2022 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA) | 978-1-6654-8090-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/AEECA55500.2022.9918852

Improved Scheme of Practical Byzantine Fault Tolerance Algorithm based on Voting Mechanism

Zhongxian Chen
School of Computer Science
University of South China
Hengyang, China
zxiann@126.com

Minsheng Tan*
School of Computer Science
University of South China
Hengyang, China
tanminsheng65@163.com
*Corresponding author

Peiliang Lei
School of Computer Science
University of South China
Hengyang, China
lplshagela@163.com

Abstract—**With the continuous development of cryptocurrencies, blockchain technology has received extensive attention, and the performance of blockchain systems is constrained by consensus algorithms. In view of the problems of high communication complexity and complex consensus process in PBFT algorithm, this paper An improvement scheme of the PBFT algorithm based on the voting mechanism is proposed. Before running the consensus protocol, the nodes in the network vote to generate a set of consensus nodes, and the remaining nodes are used as accounting nodes. At the same time, the consensus process of the traditional PBFT consensus algorithm is optimized. In the improvement scheme, the master node sends a Commit message to the consensus node and the accounting node to enter the Commit state. When the master node down, the consensus node with the highest number of votes is selected as the new master node, which reduces the probability of malicious nodes becoming master nodes and improves the system stability. The experimental results show that the improved scheme proposed in this paper reduces the communication overhead in the consensus process to a certain extent, also improves the consensus delay and throughput.**

Keywords—blockchain, consensus mechanism, PBFT algorithm, voting mechanism

I. Introduction

Bitcoin [1] was proposed by a scholar named Satoshi Nakamoto on November 1, 2008. According to the design ideas of Satoshi Nakamoto in the paper, Bitcoin is a digital form of currency built on a p2p network[2], using asymmetric encryption technology of cryptography to ensure the security in all circulation links. With the continuous development of digital currency, blockchain, as the underlying technology of digital currency, has received more and more attention and development. The consensus algorithm is the core of the blockchain [3] technology, and the efficiency of the consensus algorithm [4] determines the performance of the entire blockchain system. The purpose of the consensus algorithm is to reach consensus among all nodes as quickly and efficiently as possible. The current mainstream consensus algorithms include the workload proof algorithm PoW [5], the equity proof algorithm PoS [6], and the delegated equity proof algorithm DPoS[7] and practical Byzantine fault tolerance algorithm PBFT [8] and so on. Most of the public chains use PoW、PoS、Dpos and their deformation algorithms, and the PBFT algorithm is mainly used in the alliance chain.

The PBFT algorithm provides an efficient solution to the Byzantine generals problem [9]. The PBFT algorithm allows the existence of dishonest nodes in the network, but the system needs to control the number of dishonest nodes to no more than one-third of the total number of nodes to reach a consensus[10]. With the increase of the quantity of consensus nodes ,the communication overhead of the Practical Byzantine Fault Tolerance Algorithm will increase significantly. In order to optimize the problems of high bandwidth occupancy and high communication overhead, this paper proposes an improved scheme of practical Byzantine algorithm based on voting mechanism. In the consensus process, the voting scheme selects some honest nodes to involve the process of consensus and reduces the quantity of consensus nodes. It reduces the bandwidth occupancy rate and the switching frequency of views during the consensus process, thereby improving the consensus efficiency and throughput of the entire blockchain system.

According to the network scope, blockchains are generally divided into three categories[11], the first is a permissionless public chain, the second is a permissioned private chain, and the third is a consortium chain. The public chain system is the most open and can be used by anyone without any identity authentication. It is completely decentralized and not controlled by any institution. The data and transaction information in the block are completely open and transparent. Bitcoin is a typical public chain project. The private chain system is the most closed, that is, the private blockchain of an individual or an organization that is not open to the public. The read permission of the outside world is controlled by the owner, which is limited to the internal use of enterprises, schools or national institutions. Compared with the public chain, private The security of the chain is higher, and the number and status of nodes are controllable. MultiChain [12] is a typical private chain project. The alliance chain [13] system is semi-open and requires a certain permission to access the blockchain. Some organizations work together to maintain and manage the blockchain. The alliance chain is like a chamber of commerce alliance. Only members within the alliance can share benefits and resources. Compared with the public chain and private chain, the alliance chain has higher commercial value. Typical alliance chain projects include R3 alliance and original chain.

The proof-of-work algorithm (PoW) is widely used in blockchain at this stage. The PoW mechanism uses the speed of nodes to calculate Hash to perform currency distribution and determination of bookkeeping rights.Nodes in the network can reach consensus without exchanging additional information. The cost of malicious nodes launching 51% attack [14] is very high, but PoW's over-reliance on computing resources causes energy and hardware resources to suffer. It is wasteful. Dynamically adjusting the difficulty of producing blocks in Bitcoin can stabilize the time of producing blocks at about 10 minutes, and the throughput is difficult to meet the needs of transactions.

In response to the waste of resources caused by the PoW consensus algorithm, Sunny Kin et al. proposed PoS, and proposed the concept of "coin-age". The product of holding times. In the PoS consensus algorithm, the node's acquisition of accounting rights depends on the node's equity weight. The nodes already compete through computing power. When the node's coin age is large, the difficulty of calculation will be reduced. By introducing the coin age into the calculation of the block difficulty value, the nodes with a larger coin age have a faster block generation speed.

Equity authorization proof mechanism DPoS is to solve the risk problem caused by PoS algorithm that may hold a large number of shares. In DPoS algorithm, nodes use the shares they hold to vote for witness nodes, and these witness nodes will generate Block and validate blocks, which reduces confirmation time and increases the speed at which transactions are processed. The DPoS algorithm selects witnesses in a form similar to board election, and other witnesses will verify the blocks generated by the witnesses, so it is beneficial for ordinary equity nodes to not have to consume additional resources to verify each transaction. Stakeholders can vote for a node they trust, and if the stakeholder suspects the node, they can also withdraw their vote. If the witness does not generate the block on time, it may be eliminated in the subsequent voting.

The Paxos algorithm is based on message passing. It is mainly used to ensure the consistency of messages between processes or nodes, considering issues such as node downtime and fault tolerance. There are three main roles in the Paxos algorithm. The first is the proposer, this role proposes a value, for voting, the proposer receives the client request. The second is the receiver, which votes for each proposed value and receives the stored value, such as three nodes A, B, and C. Generally, all nodes in the cluster play the role of the receiver, participate in consensus negotiation, and receive storage. data. Another is the learner. The learner is informed of the voting result, receives the consensus value, stores and saves it, and does not participate in the voting process. Generally speaking, learners are data backup nodes, passively accepting data. The Paxos consensus algorithm has two consensus stages. The first is the preparation request stage. The proposer selects a proposal numbered m and sends a preparation message numbered m to more than 1/2 of the recipients in the entire network. When the receiver receives the prepare message, if the proposal number value is greater than all the prepared messages that have been replied, the receiver will respond correctly to this message. The second phase is the acceptance phase, if the proposer receives the prepare request response sent by the majority of the acceptors, it sends an accept request to each acceptor, proposes a proposal and then the other acceptors verify it and it is accepted.

Raft consensus protocol also has three characteristics: strong leader, leader election, and member change. In Raft, there are three types of nodes, namely, the master node (Leader), the candidate node (Candidate), and the slave node (Follower). In each term (Term), the master node maintains its leadership through the heartbeat mechanism (Heartbead). If the slave node does not receive a heartbeat message within a period, the system will start to elect a new master node. All nodes in the system can participate in the election to become candidate nodes. At this time, other nodes vote for the candidate nodes. The node that gets the majority of votes and meets certain requirements will become the master node. The master node will accept the operation request sent by the client and generate The log data (in the blockchain, the block is packaged and generated) is multicast to the slave nodes. After the slave node receives the log data sent by the master node, it will be synchronized and completely follow the pace of the master node.

In the process of master node election, the slave node will enter the next term and become a candidate node state, and then send a request for voting to other nodes. At this time, there will be three situations, namely, becoming the master node by itself, another node One node becomes the master node, and no node becomes the master node. For the first case, the first-come-first-served principle is followed in the election process of a term, and the node will vote for the first node he receives to request a voting request, so , in most cases in each term, a node will win the majority of the votes in the cluster to become the master node, thereby promoting the consensus to continue. For the second case, when a node becomes a candidate node in a term, it receives the heartbeat information of the master node in a higher term before it has obtained the majority of votes, then it will automatically adjust itself to the slave node. Node status, if the term in the received heartbeat information is less than its own term, it will continue to maintain the candidate node status and continue to collect voting information sent by other nodes. For the third case, before the start of a term, multiple nodes may become candidate nodes. At this time, the votes may be divided, and no candidate node obtains more than half of the votes. At this time, each candidate node will time out. , and enter the next term to restart the election, the candidate node sends a request for voting, and continues the competition of the master node.

In 1999, Migule Castro and others first proposed a practical Byzantine consensus algorithm, which greatly reduced the time complexity of the algorithm for solving the Byzantine generals problem. And it was recognized as the optimal solution to the Byzantine generals problem. To avoid the waste of computing resources,the PBFT consensus algorithm has the following three sub-protocols: checkpoint protocol, consensus protocol and view conversion protocol. During the normal operation of the improved system, the system nodes are mainly divided into master nodes and replica nodes. The master node is mainly used for Receive the transaction sent by the client and package the block, and then the master node initiates a three-stage consensus process to achieve consensus among the consensus nodes. The PBFT algorithm can reach a consensus in the case of n>3f+1. If the master node is a malicious node during the consensus process or the processing transaction times out, and then replace the master node to ensure the system operate normally. The core of the PBFT consensus algorithm is three Phase consensus protocol, the specific process is shown in Figure 1.

1) Pre-Perpare stage: After master node p receives the request message <*Request, o, t, c*> from the client c, after verifying the validity of the request, it encapsulates the Pre-Prepare message and broadcasts the message to the rest of the replicas Node, the message sent is <<*Pre-Prepare,v,n,d*>,*m*>, where v is the view number, n is the sequence number, m is the message, and d is the digest of m.

2) Prepare phase: After replica nodes 1, 2, and 3 receive the Pre-Prepare message sent from the master node, they will first verify the correctness of the master node's signature and message digest. If the verification passes, the replica node will construct a Prepare message <*Prepare,v,n,d,i*>, where i is the node number. In the Prepare phase of the consensus, the master node and the replica node will collect the Prepare message from the replica node, and verify Accuracy of Prepare messages. When the node receives more than 2f verified Prepare messages, it means that the Prepare phase has been completed, and the node enters the next consensus phase.

3) Commit stage: All nodes in the system will generate a Commit message <*Commit, v, n, d, i*> and broadcast the message. If more than 2f verified Commits are received at a certain point, it means that the entire system is in progress. Most nodes of the system have entered the Commit stage, and the consensus has been reached. The nodes generate a Reply message <*Reply,v,t,v,i,r*> where t is the timestamp and r is the request result, and then sends the Reply message to the client end c.

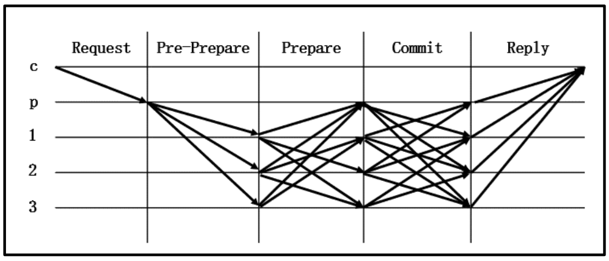

Fig 1. Scheme process

II. Improvements in this article

The PBFT algorithm is more applied to the alliance chain. The advantages are high reliability, low latency and high throughput, but there are also some disadvantages. One of the disadvantages is that the consensus protocol of the PBFT algorithm is cumbersome in the working process and occupies a large amount of bandwidth, the time complexity of communication reaches O(n2). Therefore, in this paper there is a improved PBFT scheme. Firstly, the quantity of nodes involving the consensus process in the system is determined by the total consensus nodes in the system. Afterwards, all nodes vote to elect the consensus node set, and. Because the consensus process is carried out in the consensus node set voted by the nodes, the probability of a node in set is a malicious node is much smaller. During the consensus process, the master node sends a Commit message to other nodes and accounting nodes in the consensus node set, and the consensus node sends a Reply message to the client after receiving the Commit message, which effectively reduces the overhead of networks.

A. The overall process of the program

The plan is to modify the original PBFT algorithm, and filter out a relatively honest set of consensus nodes before the consensus protocol. The remaining nodes are used as accounting nodes, and the master node is responsible for processing and sending Commit messages to nodes in the system. Before the consensus protocol is executed, all the nodes in the consensus network vote to select the relatively honest nodes in the system as consensus nodes. The number of consensus nodes is at least 4, and the remaining nodes are accounting nodes. The overall process is shown in Figure 2:

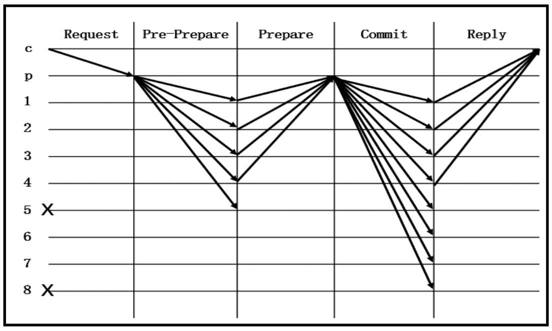

Fig 2. Scheme process

In Figure 2, nodes 1-5 are a set of consensus nodes, and node 5 cannot participate in the consensus protocol if it fails during operation. Nodes 6-8 are accounting nodes, wherein node 8 is a Byzantine node.

1) First, before executing the consensus protocol, all nodes in the network vote to divide the nodes into consensus nodes and accounting nodes, and dynamically adjust the number of consensus nodes in the system according to the total number of nodes in the system and the state of system operation.

2) In the consensus node set, the node with the most votes will be the master node p and responsible for receiving transaction requests from clients, starting the current round and generating new blocks.

3) After the node p receives the transaction request from the client c, it checks the transaction request, signs the generated Pre-Prepare message and broadcasts it, after passing the inspection, and saves the transaction in the log file.

4) Consensus node in the set check the signature of the message,when receiving message from the node p.After the check is passed, the honest nodes in the set will approve the transaction. After that, the Prepare message is generated and signed, and then sent to the master node, and the message record is saved in the log file.

5) The master node continues to receive Prepare messages from the consensus nodes and check the messages. After receiving at least 2f Prepare messages that pass the check, it means that the transaction is approved, and a Commit message is generated and broadcast to the network. After verifying the Commit message, it enters the Commit state, updates the newly generated block to the local blockchain, and sends a Reply message to the client. The accounting point directly writes the new block to the local blockchain at the blockchain height.

6) The client continues to receive Reply messages from the consensus nodes. If at least 2f messages are received, the consensus is reached, and the current round of consensus ends.

After multiple rounds of consensus, there is still a certain probability in the consensus process that the master node failure, network reasons or some irresistible factors will cause the master node to request timeout. At this time, the master node is considered to be a malicious node, it will trigger an attempt to switch the protocol to solve the problem of the master node's malicious behavior. Due to the introduction of a voting mechanism, the view conversion protocol of the improved scheme increases the profile of more honest nodes becoming master nodes, reduces the probability of using the view conversion protocol and the communication overhead of running the view conversion protocol.

B. View Conversion Protocol

The nodes in the set need to execute the consensus protocol under the same view. The view conversion protocol is responsible for maintaining the normal operation of the system when the master node is down or request times out or no new blocks are generated within a limited time. In the traditional PBFT protocol, the view conversion protocol goes through three steps. First, the replica node increases the view number by 1,at the same time broadcasts a view change message. And the replica node receives at least 2f view-change messages. After that, the new master node will receive a ack message. After all those changes the system goes on a new view.

III. Experiment analysis

In order to actually evaluate the performance of the improved scheme, this experiment uses the Go language to simulate the improved scheme and the traditional PBFT process, and then analyzes and discusses the experimental results.

A. Consensus latency and communication overhead

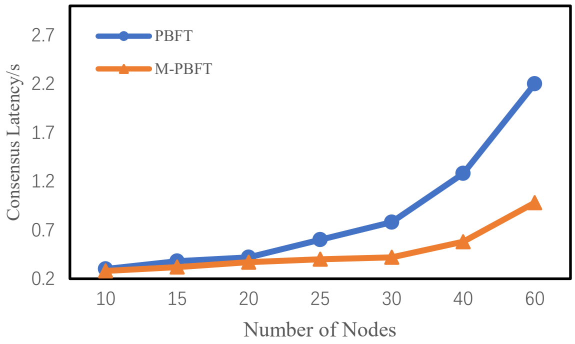

Fig. 3. Consensus Latency Experiment Results

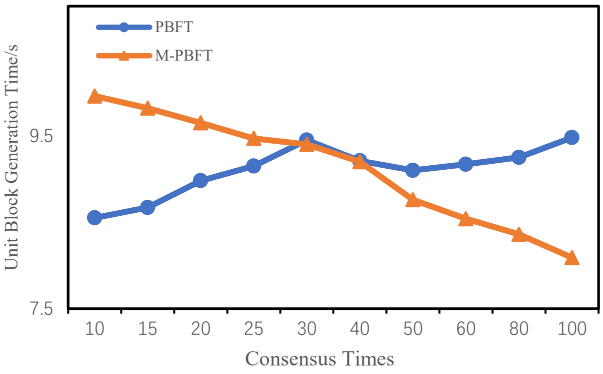

Fig. 4. Experiment Results of Unit Block Time

Consensus delay is a time period from the master node to package new transactions to the network confirmation block. This experiment uses the experimental environment written in Go language to package 1500 transactions per block, and tests are performed in simulation 10, 15, The average completion time of each block in the network of 20, 30, 40, and 60 nodes is compared and analyzed by taking the average value of multiple tests. The results are shown in Figure 3. It shows that with the continuous increase of the number of nodes in the network, the improved scheme in this paper proposed is obviously superior to the traditional PBFT algorithm in terms of consensus delay. The time required to complete consensus does not increase dramatically with the node quantity increase . The improvement scheme optimizes the consensus process in large-scale networks, reduces communications quantity and overhead, and greatly reduces the consensus delay. Therefore, the improvement scheme is more suitable for large-scale consortium chains. The communication times of the improved scheme proposed in this paper is n in the Commit stage, and the communication times of other stages are smaller than the traditional PBFT algorithm. Figure 4 shows the time experiment results of unit block generation. It can be seen that when there have same nodes the improvement scheme reduces the number of nodes participating in the consensus protocol when the number of nodes is too large and The number of communication stages in the consensus process is reduced, so that the unit block generation time is lower than that of the traditional PBFT algorithm.

B. Throughput Test

Throughput is the number of transactions processed by the blockchain system per unit time. Throughput is an important criterion for measuring the efficiency of the blockchain system. The calculation formula is:

$$TPS = Transactions / \Delta t$$

Where Δt is the block time of the system, and Transactions is the quantity of transactions processed by the network within Δt.

The number of nodes in the system has a greater impact on throughput. The experimental results of the throughput test are shown in Figure 5. It can be seen from the figure that with the continuous increase of the number of consensus nodes, the throughput of the two algorithms shows a downward trend. The throughput of the improved scheme proposed in this paper has obvious advantages. The improvement is that the proposed scheme reduces the communication overhead when the number of nodes is large. And the proposed scheme reduces the system overhead and quickly switches the master node when there is a problem with the master node, and reduces the system overhead, at the same time increases the throughput.

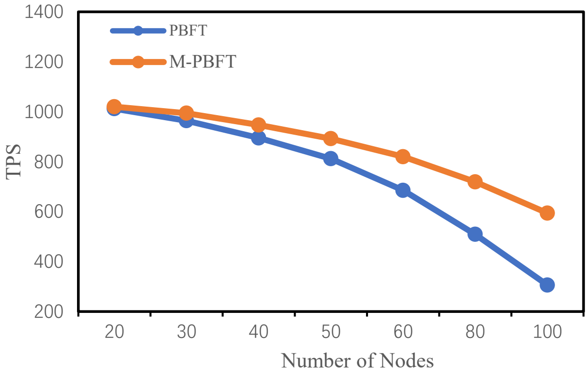

Fig. 5. Throughput Test

IV. Conclusion

This paper optimizes the PBFT algorithm and proposes an improved scheme based on voting mechanism. Make the relatively honest nodes in the system form the consensus part, and the remaining nodes become replica nodes. In the consensus process, the quantity of communication in several stages is reduced. When the master node have something wrong, the node can be quickly switched, which improves the system stability. While ensuring the fault tolerance of the algorithm, the overall communication overhead is reduced, and the consensus delay and throughput are significantly improved. Finally, the effectiveness of the improved scheme is verified by experiments.

Acknowledgment

This work was supported by Project 2022JJ50153 of the Hunan Provincial Natural Science Foundation of China, Project 18A230 of the Hunan Provincial Education Office Science Research of China.

References

[1] NAKAMOTO S.Bticoin: a peer-to-peer electronic cash system[EB/OL] 2008. http://bitcoin.org//Bitcoin.pdf

[2] Alaya Bechir. Efficient privacy-preservation scheme for securing urban P2P VANET networks[J]. Egyptian Informatics Journal, 2020(prepublish)..

[3] Zhang Liang, Liu Baixiang, Zhang Ruyi, Jiang Binxin, Liu Yijiang. Overview of Blockchain Technology [J]. Computer Engineering, 2019, 45(05): 1-12. DOI: 10.19678/j.issn.1000-3428.0053554.

[4] Tan Minsheng, Yang Jie, Ding Lin, Li Xingjian, Xia Shiying. Overview of Blockchain Consensus Mechanism [J]. Computer Engineering, 2020, 46(12): 1-11. DOI: 10.19678/j.issn.1000-3428.0059070.

[5] JAKOBSSON M,JUELS A. Proofs of work and bread pudding protocols(extended abstract)[M]//PRENEEL B.Secure Information Networks. Berlin, Germany:Springer, 1999:258-272.

[6] Seyed Mojtaba Hosseini Bamakan,Amirhossein Motavali,Alireza Babaei Bondarti. A survey of blockchain consensus algorithms performance evaluation criteria[J]. Expert Systems With Applications,2020,154(prepublish).

[7] Larimer D.Delegated proof-of-stake consensus [EB/OL]. [2020-11-10].https://bitshares.org/technology/delegated-proof-of-stake-consensus.

[8] CASTRO M, LISKOV B. Practical Byzantine fault tolerance[C]//Proceedings of the 3rd Symposium on Operating Systems Design and Implementation. [S. l.]USENIX Association, 1999:173-186.

[9] Junxing Wang. A simple Byzantine Generals protocol[J]. Journal of Combinatorial Optimization,2014,27(3).

[10] Fan Jie,Yi Letian,Shu Jiwu.A Review of Byzantine System Technology Research[J].Journal of Software,2013,24(06):1346-1360

[11] Liu Fengming, Chen Yuetong.Review of Blockchain Technology Research[J].Journal of Shandong Normal University(Natural Science Edition), 2020, 35(03):299-311.

[12] Komal Kundan Sharma,Jyoti Raghatwan,Mrunalinee Patole,Vina M. Lomte. “Voting System using Multichain Blockchain and Fingerprint Verification”[J]. International Journal of Innovative Technology and Exploring Engineering (IJITEE),2019,9(1).

[13] Gao Wuqi,Mu Wubin,Huang Shanshan,Wang Man,Li Xiaoyan. Improved Byzantine Fault-Tolerant Algorithm Based on Alliance Chain[J]. Wireless Communications and Mobile Computing, 2021.

[14] Wei Songjie, Lv Weilong, Li Shasha. A review of typical security issues in public blockchain applications [J]. Journal of Software, 2022, 33(01): 324-355. DOI: 10.13328/j.cnki.jos.006280.