Hindawi
Mathematical Problems in Engineering
Volume 2021, Article ID 5542078, 7 pages
https://doi.org/10.1155/2021/5542078

*Research Article*

# Optimization of PBFT Algorithm Based on Improved C4.5

**Xiandong Zheng, Wenlong Feng, Mengxing Huang, and Siling Feng**

*School of Information and Communication Engineering, Hainan University, Haikou 570228, China*

Correspondence should be addressed to Wenlong Feng; fwlfwl@163.com

Received 6 January 2021; Revised 9 February 2021; Accepted 20 February 2021; Published 4 March 2021

Academic Editor: Leandro F. F. Miguel



Aiming at the problems of PBFT algorithm of consortium blockchain, such as high communication overhead, low consensus efficiency, and random selection of leader nodes, an optimized algorithm of PBFT is proposed. Firstly, the algorithm improves C4.5 and introduces weighted average information gain to overcome the mutual influence between conditional attributes and improve the classification accuracy. Then classify the nodes with improved C4.5, and select the ones with a high trust level to form the main consensus group. Finally, the integral voting mechanism is introduced to determine the leader node. Experimental results show that compared with traditional PBFT algorithm, the communication times of the improved PBFT algorithm are reduced greatly, which effectively alleviates the problem that the number of nodes in traditional PBFT algorithm increases and the traffic volume is too large, and significantly reduces the probability of the leader node doing evil and improves the consensus efficiency.

## 1. Introduction

Blockchain technology was first proposed by Satoshi Nakamoto in the literature [1] in 2008. With the rapid development of science and technology, blockchain technology has attracted much attention in recent years. Related applications based on blockchain are also emerging, involving many fields such as financial transaction [2], edge computing [3], energy [4], medical care [5], and data security [6–8]. As the consensus algorithm is the core content of blockchain technology, the consensus algorithm has been paid more and more attention by researchers. There are several existing consensus algorithms, such as workload proof (Pow), equity proof (PoS), share authorization proof DPoS, PBFT, and raft [9]. Among them, Pow, PoS, and DPoS algorithms are implemented in the public blockchain, the Raft algorithm is applied in the private blockchain, and PBFT is widely used in the consortium blockchain. Existing Byzantine fault-tolerant algorithms either have poor scalability or low fault-tolerant rate. Besides, the random selection of leader nodes will cause the switching of views and affect the whole consensus process. In view of the problems of the Byzantine fault-tolerant algorithm mentioned above, in [10–12], the leader node of the PBFT algorithm is tried to improve in two ways; in [10], the reputation module is introduced; and in [11, 12], credit mechanism and credit reward and punishment scheme are introduced. [13] adopts the separation of negotiation and execution to improve the problem of view change. Jeon et al. [14] propose a method to increase the number of user nodes entering the blockchain by grouping. In [15], the trust value is combined with the grouping mechanism. In [16], the signature algorithm is combined with PBFT algorithm. In [17], the DPoS consensus algorithm is combined with PBFT consensus algorithm to complement each other. In [18], nodes are clustered and layered by the clustering algorithm, which is improved from the node scale. At present, there are many methods to improve the practical Byzantine algorithm, but all of them focus on scalability (transaction throughput, node scalability, etc.), fault tolerance, and security [19], and pay less attention to the communication times and consensus efficiency between nodes. In this paper, we design a PBFT-optimized algorithm based on improved C4.5, aiming to solve the problems of excessive communication times and low consensus efficiency among large-scale nodes.

The main contributions of this paper are summarized as follows:

(1) We improve the C4.5 algorithm and introduce weighted average information gain to overcome the mutual influence between conditional attributes and improve the classification accuracy.

(2) We propose a PBFT-optimized algorithm based on improved C4.5, which classifies the consensus nodes and improves the consensus efficiency of the leader nodes.

(3) We compare the optimized PBFT algorithm with the traditional PBFT algorithm and analyze the communication complexity and node security in detail.

The rest of this paper is arranged as follows: Section 2 introduces the improved design of the C4.5 algorithm. In Section 3, the optimized PBFT algorithm based on the C4.5 algorithm is studied. Section 4 presents the experimental comparison and analysis with the traditional PBFT algorithm, and Section 5 is the conclusion of this paper.

## 2. Improved Design of the C4.5 Algorithm

*2.1. Overview of the C4.5 Algorithm.* The C4.5 decision tree [20–22] is a classical decision tree algorithm, which determines the features to be classified by calculating and comparing the information gain rate of each feature, and its basic flow is as follows:

(1) Let the sample set $S$ have $s$ training samples, divide the sample set into $m$ classes, and the number of instances of Class $i$ is $S_i$, $(S_i/s)$ is the probability $P_i$, $\mathrm{Info}(S)$ is the category information entropy, and its calculation formula is

$$\mathrm{Info}(S) = -\sum_{i=1}^{m} p_i \log_2(p_i). \tag{1}$$

(2) If the selected feature $A$ is a split feature, the sample set $S$ is divided into $k$ subsets $\{S_1, S_2, \ldots, S_k\}$. Let feature $A$ have $k$ uncorrelated values $\{a_1, a_2, \ldots, a_k\}$. Then the number of training individuals that should be class $i$ in $S_j$ is $S_{ij}$. $\mathrm{Info}(S)$ is $a_i$ conditional information entropy. Its calculation formula is

$$\mathrm{Info}_A(S) = -\sum_{j=1}^{k} \frac{S_{1j} + S_{2j} + \cdots + S_{nj}}{S} * \mathrm{Info}(S_j), \tag{2}$$

where $\mathrm{Info}(S_j) = -\sum_{i=1}^{m} p_{ij} \log_2(p_{ij})$ and $p_{ij} = (S_{ij}/S_j)$ is sample probability of class $i$ in $S_j$.

(3) Calculate the information gain of conditional attribute $A$:

$$\mathrm{Gain}(A, S) = \mathrm{Info}(S) - \mathrm{Info}_A(S). \tag{3}$$

(4) The information entropy of attribute $A$ in sample $S$:

$$\mathrm{Info}(A) = -\sum_{j=1}^{k} p_j \log_2(p_j). \tag{4}$$

(5) The information gain rate of attribute $A$:

$$\mathrm{Gain-Ratio}(A) = \frac{\mathrm{Gain}(A, S)}{\mathrm{Info}(A)}. \tag{5}$$

*2.2. Improvement of the C4.5 Algorithm.* The C4.5 algorithm uses the information gain to select the test attribute when generating the decision tree. The higher the information gain, the stronger the correlation between the attribute and the class attribute, and the higher the possibility that the attribute is selected as the test attribute. The C4.5 algorithm considers the impact of attributes on class attributes but does not consider the mutual influence between conditional attributes. Combined with the specific application scenario of blockchain consensus mechanism, when the trust level is classified according to the node attribute values, the node condition attributes influence each other, and the classification accuracy will decrease. To improve the accuracy of node trust classification, the concept of weighted average information gain is introduced, as shown in Algorithm 1.

$$w\_\mathrm{avgGAIN}(A_I) = \frac{\sum (\mathrm{Info}(A) - \mathrm{Info}_I(A))}{n} \cdot \omega_i, \tag{6}$$

where $\mathrm{GAIN}(A_I)$ is the sum of information gains of other attributes to attribute $A$, indicating the degree of association between attribute $A$ and other attributes, $I$ represents all conditional attribute sets except feature $A$, and $\omega_i$ represents the importance of each conditional attribute, $0 \le \omega_i \le 1$, $\sum \omega_i = 1$.

The improved information gain rate is

$$\mathrm{Gain-Ratio}(A) = \frac{\mathrm{Gain}(A, S)}{\mathrm{Info}(A) + w\_\mathrm{avgGAIN}(A_I)}. \tag{7}$$

The lower the correlation between attribute $A$ and other attributes, the lower the $w\_\mathrm{avgGAIN}(A_I)$, the higher the information gain rate, and the higher the probability of being a split attribute, thus overcoming the influence of the relationship between conditional attributes on the classification accuracy.

## 3. Optimization of PBFT Algorithm Based on Improved C4.5

Optimized PBFT algorithm based on improved C4.5 mainly includes optimization of PBFT algorithm and leader node selection strategy based on an integral voting system. The optimization of PBFT algorithm mainly reduces communication times and improves consensus efficiency by grouping consensus nodes after being classified by the C4.5 algorithm. The leader node selection strategy based on the integral voting system is based on the comprehensive evaluation of node behavior and vote rate. The higher the integral value, the greater the probability of selecting the leader node. The algorithm design process is as follows:

```
Input: samples, adjustment weight
Output: decision tree
(1) Create root node N;
(2) if samples are empty then
(3)    Return the previous step;
(4)    else if sample is the same attribute then
(5)       Marked as leaf nodes and marked as the most class;
(6)       else if candidate attribute set is empty then
(7)          The class with the largest sample size
(8)       end
(9)    end
(10) end
(11) Calculate the weighted average information gain and information gain rate to determine the split point;
(12) Run the function continuously and recursively on a subset of data;
(13) Return to the root node to continue execution;
```

Algorithm 1: Optimizing the C4.5 algorithm.

Step 1: use the C4.5 algorithm to classify participating network nodes according to the level of trust

Step 2: select nodes with a high trust level as the main consensus group, and other nodes as subconsensus groups

Step 3: subconsensus group implements PBFT consensus

Step 4: select the leader node of the main consensus group

Step 5: the main consensus group implements the second PBFT consensus and broadcasts it to the subconsensus group

Step 6: the subconsensus group verifies and links the operation records

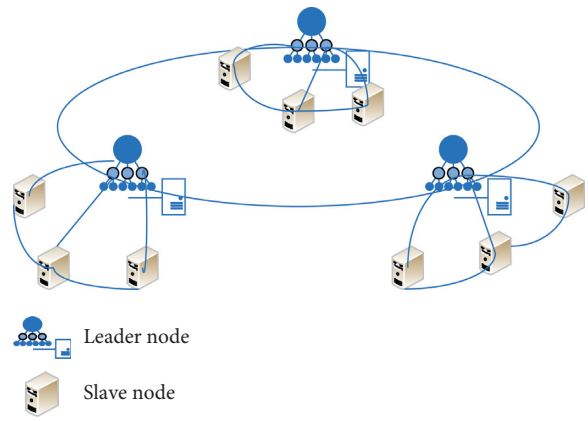


Figure 1: Consensus model.

*3.1. Optimization of PBFT Algorithm.* Combining the C4.5 algorithm with PBFT consensus mechanism, the decision tree algorithm is used to evaluate and classify the trust of participating nodes. It is assumed that there are $m$ nodes in the class with high trust, numbered as $\{0, 1, \ldots, M-1\}$ according to the level of trust. The first $K$ nodes with higher trust level are selected as the main consensus group, and each node in the group keeps a list, on which an address of the node in the group appears, which indicates that these nodes belong to the same group. There is an upper limit $(M/K)$ for the number of nodes in this group. The blockchain consensus model is shown in Figure 1.

When new node $a$ joins, it issues its request with a public key $P$ and then enters a waiting state. After receiving the request information from the new node, the leader node $b$ in the network determines that the number of nodes in its group has not reached the upper limit, attaches a timestamp to the joining information and the public key $MS$ of its node, encrypts it with $P$, and transmits it to the main consensus group in the form of a message. Other nodes in the main consensus group may receive multiple messages at the same time after receiving the determined message, with the earliest timestamp. $b$broadcasts the message to the whole group of nodes after receiving it, updates the node list, and synchronizes the latest node list to node $a$, and then node $a$ synchronizes all data of blocks in the whole network.

The consensus process of optimizing the PBFT algorithm is shown in Figure 2. The detailed operation process is as follows:

(1) Subconsensus group consensus: the slave node in the group sends the request to the leader node. After receiving requests for a period of time, the leader node packages several requests into a block, and then broadcasts the block to its group for PBFT consensus.

(2) Consensus of the main consensus group: after the block passes the consensus verification process of the subconsensus group, the second PBFT consensus confirmation will be conducted in the main group. $K$ nodes in the main group elect a leader node through the integral reward and punishment mechanism. If there is no new request from the subgroup within this time, the node will package an empty block and send it to the consensus uplink of the main group and then proceed to the next election.

(3) Submission stage: after the block has passed the consensus of the main group, all the leader nodes will

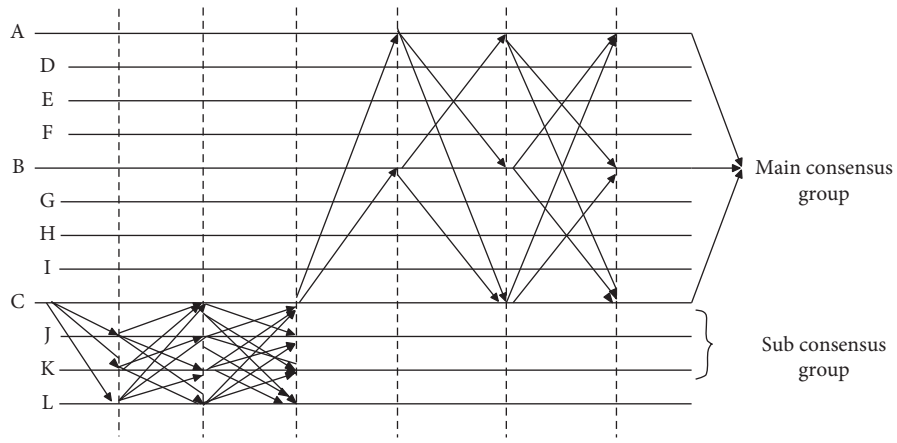


Figure 2: The consensus process of optimizing the PBFT algorithm.

digitally sign this block and collect digital signatures from other leader nodes, indicating that they agree with the authenticity and validity of this block. Then, the block attached to the digital signature set of the main consensus group is packaged into a submission message and broadcast to all the slave nodes in the subgroup to which it belongs, indicating that the block can be uplink.

(4) Execution stage: after receiving the submission message from the leader node, the slave node verifies the digital signature set attached to the block and can judge whether the block has passed the consensus verification of the main group. If the verification fails, it can be considered that the leader node to which the slave node belongs has malicious behavior, and this illegal operation can be reported, so as to achieve the role of upward supervision from the slave node. If the verification is successful, the requested content of this block can be executed, and the block can be recorded and uploaded.

*3.2. Leader Node Selection Based on the Integral Voting Mechanism.* In the main consensus group, to further reduce the probability of malicious nodes becoming leader nodes, the leader nodes are selected by the integral voting mechanism. The voting results are calculated as follows:

$$T = V_m * \alpha + V_i * \beta, \quad (\alpha + \beta = 1), \tag{8}$$

where $T$ is the final score, $V_m$ is the number okf votes, $V_i$ is the integral value, and $\alpha$ and $\beta$ are different correction parameters set according to the change of node integral value.

(1) Calculation of the integral value: when the system is initialized, each node will be assigned an integral value of 50 points. In a cycle, if a node successfully produces a block and is verified to be valid, the system will reward 10 points for the integral value. If the node fails to complete the block discharging within the specified time, the system will deduct 10 integral values. The integral value will gradually accumulate with the behavior of nodes. At this time, in order to avoid the excessive gap between the rich and the poor, the integral threshold is set to 200 points, and the system resets its integral value to 50 after exceeding 200 points.

(2) Regarding the maintenance of the integral schedule, select the node with the highest trust level in the consortium blockchain as the authoritative node to maintain the integral table. After the node succeeds in consensus, the hash value of the successful consensus content will be used as evidence to bind the integral, and the authoritative node will verify the hash value. If the verification passes, it will accumulate the integral for the node. If the verification fails, it will deduct the integral according to the node address. Finally, before the next stage of consensus starts, the establishment of the integral system in the integral table maintenance process will be completed.

(3) Punishment mechanism: a consensus node needs to pay a deposit when entering the main consensus group. If the system finds that the node has committed evil acts, the deposit will be confiscated to punish the node.

## 4. Results and Discussion

*4.1. Operational Efficiency Analysis*

*(1) Calculate the communication times of PBFT single consensus.* In the consistency protocol process of the PBFT algorithm, there are three periodic broadcasts. Assuming that the number of nodes in the whole network is $M$, the number of times of message transmission is $T = 2M(M - 1)$.

*(2) Calculate the communication times of the improved PBFT single consensus.* After the block is submitted, the PBFT consensus process will be conducted in the subconsensus group. Therefore, the communication frequency is $(2M/k(M/K - 1))$. After the subconsensus group reaches

a consensus, the leader node will conduct PBFT consensus on this block again in the main consensus group. The communication frequency of this consensus is $2K(K-1)$. After the consensus verification of the main consensus group passes, all the leader nodes will broadcast this block to their subordinate nodes to inform that this block has passed the consensus process and can be stored in uplink. The communication times of this process are $(M/(K-1))K$. Therefore, it can be calculated that the total communication times required by the consensus process are as follows: $(2M/K(M/(K-1))) + (2K(K-1)) + (M/(K-1))K$.

In the practical application scenario of consortium blockchain, the number of groups $K$ is determined in advance; that is, $K$ is a constant. The influence of the group number $K$ on the single communication volume of the improved PBFT algorithm is shown in Figure 3. When the number of groups $K = 10$ is determined, the influence of the change in the number of consensus nodes on the communication times of the traditional PBFT algorithm and the optimized PBFT algorithm is shown in Table 1.

From the above results, it can be seen that with the increase of the number of consensus nodes, the communication times of the traditional PBFT algorithm increase sharply. Compared with the improved PBFT algorithm, it can be seen that the improved PBFT algorithm can best reduce the communication times and has better advantages in the application scenario of consortium blockchain with large-scale nodes.

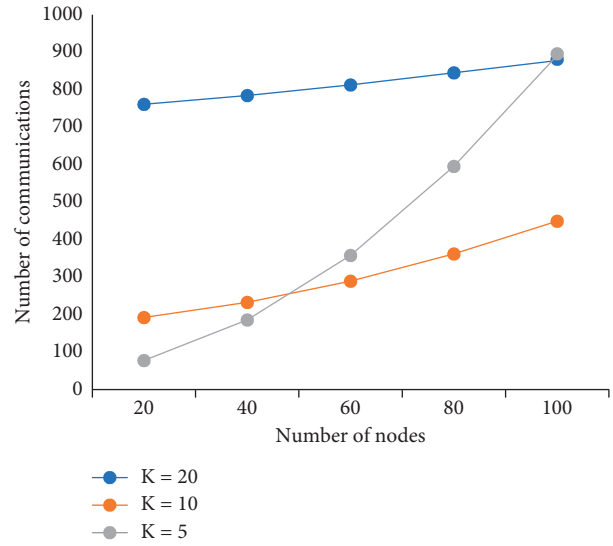


Figure 3: The influence of $K$ on the communication times of optimized PBFT.

Table 1: Comparison of communication times between PBFT and optimized PBFT.

| | 20 | 40 | 50 | 80 | 100 |
|---|---|---|---|---|---|
| PBFT | 760 | 3120 | 7080 | 12640 | 19800 |
| Optimized PBFT | 194 | 234 | 290 | 363 | 450 |

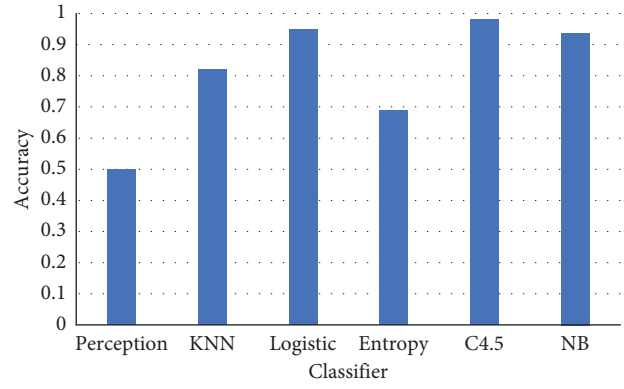


Figure 4: Comparison of classifier accuracy.

### 4.2. Security Analysis

*4.2.1. Security Analysis of Consensus Nodes of the Main Consensus Group.* In this paper, the C4.5 algorithm is used to classify the trust level of the nodes participating in the consortium blockchain network consensus. The top $K$ nodes are selected as the main consensus group. In the experiment, iris data set is selected, and the C4.5 algorithm, KNN algorithm, Naive Bayes algorithm, Logistic, perceptron, and maximum entropy classifiers are used to compare and observe the results shown in Figure 4. There are 300 experimental data in this paper, including 240 training samples and 60 test samples.

According to the experimental results, it can be concluded that the C4.5 decision tree has the best classification accuracy, and using the C4.5 decision tree can effectively improve the security of consensus nodes.

*4.2.2. Security Analysis of the Leader Node of the Main Consensus Group.* Based on the C4.5 algorithm, an integral voting mechanism is introduced to further enhance the honesty probability of the leader node. With the constant change of the node integral value, the higher the integral value, the more reliable the node will be, and the higher the honesty probability of the selected leader node will be, which can effectively avoid the possibility of mistakes of the leader node. In order to verify that the mechanism can effectively improve the honesty probability of the leader node, this section selects the Monte Carlo method for experimental analysis. Assuming that the number of repeated experiments is $m$, the consensus number of each experiment is $n$, and the frequency of the leader node being an honest node is $f$, and the frequency $P(A)$ of the selected leader node being an honest node in each repeated experiment is

$$P(A) = \frac{f}{n}. \tag{9}$$

When the number of $n$ is large enough, $P(A)$ will probabilistically converge to the theoretical probability $P(A_i)$ of the $i$-th experiment:

$$\lim_{n \longrightarrow \infty} P\left(\left|\frac{f}{n} - P(A_i)\right| < \varepsilon\right). \tag{10}$$

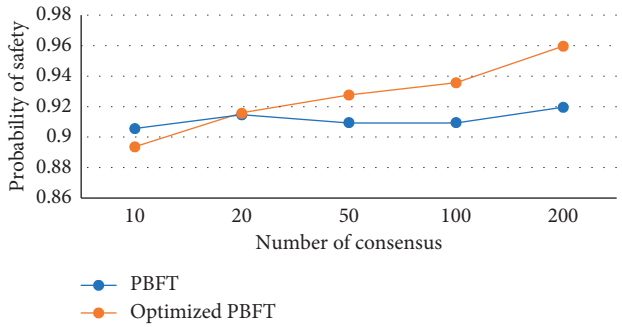


Figure 5: Security of the leader node of algorithm.

Then the frequency with which the selected leader node is an honest node can be used as an approximation of the probability of honesty of the leader node, by taking the mean value of the probability of $m$ repeated experiments as the final expected probability, namely,

$$\overline{P} = \frac{1}{m}\sum_{i}^{m} P(A_i). \tag{11}$$

The security of the algorithm is analyzed and verified through simulation experiments on the PBFT algorithm and the optimized PBFT algorithm. The security probability of the leader node is obtained by simulating the consensus number of 10, 20, 50, 100, and 200 times, and the average result of each simulation is shown in Figure 5.

The security selection probability of the leader node of the PBFT algorithm is as follows: $\overline{P}(A_1) = (0.906 + 0.915 + 0.910 + 0.910 + 0.921)/5 = 0.9124$. The security selection probability of the leader node of the optimized PBFT algorithm is as follows: $\overline{P}(A_2) = (0.894 + 0.916 + 0.928 + 0.935 + 0.96)/5 = 0.9266$. Obviously, $\overline{P}(A_2) > \overline{P}(A_1)$, the point voting mechanism can increase the probability that the leader node is an honest node, can effectively avoid the possibility of errors in the leader node, and the entire system is more secure and stable.

*4.2.3. System Security Analysis.* The algorithm uses a double-layer verification mechanism. If an error occurs in a certain link, the entire system will roll back. If the related nodes involved in the subconsensus group reject the operation, the main consensus group can get feedback immediately and feed it back to the corresponding main consensus group node. If a node in the main consensus group rejects the operation, the nodes in the main consensus group will call back the operation failure and the reason to each subconsensus group, forming an upper and lower layer supervision mechanism, which greatly guarantees system security.

## 5. Conclusions

In this paper, an improved scheme based on the decision tree algorithm is proposed to solve the problems of large communication volume, random selection of main nodes, and low consensus efficiency in PBFT consensus algorithm commonly used in consortium blockchain. According to the C4.5 algorithm, the consensus nodes of traditional PBFT algorithm are classified and sorted according to the trust level, which ensures the reliability of the leader node to a certain extent, reduces the number of views switching, and obtains higher consensus efficiency. At the same time, by grouping nodes, the global decentralization consensus is improved to a hierarchical multicentralization consensus, which effectively alleviates the problem that the number of nodes increases and the traffic is too large due to the simple use of the PBFT algorithm. The next step is to study the uncertainty of network nodes, hoping to dynamically sense the changing state of the number of nodes.

## Data Availability

The data used to support the finding of this study are included within the article.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This work was supported by the National Key Research and Development Project under Grant 2018YFB1404400 and National Natural Science Foundation of China under Grant 62062030.

## References

[1] S. Nakamoto, *Bitcoin: A Peer-To-Peer Electronic Cash System*, HN Publishing, Guldborg, Denmark, 2008.

[2] S. Anwar, V. K. Shukla, S. S. Rao, B. K. Sharma, and P. Sharma, "Framework for financial auditing process through blockchain technology, using identity based cryptography," in *Proceedings of the 2019 Sixth HCT Information Technology Trends (ITT)*, pp. 99–103, November 2019, https://doi.org/10.1109/ITT48889.2019.9075120.

[3] Y. Yan, Y. Dai, Z. Zhou, W. Jiang, and S. Guo, "Edge computing-based tasks offloading and block caching for mobile blockchain," *Computers, Materials & Continua*, vol. 62, no. 2, pp. 905–915, 2020.

[4] M. H. D. Khan, U. Mujahid, M. Najam-Ul-Islam, and H. Choi, "A security analysis of blockchain based decentralized energy exchange system," in *Proceedings of the 2020 9th International Conference on Industrial Technology and Management (ICITM)*, pp. 251–255, Oxford, UK, February 2020.

[5] M. S. Christo, A. Anigo Merjora, G. Partha Sarathy, C. Priyanka, and M. Raj Kumari, "An efficient data security in medical report using block chain technology," in *Proceedings of the 2019 International Conference on Communication and Signal Processing (ICCSP)*, pp. 606–610, Chennai, India, April 2019.

[6] Q. Wang, F. Zhu, S. Ji, and Y. Ren, "Secure provenance of electronic records based on blockchain," *Computers, Materials & Continua*, vol. 65, no. 2, pp. 1753–1769, 2020.

[7] J. Wang, W. Chen, L. Wang, R. Simon Sherratt, O. Alfarraj, and A. Tolba, "Data secure storage mechanism of sensor networks based on blockchain," *Computers, Materials & Continua*, vol. 65, no. 3, pp. 2365–2384, 2020.

[8] H. Chen, W. Wan, J. Xia et al., "Task-attribute-based access control scheme for iot via blockchain," *Computers, Materials & Continua*, vol. 65, no. 3, pp. 2441–2453, 2020.

[9] X. Cai, Y. Deng, Sean et al., "Blockchain principle and its core technology," *Chinese Journal of Computers*, vol. 42, pp. 1–51, 2019.

[10] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Toward secure blockchain-enabled internet of vehicles: optimizing consensus management using reputation and contract theory," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2906–2920, 2019.

[11] W. Fang, Z. Wang, H. Song, Y. Wang, and Y. Ding, "An optimized PBFT consensus algorithm for blockchain," *Journal of Beijing Jiaotong University*, vol. 43, no. 5, pp. 58–64, 2019.

[12] Y. Wang, S. Cai, C. Lin et al., "Study of blockchains's consensus mechanism based on credit," *IEEE Access*, vol. 7, pp. 10224–10231, 2019.

[13] Z. Han, N. Gong, and J. Ren, "Improvement of a practical Byzantine fault-tolerant algorithm for blockchain," *Computer Applications and Software*, vol. 37, no. 2, pp. 226–233, 2020.

[14] S. Jeon, I. Doh, and K. Chae, "RMBC: randomized mesh blockchain using DBFT consensus algorithm," in *Proceedings of the 2018 International Conference on Information Networking (ICOIN)*, pp. 712–717, Chiang Mai, Thailand, January 2018.

[15] S. Gao, T. Yu, J. Zhu, and W. Cai, "T-PBFT: an EigenTrust-based practical Byzantine fault tolerance consensus algorithm," *China Communications*, vol. 16, no. 12, pp. 111–123, 2019.

[16] F. Yi, J. Deng, L. Cong, and C. Liu, "An improved PBFT blockchain consensus algorithm based on ring signature," *Computer Engineering*, vol. 45, no. 11, pp. 32–36, 2019.

[17] Y. Wu, P. Song, F. Wang, and R. Aguilar-Lopez, "Hybrid consensus algorithm optimization: a mathematical method based on POS and PBFT and its application in blockchain," *Mathematical Problems in Engineering*, vol. 2020,13 pages, 2020.

[18] Z. Chen and L. Qiang, "Improved PBFT consensus mechanism based on K-medoids," *Computer Science*, vol. 46, no. 12, pp. 101–107, 2019.

[19] W. Yang, E. Aghasian, S. Garg, D. Herbert, L. Disiuta, and B. Kang, "A survey on blockchain-based internet service architecture: requirements, challenges, trends, and future," *IEEE Access*, vol. 7, pp. 75845–75872, 2019.

[20] B. Xu, D. Huang, and B. Mi, "Research on E-commerce transaction payment system basedf on C4.5 decision tree data mining algorithm," *Computer Systems Science and Engineering*, vol. 35, no. 2, pp. 113–121, 2020.

[21] F. Es–Sabery and A. Hair, "An improved ID3 classification algorithm based on correlation function and weighted attribute," in *Proceedings of the 2019 International Conference on Intelligent Systems and Advanced Computing Sciences (ISACS)*, pp. 1–8, Taza, Morocco, December 2019.

[22] M. Wirawan, T. Widiyaningtyas, and N. B. Siti, "Nutritional status of infants classification by calculating anthropometry through C4.5 algorithm," in *Proceedings of the 2019 International Conference on Electrical, Electronics and Information Engineering (ICEEIE)*, pp. 216–219, Denpasar, Bali, Indonesia, October 2019.