzantine Generals Problem [11]. It is widely used in areas such as consortium chains [12] and distributed databases. The PBFT algorithm divides the nodes into primary nodes and normal nodes. One node in the system will be treated as the primary node and all other nodes are child nodes. All the nodes in the system communicate with each other and the ultimate goal is that everyone can reach a consensus by majority principle and the PBFT algorithm ensures that the system can be consistent and secure, despite the presence of a faulty node or a node with malicious behaviour.

Along with the increasing development of blockchain technology, the possibility of its application in intellectual property has been discovered by various parties. In the exploration of applying blockchain technology to intellectual property trading platforms, Niu et al. [13] developed a digital music copyright protection and trading system using the Hyperledger Fabric platform. They stored music feature fingerprints on the InterPlanetary File System (IPFS) and stored the hash addresses returned by IPFS on the blockchain. This method overcomes the limitations of blockchain scalability and storage space costs, while utilising robust and covert audio watermarking technology for proof of originality and rights protection. Fang et al. [14] designed a supply chain transaction system based on blockchain technology to achieve the more effective management of transaction information, funds, and outcomes, as well as upstream and downstream collaboration. Cao et al. [15] proposed a distributed trading platform capable of automatic transaction settlement. They utilised the ant colony algorithm to solve the Nash equilibrium problem and introduced a reputation scoring system to regulate and constrain buyers and sellers. The study in [16] proposes a blockchain-based patent registration and transaction system, effectively addressing some key challenges in patent protection. However, it offers limited designs related to patent transactions and does not present a specific solution for intellectual property transactions based on blockchain technology. The study in [17] proposes a business system based on the SPV (Simplified Payment Verification) business model to carry intellectual property rights, adopting the "business-transaction dual-chain" architecture, which effectively solves business challenges such as trustworthy data and traceable behaviour. However, the system still faces issues such as low transaction efficiency and imprecise transaction matching. While blockchain-based intellectual property protection projects have achieved certain results in practical applications, they still encounter challenges and

limitations, including issues related to technological maturity, privacy protection, legal regulations, and so on [18].

In terms of intellectual property management and protection based on blockchain technology, the study in [19] proposes a code copyright management system based on blockchain, which consists of full nodes and light nodes, providing better response speed and storage efficiency. The authors of [20] proposed a credit-based intellectual property protection model with an improved PBFT consensus mechanism, by using blockchain and smart contract technology to address the problems of difficulty in registering property rights and chaotic transactions of digital works at the current stage, in which the traditional database is used to store the complete content of the work, while the blockchain is used to store data such as the unique abstracts and property rights information of the digital work. The study in [21] proposes an intellectual property management model based on Ethernet smart contracts, to address the pain point problem in the existing intellectual property management scheme, whereby information interaction can only be determined on the platform and the transfer of rights and interests still needs to be carried out offline. The authors of [22] propose a blockchain-based privacy protection and traceability intellectual property identity management scheme to achieve security and traceability identity management and authentication of intellectual property rights. Li et al. [23] propose a distributed application copyright management and transaction system based on Ether, which utilises smart contracts to achieve the automatic generation and processing of transactions according to protocols. The authors of [24] designed and constructed a digital music copyright management system by using the VNT Chain blockchain platform, which adopts blockchain, the interplanetary file system (IPFS), and MySQL as the storage engine, according to the different needs of business data, while adopting the Shazam algorithm as the feature fingerprint extraction and similarity comparison algorithm for music files.

In summary, scholars have carried out a lot of work on the use of blockchain technology for intellectual property transactions and their management and protection, all of which are of great practical significance. However, there are very few studies on intellectual property transactions combined with a blockchain consensus mechanism; there are even fewer studies on consensus algorithm improvements for actual intellectual property transaction application scenarios, but there are more theoretical studies on a separate intellectual property problem point or for the whole industry problem. With the change in market demand, the performance of intellectual property transaction systems based on blockchain technology becomes a constraint, such as communication complexity, consensus delay, throughput, and so on. In this paper, a consensus algorithm D-PBFT, based on DIANA improvement, is proposed for the actual IPR transaction scenarios of the coalition chain. The contributions of this paper are as follows:

(1) Design a fair and intuitive node reward and punishment mechanism. Use the hierarchical clustering of the DIANA algorithm to divide nodes into clusters based on similarity and divide the nodes into grades through a reward and punishment mechanism to achieve consistency of node status and authority, timely evaluation, and feedback of node behaviours.
(2) Improve the primary node selection mechanism. Classify nodes dynamically, construct agent nodes and alternate nodes, and adopt a majority voting strategy to achieve the reliability of primary node selection.
(3) Optimise the consistency protocol process. Consensus is carried out once each within both the cluster and among all primary nodes, to ensure the accuracy of the consensus results.
(4) Experimentally evaluate the D-PBFT algorithm and compare it with the PBFT algorithm, in terms of communication complexity, throughput, and consensus delay.

The next pages are organised as follows: Section 2 describes the detailed design scheme of D-PBFT, Section 3 presents an experimental analysis of the performance of the algorithm, and Section 4 concludes the paper.

2. Overall Design of the D-PBFT Algorithm

The D-PBFT consensus algorithm consists of the following three parts: node similarity clustering, reward and punishment mechanism, and primary node selection. The node similarity is used to make nodes with higher node similarity become a cluster, which makes the division between nodes more accurate, avoids all nodes participating in consensus at the same time, reduces the communication cost of the system, and improves the transaction consensus efficiency. The reward and punishment mechanism makes the nodes have a grade division among them and the nodes of different grades have different rewards and punishments, different node statuses, and different corresponding permissions, so that node behaviours can be evaluated and fed back in a timely manner and node motivation can be improved. The clusters are divided into classes through the reward and punishment mechanism so that the nodes are divided into agent nodes and alternate nodes; the nodes with higher credibility are selected as the primary nodes, which ensures the reliability of the primary nodes and reduces the probability of the Byzantine nodes acting as the primary nodes. The consensus process is shown in Figure 1.

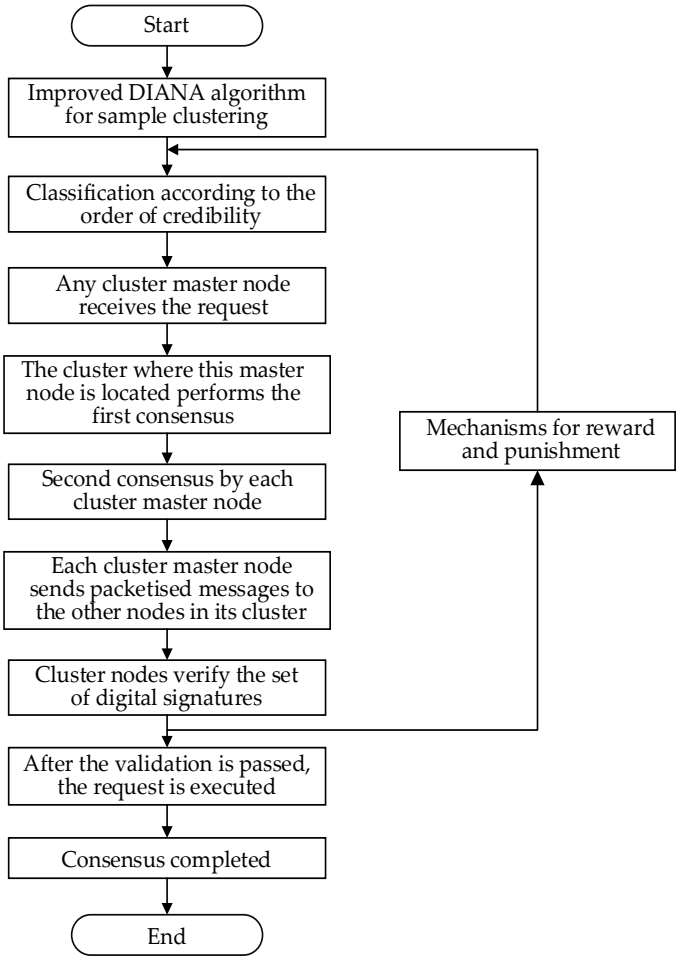

Figure 1. The process of D-PBFT.

Firstly, the nodes in the initial set of nodes provided by the intellectual property trading system are divided into clusters according to node similarity, using the DIANA algorithm. Secondly, the nodes are classified into ranks in each of the resulting clusters, through a reward and punishment mechanism, and the primary node selection mechanism obtains the set of proxy and alternate nodes for each cluster and elects the primary node based on the proxy node voting. When there is a request in the system that needs to be processed, the cluster where the primary node that receives the request first carries out the first PBFT consensus; if the first PBFT consensus is successful, the second consensus

is carried out between all the primary nodes. If this consensus is successful, the whole transaction consensus ends and the intellectual property trading system transaction is successful. If the consensus fails to meet the requirements, the Byzantine node is replaced, the corresponding reputation penalty is given, and the transaction consensus process is restarted.

In practical applications, since this algorithm is based on PBFT, when Byzantine nodes (nodes that are down or experiencing other faults and cannot function properly) exist, the set of proxy nodes in each cluster needs to satisfy the condition $n > 3f + 1$ to ensure the correctness of the consensus results. In this case, n denotes the entire number of nodes and f is the maximum number of Byzantine nodes that may be accommodated within the complete node set, meaning that the total number of agent nodes is at least two-thirds of the total.

Suppose Figure 2 is a scenario of an intellectual property transaction. The two parties to the transaction submit an intellectual property transaction proposal to the blockchain network after consensus, including the specific content of the transaction, the participants, the transaction conditions, and other information. The transaction request is generated through a smart contract and sent to the blockchain network. Each node in the blockchain network will carry out consensus, confirm the validity of the transaction, and add the transaction block to the entire blockchain. The final transaction result will be fed back to both parties to ensure that both parties are aware of the result and status of the transaction.

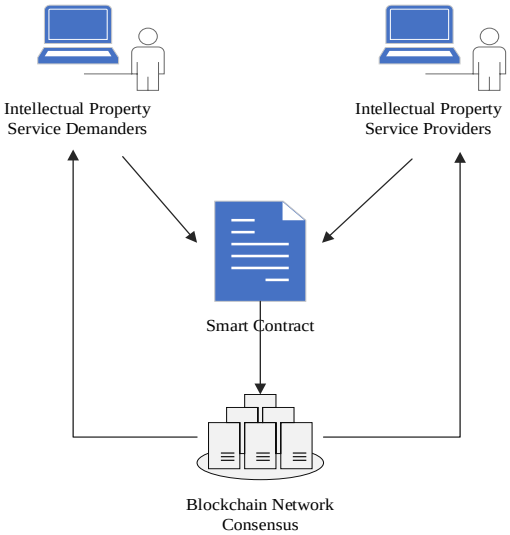

Figure 2. Intellectual property transaction scenarios.

The consensus process of the D-PBFT consensus algorithm for intellectual property transactions in blockchain networks is shown in Figure 3. The blockchain network receives the client transaction request and sends the transaction request to the primary node of any cluster; then, the primary node receives the client request and initiates the consensus of the transaction content within the cluster and sends the result of the successful consensus to the primary nodes of other clusters for the second consensus. After successful consensus, the primary node sorts and packages the objects to generate blocks for successful consensus transactions and sends the blocks to each cluster to ensure the consistency of each node's block ledger.

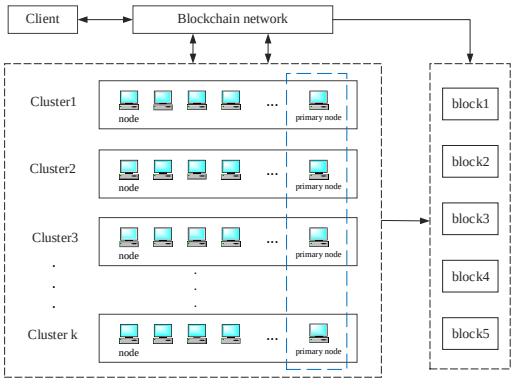

Figure 3. D-PBFT algorithm's general architecture.

2.1. DIANA Algorithm

The DIANA algorithm belongs to the hierarchical clustering algorithm and has a top-down splitting strategy. The idea of splitting is that, initially, all nodes are considered as a cluster and are split according to their similarity; the DIANA algorithm uses Euclidean distance to express the similarity of two nodes. The method of splitting is to first find the node with the smallest average similarity in the cluster as the starting point of the new cluster; then, in the old cluster, the algorithm is constantly looking for the node whose Euclidean distance to the new cluster is less than the Euclidean distance to the old cluster, which is divided into new clusters, and will keep on iterating until the number of clusters in the split reaches the specified value of k, whereby the algorithm is terminated.

Initialisation—all nodes are considered as one large cluster. Assume that there are n object nodes with p-dimensional features as a clustered dataset, denoted by $v_{\mathrm{i}} = \{v_{i1}, v_{i2}, v_{i3}, \ldots, v_{ip}\}(i = 1, 2, \ldots, n)$, a coordinate value of the ith node mapped in the p-dimensional feature space. The dataset $V = \{v_1, v_2, \ldots, v_n\}$ of all the above nodes can be expressed as shown in Equation (1):

$$V = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1p} \\ v_{21} & v_{22} & \vdots & v_{2p} \\ \vdots & \vdots & & \vdots \\ v_{n1} & v_{n2} & \cdots & v_{np} \end{bmatrix} \tag{1}$$

Equation (2) represents the distance function between two nodes v_{i} and v_{k}, as follows:

$$\mathrm{d}(v_i, v_k) = \|v_i - v_k\| = \sqrt{\sum_{j=1}^{p} \left(v_{ij} - v_{kj}\right)^2} \tag{2}$$

Select the least similar cluster—find the current least similar cluster and divide it into two sub-clusters. For the two newly generated sub-clusters, recalculate the similarity between them and the other clusters. Repeat the above steps to gradually split the cluster into more clusters, based on the similarity between the clusters, until a predetermined number of clusters is reached or a certain threshold of dissimilarity is reached.

The goal of the DIANA algorithm is to split the dataset into k clusters, each of which is denoted by V_{i}, node v_{i}, belonging to a cluster; V_{i} has a higher similarity to other nodes in the same cluster and a lower similarity to nodes in other clusters [25].

Determining the centroid node of objects within the same cluster is an essential aspect of clustering [26]. During iterations, the DIANA algorithm requires the repeated calculation of the Euclidean distance between all nodes within a cluster and other nodes, which increases the computational complexity of the algorithm. Considering the practical application scenario of intellectual property transactions on a consortium blockchain, there is instability in the number of nodes taking part in blockchain consensus, whereby the

quantity of nodes engaged in blockchain consensus is not constant and there is a disparity in hardware conditions among nodes. Additionally, the operating environment conditions are unstable. Consequently, in the context of intellectual property transactions, once the initial cluster centroid nodes are selected, it is undesirable to waste the computational resources of the consensus cluster by frequently changing the centroids within the cluster. Several improvements can be considered, based on the characteristics of consortium blockchains, to make the DIANA algorithm more suitable for the model of intellectual property transactions on a consortium blockchain.

Combining the clustering algorithm with a consortium blockchain involves treating the primary nodes of the consortium blockchain consensus process as the centroids of each cluster. Applying the DIANA algorithm to the intellectual property trading environment, the initial K centre nodes are not randomly selected from all the samples, but the nodes with higher credibility are screened out to become the initialised clustering centre nodes through the reward, punishment, and selection mechanism. The improved DIANA algorithm better controls the probability of replacing cluster centroid nodes, thereby ensuring the reliability of leadership nodes.

The design process of the DIANA algorithm (Algorithm 1) is as follows: When there are a sufficient number of nodes in the j-th cluster, the enough(j) function outputs true. If not, the procedure returns false, guaranteeing that the amount of nodes in the k clusters after applying the algorithm satisfies the likelihood under the Byzantine condition and meets the criterion n > 3f.

Algorithm 1 DIANA

```
Input:
        Data set D,
        Number of nodes n,
        Number of clusters k
Output:
        k clusters
1: Initialize all samples to a single cluster
2: for j in range(1, n/k)
3:     q = 1
4:     while q < k do
5:     Find the cluster with the largest diameter C
6:         if X is the sample with the largest average dissimilarity in C
7:         cls_new = {X}; cls_old = C − X
8:         Repeat
9:             for i = 1, 2, . . ., len(cls_old) do
10:            The Euclidean distance between cls_old[i] and cls_new is L
11:            The Euclidean distance between cls_old[i] and cls_old – cls_old[i] is R
12:                if L < R &&! enough(j) then
13:                cls_new = cls_new ∩ \cap ∩ cls_old[i]
14:                cls_old = cls_old – cls_old[i]
15:                end
16:            end
17:        end
18:    break for
19:    Until cls_old and cls_new no longer change
20:    Delete C, add cls_old and cls_new
21:    q = q + 1
22: end
```

2.2. Mechanisms for Incentives and Penalties

In this paper, the credibility, reward, and punishment mechanism is introduced into the consensus algorithm and the node with the highest credibility is selected as the master node, so as to make the intellectual property transaction system more secure and credible.

The credibility of the node is rated according to the behaviour of the consensus node in the consensus process and, according to the defined credibility threshold interval to assess what kind of state the node is in, the rewards and punishments for nodes in different grades, as well as different nodes in different states with different corresponding permissions, so that the node's behaviour can be evaluated and fed back in a timely manner, to ensure that the process of intellectual property transaction is highly transparent and to avoid the platform's falsification behaviour. Node trustworthiness is defined as the probability that a node correctly participates in all the consensus processes in the coalition chain. The higher the trustworthiness, the more likely the node is to be a normal node and the more likely it is to be selected as the master node and vice versa.

The node trustworthiness, reward, and punishment mechanism strategy is as follows: The maximum credibility value for a node is set at 100. Once the credibility value reaches 100, it ceases to increase further. When the credibility value of a node is equal to or greater than 90, the credibility level of the node is upgraded to "high". When the credibility value of a node is less than 90 but greater than 60, the credibility level of the node is downgraded to "medium". When the credibility value of a node is less than or equal to 60, the credibility level of the node is categorised as "low". The initial credibility of newly added nodes in the consensus model is defined as "normal", with a quantified credibility value of "60". When nodes engage with the algorithm of consensus, their credibility undergoes changes through adjustment mechanisms. Participants in the actual consortium blockchain can adjust the high credibility threshold and low credibility threshold based on the business requirements of intellectual property transactions, to control trust.

The reward mechanism is as follows:

$$R_{\mathrm{ij}} = \begin{cases} R_{ij}^{previous} + 0.1 \times N_1, R_{ij}^{previous} \geq 90 \\ R_{ij}^{previous} + 0.3 \times N_1, 60 \leq R_{ij}^{previous} < 90 \\ R_{ij}^{previous} + 0.6 \times N_1, R_{ij}^{previous} \leq 60 \end{cases} \tag{3}$$

The penalty mechanism is as follows:

$$R_{\mathrm{ij}} = \begin{cases} 60, R_{ij}^{previous} \geq 90 \\ R_{ij}^{previous} - N_2, 60 \leq R_{ij}^{previous} < 90 \\ R_{ij}^{previous} - N_3, R_{ij}^{previous} < 60 \end{cases} \tag{4}$$

In the above Equations (2) and (3), R_{ij} denotes the worth of the reputation of the j-th consensus node in the i-th cluster at the current moment and $R_{ij}^{previous}$ denotes the worth of the reputation of the j-th consensus node in the i-th cluster at the previous moment. The details about the selection method of the consensus node will be elaborated in the next section; N_1 indicates that the node completes the consensus correctly and is a trusted node and the trustworthiness, subsequently, increases by a corresponding value, as a reward for each correct completion of consensus by a node of this class. N_2, N_3 means that the node has made an error in consensus and is a Byzantine node and for each error made by this type of node, the trustworthiness is reduced by the corresponding value as a penalty. N_1, N_2, N_3 can be changed according to the actual application scenarios; in this paper, $N_1 = 1$, $N_2 = 15$, and $N_3 = 10$. In particular, when the primary node becomes a Byzantine node, the current credibility value is directly reduced to 60 and the trustworthiness is reduced to "low,", which restricts it from participating in the election of consensus nodes for a period of time.

2.3. Selection of Primary Nodes

In the scenario of intellectual property transactions, the consensus mechanism's nodes are separated as agent nodes and alternate nodes. The agent node is preferred to be selected as the primary node and participates in all the consensus processes; the alternate node

is composed of nodes with lower initial points and nodes with errors in the consensus process, which do not participate in the consensus, but need to save the consensus results. The states of the nodes can be converted to each other.

Nodes with credibility "high" are selected as the agent node group. Nodes with credibility "medium" and "low" are selected as the alternate node group. According to the actual application requirements, the proportion of nodes in each group will be adjusted appropriately. Still, the number of agent nodes should not be less than 2/3 of the total number of nodes. Nodes in the "high" category have the highest credit rating and are given priority to act as primary nodes. When the "high" nodes have been selected, or there are no "high" nodes, the primary node can be selected from the "medium" nodes. The "low" nodes are not suitable to be the primary nodes and cannot participate in the consensus due to their low credit level and the "normal" nodes can neither choose the primary nodes nor participate in the consensus. Privilege classification can significantly improve nodes' enthusiasm and effectively prevent malicious nodes from becoming primary nodes. This effectively reduces the communication loss caused by malicious nodes participating in consensus and enhances the system's efficiency. Distinguish node permissions are shown in Table 1.

Table 1. Classification of node privileges.

Credit Level	Priority as a Primary Node	Act as a Primary Node	Act as a Slave Node
high	yes	yes	yes
medium	no	yes	no
low	no	no	no
normal	no	no	no

The primary node of each cluster is determined by the trustworthiness of the node and, in the initial state, all nodes are scored and ranked based on the composite behaviour, which consists of the node's historical behaviour and reputation. In this consensus algorithm, the clusters are divided according to the rules of the DIANA algorithm, the nodes in each cluster are sorted in descending order according to their trustworthiness, and, after the division of the clusters, the upward rounding function is used to derive the group of agent nodes in each cluster.

The agent node with the highest and second-highest trustworthiness ranks is chosen as the backup primary node from the group of agent nodes. Next, the agent nodes are voted on and the agent node with the highest value at the conclusion of the voting process is designated as the primary node. The selection method of the primary node improves the reliability of the primary node, while the probability of the Byzantine node becoming the primary node is greatly reduced; at the same time, it improves the enthusiasm of the nodes to participate in the election of the consensus node and it also reduces the degree of centralisation of the intellectual property transaction system. It reduces the number of nodes participating in consensus in the consensus process of intellectual property transactions, reduces the communication overhead, and improves the consensus efficiency.

2.4. Optimising Conformance Agreement

The traditional PBFT algorithm divides the processing of a request into the following three stages [27]: pre-preparation, preparation, and submission. In each phase, nodes receive messages from other nodes, which are processed and forwarded according to certain rules. Figure 4 shows the flowchart of the consistency protocol interaction for the PBFT algorithm.

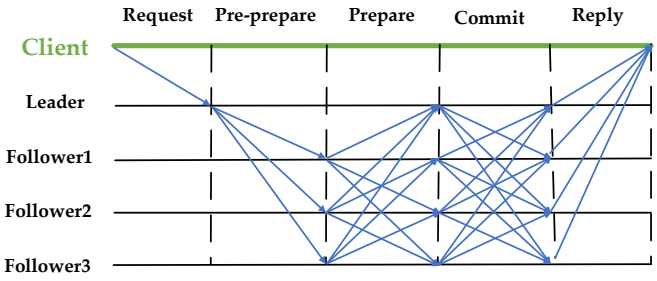

Figure 4. PBFT conformance protocol flow.

The D-PBFT consensus algorithm in the intellectual property transaction scenario contains two consensus phases, as follows: the first consensus consists of a request phase, a pre-preparation phase, a preparation phase, and a submission phase, while the second primary node consensus consists of a pre-preparation phase, a preparation phase, a submission phase, and a feedback phase. Figure 5 shows the flowchart of the conformance protocol interaction of the algorithm.

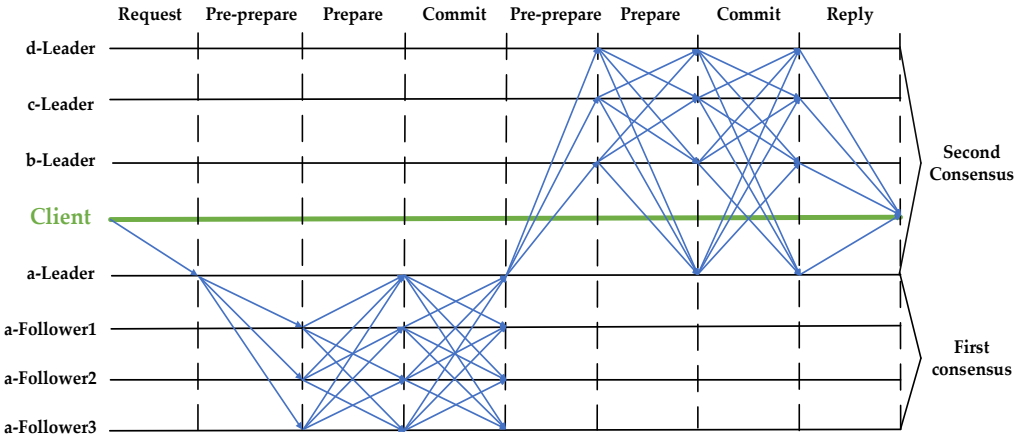

Figure 5. D-PBFT conformance protocol flow.

Consensus implementation phase:

First intra-cluster consensus:

Request phase—the client sends the intellectual property transaction request message to the primary node of any cluster.

Pre-preparation phase—after the primary node receives the intellectual property transaction request, it first checks its correctness, generates a unique sequence number and records the request hash value locally after successful checking, and then generates a pre-preparation message to broadcast to other consensus nodes in the cluster where it is located; then, other consensus nodes start to verify the validity of the message after receiving the message and record the message locally.

Preparation phase—when the primary node receives the same pre-prepared message from 2f + 1 consensus nodes in the cluster, it enters the preparation phase, where f is the number of Byzantine nodes. Then, all the consensus nodes in the cluster broadcast the ready message to other nodes, indicating the acceptance of the sequence number request.

Commit phase—each consensus node enters the commit phase when it receives the same prepare message from 2f + 1 consensus nodes in the cluster, where all consensus nodes broadcast a commit message to the other nodes, to synchronise the state of all nodes in the cluster. The first intra-cluster consensus is considered successful when each consensus node receives the same commit message from 2f + 1 consensus nodes in the cluster.

Second inter-master node consensus:

Pre-preparation phase—after the first successful consensus, the primary node of the cluster broadcasts the generated pre-preparation message to other primary nodes. Other

primary nodes receive the message and start to verify the validity of the message and record the message locally.

Preparation phase—A primary node enters the preparation phase when it receives 2f + 1 identical pre-prepared messages from other primary nodes. Then, all the primary nodes broadcast a ready message to the other primary nodes, indicating the acceptance of the sequence number request.

Commit phase—each primary node enters the commit phase when it receives the same prepare message from 2f + 1 other primary nodes, where each primary node takes the collected set of digital signatures, as well as the intellectual property transaction request, and generates a packetised message to broadcast to the other nodes in its cluster, to notify the other nodes in the cluster of the successful consensus and to synchronise the status of all nodes.

Response phase—after receiving the packaged message from its primary node, the nodes in each cluster verify the digital signature collection, in order to ensure that the transaction request has passed the second consensus. After the verification passes, the content of this transaction request is executed and, finally, the result is returned to the client and the whole network starts to update the ledger to ensure the consistency of the data and the completion of the intellectual property transaction.

When a fresh node seeks to be added, the cluster with maximum similarity is selected, by calculating the similarity of the node with each cluster, joining the cluster when there is no transaction consensus within the cluster, and classifying the node into categories based on its initial trustworthiness. The pseudo-code of the D-PBFT consensus algorithm (Algorithm 2) is as follows:

Algorithm 2 D-PBFT

```
Input:
      transaction request
      v is the view number, p is the sort number, and r is a summary of the client request
      message m.
Output:
      consensus to k cluster
1: any cluster primary node receives the messages received from the client;
2: send a request message to the next level node;
3:    while request valid = true do
4:       send pre-prepare message;
5:       number1 = 1;
6:       if pre-prepare valid = true then
7:          number1 = number1 + 1;
8:          if number1 > 2f + 1 then
9:              send pre-prepare message to other cluster primary nodes;
10:           number2 = 1;
11:              if pre-prepare valid = true then
12:                 number2 = number2 + 1;
13:                 if number2 > 2f + 1 then
14:                     send commit certificate;
15:                 end
16:                 if commit valid = true then
17:                     send reply to client;
18:                 end
19:              end
20:           end
21:       end
22:    end while
23:    send consensus request to k cluster;
24: end
```

3. Experimental Analysis

Intellectual property rights include patents, soft writings, theses, etc. Here, we take patents as an example of intellectual property transaction consensus. Communication complexity, system fault tolerance, transaction throughput, and delay are analysed to contrast the outcomes from optimisation with those from earlier.

3.1. Communications Complexity Analysis

Communication complexity is the number of messages exchanged between nodes during the process of reaching a conformance agreement. In the knowledge property transaction consensus system, there are a total of n nodes. The communication complexity for a patent to perform a transaction is displayed in Table 2.

Table 2. PBFT communication complexity.

Algorithm	Pre-Preparation Phase	Preparation Phase	Submission Phase	Communications Complexity
PBFT	$n-1$	$(n-1)$ $(n-1)$	n $(n-1)$	$2n$ $(n-1)$

Using the D-PBFT algorithm, The communication complexity analysis of a patent performing one transaction is shown in Table 3.

If the trading system's n nodes are split up into k groups, each group will have n/k nodes, resulting in the following communication complexity:

Table 3. D-PBFT communication complexity.

D-PBFT Algorithm	Pre-Preparation Phase	Preparation Phase	Submission Phase	Communications Complexity
First consensus	$n/k-1$	$(n/k-1)$ $(n/k-1)$	n/k $(n/k-1)$	$2n/k$ $(n/k-1)$ + $2k$ $(k-1)$
Second consensus	$k-1$	$(k-1)$ $(k-1)$	k $(k-1)$	

Then, the communication complexity of the patent transaction can be calculated using the D-PBFT algorithm $T1 = 2n/k\ (n/k-1) + 2k\ (k-1)$; by comparison, it can be seen that the communication complexity of this algorithm for patent transaction T1 is less than the communication complexity of the ND-PBFT algorithm, mentioned in the study in [28], T2, $T2 = 2\ (n/k + k)\ (n/k + k - 1)$.

It is established that the ND-PBFT algorithm has a lower communication complexity than the original PBFT. The algorithm in this study has a lower communication complexity than the ND-PBFT algorithm, which means that it has a significantly lower communication complexity than the original PBFT algorithm. If the nodes participating in the consensus of each cluster in the actual intellectual property trading system are less than n/k, the actual communication complexity will be even smaller. The communication complexity gap in the intellectual property rights transaction system will widen as the number of nodes increases.

The independent variable are set as the number of nodes within the network for a patent to conduct a transaction and the nodes are divided into four fixed groups. The experiment sets the number of nodes as 16, 20, 24, 28, 32, 36, and 40 and compares the communication complexity of the two algorithms to conduct a patent transaction by increasing the number of nodes. While adjusting the number of nodes, it is always ensured that the number of consensus nodes in D-PBFT, ND-PBFT, and PBFT is the same. The final experimental results are shown in Figure 6.

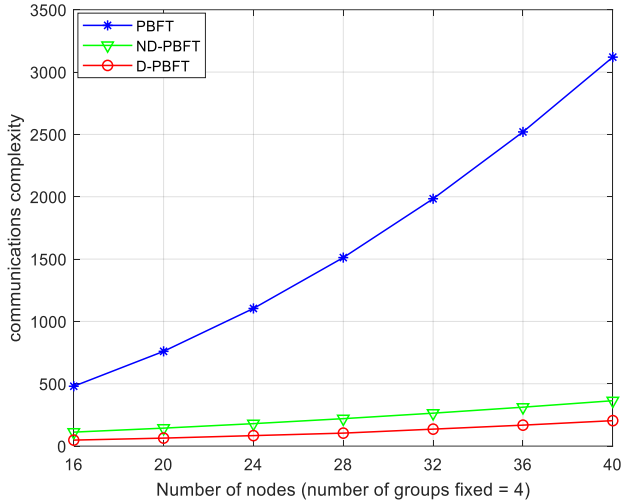

Figure 6. Comparison of the relationship between the number of nodes and communication complexity (number of groups fixed = 4).

The independent variables are set as the number of groups in a system where a patent conducts a transaction; specifically, they are set to 4, 5, 6, 7, 8, 9, 10. The total number of nodes is fixed at 40 and it is important to always ensure that the number of consensus nodes in D-PBFT, ND-PBFT, and PBFT is the same. Figure 7 displays the outcomes of the experiment.

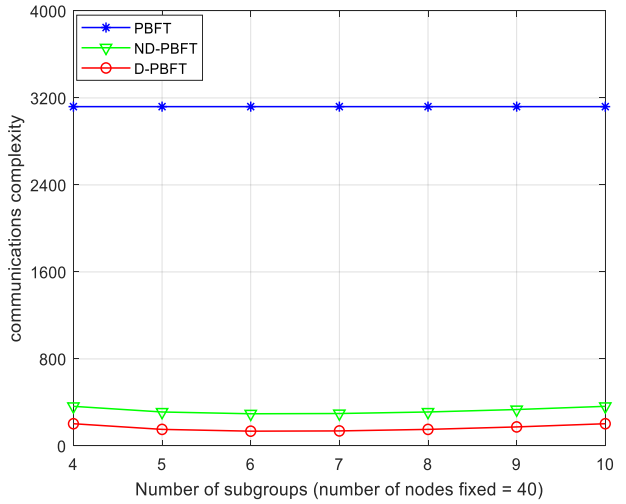

Figure 7. Comparison of the relationship between the number of groups and communication complexity (number of nodes fixed = 40).

In distributed systems, nodes need to communicate frequently with each other and a lower communication complexity means a higher efficiency and a faster speed. The communication complexity of the D-PBFT algorithm is much less than that of the PBFT algorithm and the ND-PBFT algorithm, in both cases of a fixed number of groups and a fixed number of nodes. And with the increase in nodes, the communication complexity of the PBFT algorithm grows exponentially and that of the D-PBFT algorithm grows linearly, so the D-PBFT algorithm is more suitable for an intellectual property trading environment.

3.2. Throughput Analysis

Throughput is expressed in blockchain as the number of transactions packed per unit of time and is generally expressed as *TPS*. The expression for throughput is as follows:

$$TPS = \frac{transations}{\Delta t} \tag{5}$$

The experiment sets the client to send 2000 transaction requests and records the number of transactions per second that can complete the consensus. The independent variable is set as the number of nodes, as 16, 20, 24, 28, 32, 36, and 40, and by adding more nodes, the throughput of the two approaches is compared. It is important to always make sure that there is the same number of consensus nodes in PBFT and D-PBFT, when changing the number of nodes. Figure 8 displays the final findings of the experiment.

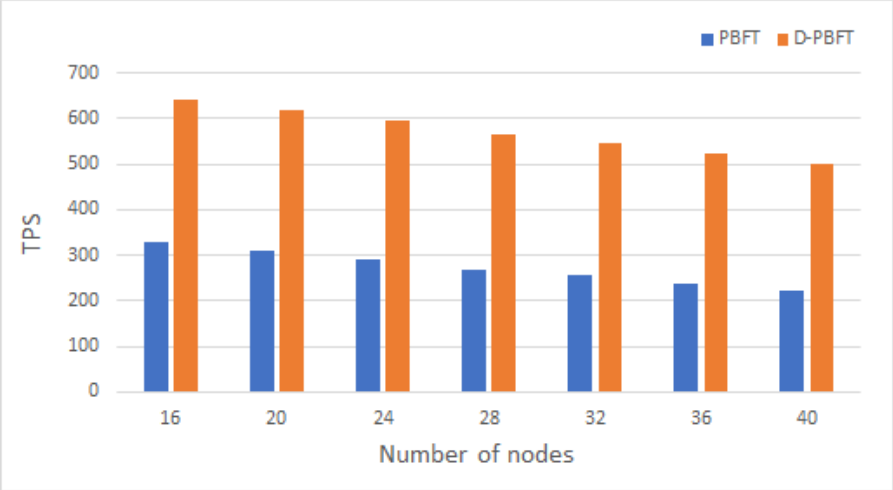

Figure 8. Comparison of the relationship between the number of nodes and throughput.

The higher throughput of the consensus algorithm means that the system can handle more transaction requests. As the number of nodes increases, the throughput of both the D-PBDT and PBFT algorithms shows a decreasing trend. However, in the whole process, the throughput of the D-PBFT algorithm is always greater than the throughput of the PBFT algorithm. In the intellectual property trading system, a high throughput can ensure that the system can process a large number of transaction requests in time, improve the efficiency of transaction processing, and shorten the time of transaction confirmation, so as to improve the overall performance of the system and the user experience. The D-PBFT algorithm is able to deal with more transactions under the same number of nodes and, thus, is more suitable for the intellectual property trading environment.

3.3. Time Delay Analysis

Latency is the time interval between when a transaction is submitted and when the network determination takes effect. It is a measure of network performance and consensus algorithm runtime. In a blockchain system without forks, a transaction is uploaded to the chain, which means that the transaction is valid; however, in a blockchain system with forks, it is necessary to wait for some time to indicate that the transaction is determined to be valid. The lower the latency, the higher the efficiency. The formula for calculating latency is as follows:

$$T_{delay} = T_{txbroadcast} + T_{consensus} + T_{blockbroadcast} \tag{6}$$

where $T_{txbroadcast}$ denotes the time from transaction generation to the time the transaction is received by the consensus node, $T_{consensus}$ denotes the consensus time of the transaction,

and $T_{blockbroadcast}$ denotes the consensus block broadcast time. The independent variable is set as the number of nodes in the system of a patent conducting a transaction, the nodes are divided into four fixed groups, the number of nodes is experimentally set as 16, 20, 24, 28, 32, 36, and 40, and the transaction consensus communication complexity of the two algorithms is compared, by increasing the number of nodes. Adjusting the number of nodes always ensures that the number of consensus nodes in D-PBFT and PBFT are the same. Figure 9 displays the outcomes of the experiment.

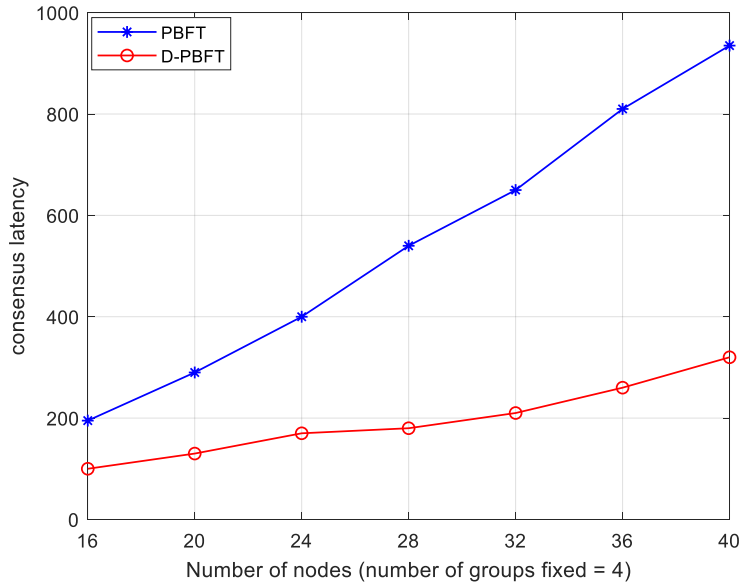

Figure 9. Comparison of the relationship between the number of nodes and the consensus delay (number of groups fixed = 4).

The independent variable is set as the number of groups in a system where a patent undergoes consensus once; specifically, they are set to 4, 5, 6, 7, 8, 9, 10. The total number of nodes is fixed at 40, and it is important to always ensure that the number of consensus nodes in D-PBFT and PBFT is the same. Figure 10 displays the outcomes of the experiment.

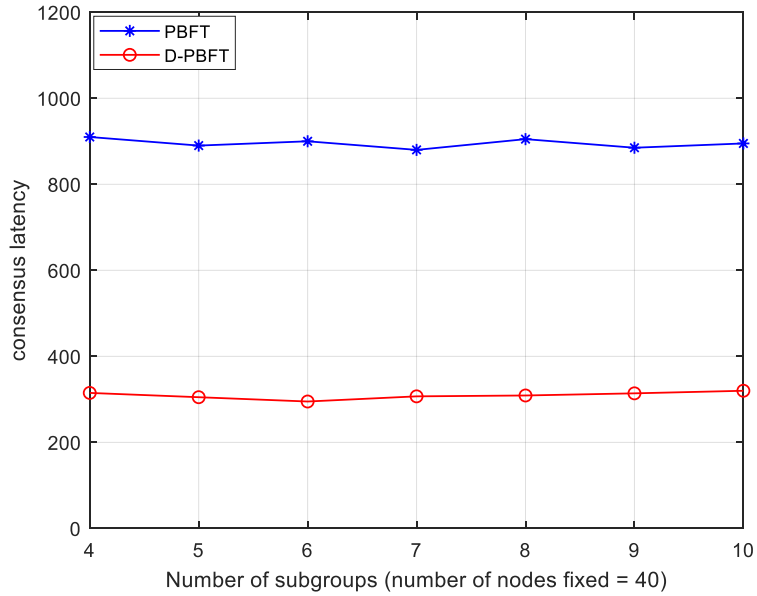

Figure 10. Comparison of the relationship between the number of clusters and consensus latency (number of nodes fixed = 40).

In intellectual property trading systems, lower latency ensures the quick confirmation and execution of transactions, reducing transaction wait times and improving user experience and satisfaction. In addition, smaller latency also reduces the transaction risk and reduces the uncertainty and potential loss caused by excessive latency. The delay of the D-PBFT algorithm is significantly better than that of the PBFT algorithm, in both cases of a fixed number of groups and a fixed number of nodes. With the increase in the number of nodes, the delay of the PBFT algorithm increases faster, while the delay of the D-PBFT algorithm increases slower and is more stable, which improves the operational efficiency of the system, so the D-PBFT algorithm is more in line with the needs of intellectual property trading systems.

4. Summary and Outlook

Aiming at the shortcomings of the traditional PBFT consensus algorithm in the intellectual property transaction scenario, such as large communication overhead, random master node selection leading to the election of Byzantine nodes as the master node, and prolonged consensus time, a secure and efficient PBFT improvement algorithm, D-PBFT, is proposed. Firstly, the nodes in the system are divided into a number of clusters by the similarity degree, through the DIANA hierarchical clustering mechanism, so as to make the division of the nodes more accurate. Secondly, the nodes in each cluster are divided into ranks through the reward and punishment mechanism and the master node and other sets of nodes in each cluster are selected using the master node selection mechanism, which ensures the reliability of the master node, reduces the probability of the Byzantine node acting as the master node, and improves the motivation of the nodes. Finally, the consistency protocol is optimised to carry out two consensus processes within the cluster and between the master nodes; the consensus is completed through the nodes with high trust to improve the consensus accuracy. The experimental results show that, under the same conditions, the D-PBFT algorithm compares with the traditional PBFT algorithm and there is a big improvement in communication complexity, consensus delay, and transaction throughput, which is more suitable for intellectual property rights transaction scenarios, improves the transaction processing speed and throughput, reduces the time of transaction confirmation, ensures the consistency and reliability of intellectual property rights transactions, and makes intellectual property rights transactions more efficient. In addition, the application of blockchain technology in intellectual property transactions promotes its innovation and development. Blockchain technology provides a high degree of transparency and traceability for intellectual property transactions through distributed ledgers and tamper-proof records, which can effectively prevent intellectual property theft and infringement and can provide better protection for intellectual property rights holders. Transaction costs are greatly reduced by removing middlemen, streamlining processes, and automating execution.

In future work, we will continue to study the characteristics of blockchains for intellectual property application scenarios, improve the reward and punishment and selection mechanism in the algorithm, consider optimising the algorithm by aggregating signatures, and increase the function of nodes dynamically joining and exiting the network, so as to promote the solution to the efficiency problem of the participation of a large number of network nodes in the consensus in the intellectual property transaction system, promote the determination of the value of intellectual property rights and the enhancement of the liquidity, and provide better technical support for intellectual property rights.

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

Funding: This work was supported by the Key R&D Project of Hainan Province, under Grant ZDYF2024GXJS024, and the Funding for Academician Team Innovation Center in Hainan Province.

Data Availability Statement: The data used to support the findings of this study are included within the article.

Acknowledgments: The authors would like to thank the editors and the reviewers for their valuable time and constructive comments.

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

References

1. Yin, J.; Ge, Z.; Song, W. Research on the Construction of Intellectual Property Operation Platform under the Background of "Internet +". *Technol. Investig.* **2017**, *8*, 179–194. [CrossRef]
2. Cheng, J. Research on intellectual property transaction innovation system—Conceptualisation of "blockchain intellectual property transaction platform". *Technol. Mark.* **2020**, *27*, 85–86.
3. Yu, Z. Design and Implementation of SOA-Based Intellectual Property Transaction System. Master's Thesis, Tianjin University, Tianjin, China, 2014.
4. Jin, J. Research on the Design of Intellectual Property Service System Based on Service Design. Master's Thesis, East China University of Science and Technology, Shanghai, China, 2019.
5. Xia, Y.; Sheng, G. Trust Relationship Analysis of Knowledge Property Cloud Trading System Based on Blockchain Principles. *Financ. Account. Mon.* **2021**, *2021*, 7. [CrossRef]
6. Sun, X. Risk origins and avoidance of intellectual property cloud transactions. *People's Forum* **2020**, *23*, 116–117.
7. Yuan, Y.; Wang, F. Current situation and prospect of blockchain technology development. *J. Autom.* **2016**, *42*, 481–494.
8. Fu, C. Framework of Decentralized Storage System Based on Distributed Ledger. Ph.D. Thesis, University of Electronic Science and Technology, Chengdu, China, 2020.
9. Yuan, Y.; Ni, X.; Zeng, S. Development status and prospect of blockchain consensus algorithm. *J. Autom.* **2018**, *44*, 2011–2022.
10. Castro, M.; Liskov, B. Practical Byzantine fault tolerance. In *Symposium on Operating Systems Design & Implementation*; USENIX: Berkeley, CA, USA, 1999; pp. 173–186.
11. Lamport, L.; Shaostak, R.; Pease, M. The Byzantine generals problems. In *Concurrency: The Works of Leslie Lamport*; ACM: New York, NY, USA, 2019; pp. 382–401.
12. Zheng, Z.; Xie, S.; Dai, H.; Chen, X.; Wang, H. An overview of blockchain technology: Architecture, consensus, and future trends. In Proceedings of the IEEE International Conference on Big Data, Honolulu, HI, USA, 25–30 June 2017; pp. 557–564.
13. Niu, X.; Han, D.; Sun, Z. Music copyright protection and transaction system based on consortium blockchain. *Comput. Appl. Res.* **2022**, *39*, 6.
14. Fang, Y.; Zhou, C.; Lei, X. Design of supply chain transaction system based on blockchain technology. *Comput. Eng.* **2021**, *47*, 23–31.
15. Cao, X. Design and Implementation of Distributed Energy Trading Platform Based on Blockchain Technology. Ph.D. Thesis, Beijing University of Posts and Telecommunications, Beijing, China, 2021.
16. Hu, J.; Zhu, P.; Qi, Y. A patent registration and trading system based on blockchain. *Expert Syst. Appl.* **2022**, *201*, 117094. [CrossRef]
17. Liu, Y. Research and Implementation of Intellectual Property Asset Management and Trading System Based on Blockchain. Ph.D. Thesis, Xi'an University of Electronic Science and Technology, Xi'an, China, 2020.
18. Zhang, S.; Tian, C.; Li, B. A review of identity authentication based on blockchain technology. *Comput. Sci.* **2023**, *50*, 329–347.
19. Jing, N.; Liu, Q.; Sugumaran, V. A blockchain-based code copyright management system. *Inf. Process. Manag.* **2021**, *58*, 102518. [CrossRef]
20. Sun, J.; Meng, X.; Zhang, H.; Chang, X. A blockchain intellectual property protection model based on improved PBFT. *Comput. Eng.* **2020**, *46*, 134–141.
21. Zhang, H.; Du, R. AI Clock. Research on intellectual property management model based on Ethernet smart contract. *Sci. Technol. Manag. Res.* **2021**, *41*, 164–169.
22. Zhuang, C.; Dai, Q.; Zhang, Y. BCPPT: A blockchain-based privacy-preserving and traceability identity management scheme for intellectual property. *Peer–Peer Netw. Appl.* **2022**, *15*, 724–738. [CrossRef]
23. Li, Y.; Wei, J.; Yuan, J. A decentralized music copyright operation management system based on blockchain technology. *Procedia Comput. Sci.* **2021**, *187*, 458–463. [CrossRef]
24. Zhang, G.; Tang, H.; Chen, J. Blockchain-based digital music copyright management system. *Comput. Appl.* **2021**, *41*, 945–955.
25. Li, W.; Mao, Y.; Peng, X. Agglomerative hierarchical clustering algorithm based on hesitant fuzzy sets. *Comput. Appl.* **2023**, *43*, 3755–3763.
26. Xu, X.; Ding, S.F.; Ding, X. Survey on density peaks clustering Algorithm. *J. Softw.* **2022**, *33*, 1800–1816.

27. Sun, H.F.; Zhang, W.F.; Wang, X.M. A robust Byzantine fault-tolerant consensus algorithm against adaptive attack based on ring signature and threshold signature. *Acta Autom. Sin.* **2023**, *49*, 1471–1482.
28. Zhu, H.; Jin, Y. DS-PBFT: A distance-based consensus algorithm for blockchain. *J. Microcomput. Syst.* **2022**, *43*, 8.