Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 187 (2021) 341–346

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Identification, Information and Knowledge in the internet of Things, 2020

An Improved Delegated Proof of Stake Consensus Algorithm

Qian Hu[a], Biwei Yan[b,*], Yubing Han[a], Jiguo Yu[a,c,d]

[a]School of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250353, PR China
[b]School of Mathematical Sciences, Qufu Normal University, Qufu, 273165, PR China
[c]Shangdong Computer Science Center(National Supercomputer Center in Jinan, Jinan, 250014, PR China
[d]Shandong Laboratory of Computer Networks, Jinan, 250014, PR China

Abstract

Aiming at the problems of the existing DPoS(Delegated Proof of Stake) consensus algorithm, such as low enthusiasm of voting nodes and difficulties in dealing with malicious nodes, we improve the traditional DPoS consensus algorithm and propose a reputation-based delegated proof of stake consensus algorithm, called Reputation-DPoS. In our Reputation-DPoS, the reputation model is introduced. By evaluating the behavior of nodes, nodes are divided into different trusted states, and high-quality nodes in the network are selected as consensus nodes to reduce security risks and improve efficiency. Besides, incentive methods of reputation and token are used to improve the enthusiasm of nodes to participate in voting. Simulation results show that our Reputation-DPoS can reduce the probability of malicious nodes being selected and optimize the state of nodes in DPoS. Nodes with good behavior will get more votes and rewards, which will motivate nodes and improve the security of the system. Insert here your abstract text.

Keywords: Blockchain; Consensus Algorithm; DPoS; Reputation Model

1. Introduction

In 2008, Nakamoto proposed the concept of bitcoin, which was the first decentralized distributed e-cash system in the world[1]. In this system, it is no longer required for a centralized financial institution to act as an intermediary for people to connect. Even if there is a lack of trust between people, it can ensure consistency and ensure the security and reliability of the system. After more than a decade of development, the Bitcoin system has withstood the test, and more and more people have taken part in the system. However, the key to the success of the bitcoin system lies in its underlying core technology, the blockchain.

* Corresponding author. Tel.: +1-826-336-9480 .
E-mail address: for_yanbiwei@163.com

10.1016/j.procs.2021.04.109

Blockchain includes data encryption, timestamp, consistency algorithm, incentive mechanism, and other technologies. It can realize decentralized peer-to-peer transaction, coordination, and cooperation in a distributed system without mutual trust and centralized control. In essence, it is a public classifies ledger. Transactions submitted by nodes will be packaged in the form of blocks. Blocks are linked by hash values to form a long chain and finally, form a large ledger. The key characteristics of blockchain technology, such as the dispersion, persistence, anonymity, and suitability, can greatly save cost, improve efficiency, and ensure its security for data storage in a traditional centralized system. These characteristics of blockchain make it not only used in a variety of financial services but also can be applied to the Internet of things, medical, energy, transportation, and other fields [2] [3] [4] [5]. Du and Chen et al. built a supply chain financial service platform based on blockchain by applying blockchain to the field of supply chain finance, which solved the problems of distrust among supply chain participants, low efficiency of capital flow and information flow, and high cost in traditional supply chain finance [6]; To solve the problems related to the management of a large number of restricted Internet of things devices, Novo proposed an access control management system based on blockchain, and verified through experiments that the blockchain technology can be fully integrated into the Internet of things technology [7]; Zhao and Bai et al. analyzed that the blockchain has great development prospects in the medical field, but because the medical system involves a large amount of private data, it is necessary to establish a security mechanism based on key management, so they designed a suitable medical blockchain Key management scheme. The implementation results show that the program has high security and performance, can effectively protect the private information on the medical blockchain, and promote the application of the medical blockchain [8].

As the key factor of blockchain, the consensus algorithm solves the problem of the consistency of blockchain in distributed scenarios. When it comes to consensus algorithms, we have to mention the classical problem of distributed algorithms, the byzantine general problem [9]. The byzantine general problem is about the consistency of communication in the network. Here's a simple example to explain it. A group of Byzantine generals each led an army to attack a small country. However, because the troops are scattered in different places and far apart, they can only use messengers to transmit information. To be successful, all generals must reach a consensus. However, there will be traitors in the army, disrupting the decisions of the generals. Therefore, when there is a known traitor, the remaining loyal generals need to reach a consensus without being affected by the traitor.

The Byzantine fault-tolerant algorithm aims to solve the Byzantine Generals problem. After that, scholars continued to study and improve the algorithm, of which PBFT is a more famous representative. After the emergence of blockchain technology, as a representative of the byzantine fault-tolerant algorithm, PBFT was applied to the blockchain to solve the consistency problem between distributed nodes [10]. PoW is the earliest consensus algorithm adopted in blockchain, which successfully solves the byzantine general problem. However, due to the problems of computational power concentration and waste of resources in pow, scholars have proposed the PoS consensus algorithm. Then, aiming at the shortcomings of the PoW and PoS consensus algorithm, the DPoS consensus algorithm is proposed.

After evaluating the advantages and disadvantages of the common consensus algorithm, a Reputation-DPoS consensus algorithm is proposed, and the reputation and token incentives are added to improve the enthusiasm of voting nodes. Besides, the selection of proxy nodes is optimized. The experimental simulation tests the effectiveness of the improved scheme.

The remaining part of this paper is organized as follows. Section 2 introduces some rough consensus algorithms. Section 3 explains the Reputation-DPoS from the perspective of the reputation model, incentive mechanism, and the selection of the consensus node. Section 4 carries on the experimental simulation analysis. Section 5 gives a summary of this article.

2. Related Work

PoW[1] consensus algorithm requires nodes to solve a mathematical problem based on the SHA-256 hash function to compete for accounting rights. In this process, miners have made great efforts. For this reason, PoW is more secure. On the other hand, the pow consensus algorithm has the problems of high resource consumption and low throughput. To solve the problems in PoW, PoS[11] proposes to replace the workload by equity. The more equity a node owns, the easier it is to fight for accounting rights. PoS reduces resource consumption to a certain extent, but it still needs to be mined.

DPoS [12] is a further improvement of PoW and PoS, and it is a consensus algorithm based on voting elections, similar to the Democratic Congress. A certain number of representatives are elected by the holders of the currency to exercise their powers on their behalf. The elected representatives participate in consensus and generate blocks in turn. If the outgoing is incompetent, the voter can vote out. Although DPoS greatly improves throughput and reduces latency, there are some problems such as low enthusiasm of voting nodes and the inability of malicious nodes to be handled in time.

3. Reputation-DPoS

3.1. Reputation model

We evaluate the nodes by introducing a reputation model. According to the performance of the node in the current cycle, reward and punish the node. If the node performs well, the reputation value will gradually increase;otherwise, the reputation value will gradually decrease.The reputation value is a way to express the credibility of a node. In Reputation-DPoS, the reputation value R is a real number between 0 and 1. The larger the number, the higher the reputation. For the newly added system node, its reputation value is initialized to 0.5. Since the proxy node and other nodes behave differently in the consensus process, we have discussed two situations separately. We use $R_i(t)$ to represent the current reputation of the node S_i in the blockchain after the tth round of voting, then $R_i(t+1)$ can be specified as follows.

if node S_i is a proxy node,then

$$R_i(t+1) = \begin{cases} \min(1, (1+y)R_i(t)), & \text{if the block it generates is successfully added to the blockchain.} \\ xR_i(t), & \text{if the block is not generated on time.} \\ 0, & \text{if the node generates invalid blocks multiple times in a row.} \end{cases} \quad (1)$$

if not,then

$$R_i(t+1) = \begin{cases} \min(1, (1+y)R_i(t)), & \text{if the node actively participates in voting.} \\ xR_i(t), & \text{if the node votes for the node with lower reputation value (Abnormal and Error).} \\ 0, & \text{if the node sends invalid transactions multiple times in a row.} \end{cases} \quad (2)$$

Where $0<x<1$ *and* $0<y<0.03$

If the proxy node does not generate a new block within a specified time in the cycle, or other consistent nodes vote for a node with a lower reputation value, the reputation value of the corresponding node will decrease. In this case, the value of x determines the speed at which the reputation value decreases, which can be set according to the requirements of the application. When the detection proxy node continuously generates invalid blocks or other nodes send invalid transactions multiple times, the reputation of the node will directly drop to zero. When a proxy node generates a block and is successfully added to the blockchain, or other nodes actively participate in voting, the reputation of these nodes will gradually increase, and the rate of increase depends on the value of y.

The trusted state of a node is determined by the reputation value R. We distinguish the following four trusted states:

- Good: R belongs to [a, 1], and its trusted state value TS = 1;
- Normal: R belongs to [0.5, a), TS = 2;
- Abnormal: R belongs to [b, 0.5), TS = 3;
- Error: R <b, TS = 4.

Conditions that need to be met: 0.5 <a <1 , and 0 <b <0.5.

Figure. 1 is the diagram of node trusted state transition. The change of node state is mainly determined by the performance of the node in the cycle. When a new node is added to the system, the initial state of the node is Normal, and the reputation value is 0.5. The system adjusts its trusted state according to the behavior of the node. When the proxy node successfully generates a block or other nodes actively participate in voting, its reputation value will exceed the threshold a, and the node status is good, which has certain advantages in subsequent competition. If the proxy node fails to generate a block in time or other nodes vote for a node with a lower reputation value, the node state becomes

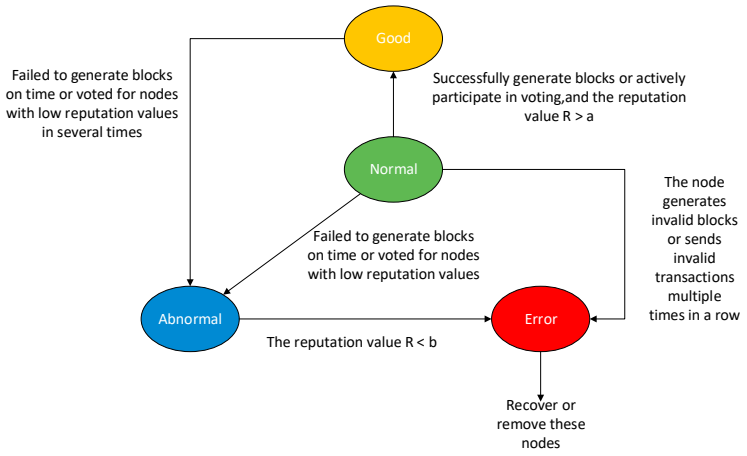

Fig. 1. Node state transition diagram.

abnormal due to the decrease in reputation value. When the proxy node continuously produces invalid blocks or other nodes send invalid transactions continuously and other irregular behaviors, the reputation value of the node continues to drop below b, and the node state becomes error, and the error node will return to the normal state or be removed after a period of time.

3.2. Incentive mechanism

The transaction fee incentive mechanism is common in the blockchain. In this part, based on the transaction fee incentive mechanism in the original blockchain based on the DPoS consensus algorithm, a certain number of rewards are assigned to the nodes based on reputation, and each node can get the corresponding proportion of rewards according to the trusted state value. Here is an example to illustrate that if node i successfully generates a block, the node's trusted state value TS is first obtained, and then the node is calculated to obtain the final transaction fee reward R. The value of R is equal to the original transaction fee reward F / TS.

Besides, this paper designs a reputation incentive mechanism.Through the transaction fee incentive mechanism, nodes selected as proxy nodes can get more rewards. But for the voting nodes, it is less attractive, and the enthusiasm to participate in the voting will be weakened. Therefore, this paper designs a reputation incentive mechanism. Incentive nodes actively participate in voting. Besides, it provides reputable stakeholders with the opportunity to enhance their creditworthiness and the possibility of earning a return on transaction costs. The more likely a node with a high reputation is to be selected as a consensus node, the higher the reward. We believe that the incentive measures are reflected by the changes in the reputation value of each node. The increase of reputation value makes the node obtain a good trusted state and improves the probability of the node being elected as a consensus node.

3.3. Selection of consensus nodes

In the DPoS consensus algorithm, currency holders vote for a certain number of representatives to exercise power on their behalf. The elected proxy nodes participate in the consensus and take turns to generate blocks. If a node does something bad, voters can vote him out of office. Also, since the voting weight is proportional to the account balance, it is easy to be in the hands of a few people, malicious nodes are easy to be selected as consensus nodes, and the long-term voting enthusiasm of other nodes will be weakened.

Therefore, in our Reputation-DPoS scheme, we take the reputation value of the node as a reference weight to select the delegation node, and we select a certain number of delegation nodes according to the voting results and reputation value. In addition, the current node's trusted state value is obtained according to the node reputation value. For nodes

with different reputation status, the actual number of votes obtained will also change accordingly. At the beginning of the next election cycle, the actual number of votes obtained by each node is calculated as follows:

$$S = V/TS \tag{3}$$

Among them, TS represents the trusted state value of the node, and V represents the number of votes obtained by the node. Rank according to the value of S, and finally select a certain number of nodes with the highest ranking.

4. Simulation

This article implements DPoS and Reputation-DPoS under the configuration of Intel i7-4712MQ CPU 2.30GHz processor, 8GB memory, 64-bit Ubuntu 19.04 system, based on the Ethereum platform. Compile them separately in the VSCode environment, and use the generated Ethereum client geth to build a private chain for experimental verification.

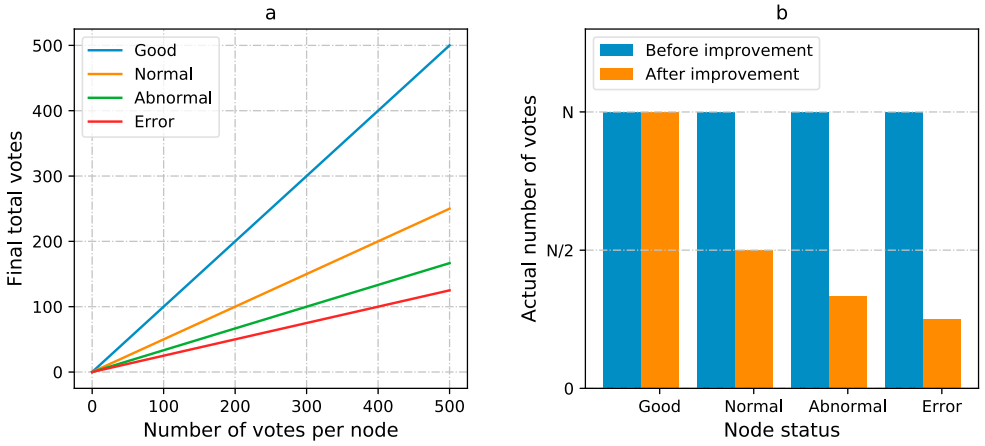

Fig. 2. (a) Changes of nodes in different states; (b) Comparison before and after improvement.

In order to verify the calculation method of the new voting result, for nodes with different credible states, when the number of votes obtained is the same, the node with a higher reputation value will eventually get a higher total number of votes. However, the node with a lower reputation value must become Proxy nodes that need more votes. In this scheme, we select four nodes with different reputation values, which correspond to different trusted states for verification.Node 1 (reputation value is 0.9, corresponding to credibility status is Good), node 2 (reputation value is 0.7, corresponding to credibility status is Normal), node 3 (reputation value is 0.4, corresponding to credibility status is Abnormal), node 4 (reputation value is 0.1, corresponding to credibility status is Error).Analyze the final total number of votes obtained by calculating the voting results of these four nodes under the same number of votes (0,100, 200, 300, 400, 500), the experimental results are shown in Figure. 2(a).

It can be seen from the experimental results graph that for nodes with different trusted states, under the condition of obtaining the same number of votes, nodes with lower reputation values need more votes to become proxy nodes. When the number of votes increases, nodes with good reputation will always be ahead of other nodes. Generally speaking, nodes with higher reputation values are more likely to be selected as proxy nodes. Unless other nodes get more votes, they have a chance to surpass the node with higher reputation value.

In Figure. 2(b) ,we compare Reputation-DPoS and DPoS. In DPoS, we can see that when the number of votes obtained by all nodes is N, the final number of votes for each node remains unchanged. However, in Reputation-DPoS, we divide the state of the nodes. Nodes are classified into Good, Normal, Abnormal, and Error states. Even if nodes receive N votes from voting nodes, the actual number of votes received will change. At the beginning of the next election cycle, the actual votes received by the nodes will be calculated and ranked based on this result, and finally the top fixed number of nodes will be selected as proxy nodes to participate in the generation and verification of the block.

5. Conclusion and future work

In this paper, we propose the Reputation-DPoS consensus algorithm, which can select high-quality proxy nodes from a large-scale blockchain distributed network to participate in consensus. Besides, we propose an incentive mechanism that encourages nodes to take an active part in voting and maintain good behavior. Finally, experimental results show that the algorithm reduces the probability of malicious nodes being selected and enhances the security of the system.

In the future, we will further optimize the reputation model and use sharding technology to further improve the throughput performance and security performance of the system.

Acknowledgements

This work was partially supported by the NNSF of China under Grants 61672321, 61771289 and 61832012.

References

[1] Satoshi Nakamoto (2019) "Bitcoin: A peer-to-peer electronic cash system."*Manubot.*

[2] Zibin Zheng, Shaoan Xie, Hongning Dai, Xiangping Chen, and Huaimin Wang. (2017) "An overview of blockchain technology: Architecture, consensus, and future trends." *In 2017 IEEE international congress on big data (BigData congress) IEEE*: 557–564

[3] Yong Yuan, Fei-Yue Wang (2018) "Blockchain and cryptocurrencies: Model, techniques, and applications."*IEEE Transactions on Systems, Man, and Cybernetics: Systems* **48**(9): 1421–1428.

[4] Leila Ismail, Huned Materwala (2019) "A review of blockchain architecture and consensus protocols: Use cases, challenges, and solutions."*Symmetry* **11**(10): 1198.

[5] Yue Wang, Jiguo Yu, Biwei Yan, Guijuan Wang, and Zhiguang Shan. (2020) "BSV-PAGS: Blockchain-based special vehicles priority access guarantee scheme." *Computer Communications* **161**: 28–40.

[6] Mingxiao Du, Qijun Chen, Jie Xiao, Houhao Yang, and Xiaofeng Ma. (2020) "Supply Chain Finance Innovation Using Blockchain." *IEEE Transactions on Engineering Management*.

[7] Oscar Novo (2018) "Blockchain meets IoT: An architecture for scalable access management in IoT."*IEEE Internet of Things Journal* **5**(2): 1184–1195.

[8] Huawei Zhao, Peidong Bai, Yun Peng, and Ruzhi Xu. (2018) "Efficient key management scheme for health blockchain." *CAAI Transactions on Intelligence Technology* **3** (2):114 –118.

[9] Leslie Lamport, Robert Shostak, and Marshall Pease. (2019) "The Byzantine generals problem." *In Concurrency the Works of Leslie Lamport*: 203–226.

[10] Miguel Castro, Barbara Liskov (1999). "Practical Byzantine fault tolerance."*In OSDI* **99**(199): 173–186.

[11] Sunny King, Scott Nadal (2012). "Ppcoin: Peer-to-peer crypto-currency with proof-of-stake." *self-published paper* **19**(1).

[12] Larimer Daniel (2014)."Delegated proof-of-stake (dpos)." *Bitshare whitepaper*.