Improved Raft Algorithm for Optimizing Authorized Nodes Based on Random Forest

Hao Tang
School of Computer and Information Engineering
Jiangxi Agricultural University
Nanchang, China

Wenlong Yi
School of Software
Jiangxi Agricultural University
Nanchang, China
Corresponding author:
yiwenlong@mail.ru

Yingding Zhao
School of Software
Jiangxi Agricultural University
Nanchang, China
Corresponding author:
zhaoyingding@163.com

Jinlong Huang
Jiangxi General Technology Engineering School
Jiujiang, China

Huanliang Xiong
School of Software
Jiangxi Agricultural University
Nanchang, China

Qiude Li
School of Software
Jiangxi Agricultural University
Nanchang, China

***Abstract*— The traditional Raft algorithm has such an issue as "vote snatching" among candidates that results in abnormal elections in the absence of a majority. To address this issue, a random forest-based method is introduced for the identification of authorized nodes. First, the attributes of authorization are introduced to the original Raft algorithm to be the label of authorized nodes, i.e., only authorized nodes are eligible for candidacy. Second, the random forest algorithm is introduced to classify the authorized nodes according to their term, broadcasting duration, timeout duration, and other characteristics to divide all nodes into preferred nodes and ordinary nodes. In the end, the main consensus group makes up of preferred nodes adopted integral accumulation to top out the preferred node with the highest integral as the authorized nodes. According to the results, the optimized Raft algorithm avoids abnormal elections and outperforms the original algorithm in election efficiency. The method offers an orderly mechanism of node operation for the Consortium Blockchain.**

Keywords—raft algorithm, blockchain, random forest, authorized node

I. INTRODUCTION

As a novel distributed ledger technology (DLT) integrated with cryptology, consensus algorithm, and P2P network, blockchain has been extensively applied in all sorts of traceability scenarios with such features as decentralization, traceability and tamper-proofing [1]–[3]. Blockchain can be divided into the following categories by its openness: public blockchain, consortium blockchain and private blockchain. Among them, the public blockchain is the least secure, while the private blockchain is the least open. Consortium blockchain performance is neither the best nor the worst in terms of its performance, which executes admission strategy towards its organization members while guaranteeing security. Now it has become the main subject in implementing blockchain research findings. As one of the core components of blockchain [4]–[5], consensus algorithm remains a hot topic in current consortium blockchain-related studies. Different from public blockchain that features complicated circumstances, consortium blockchain has known and relatively reliable nodes. Due to this reason, a non-byzantine fault tolerance algorithm is supported in consortium blockchain. Raft algorithm, a type of distributed consensus algorithm used in copy log management, is composed of three modules (leader election, log copy and security) and can be applied to solving consensus issues in a non-byzantine tolerance context. Being simple in structure, highly comprehensible and realizable, the Raft algorithm is chosen by consortium blockchain as its consensus algorithm [6]–[7]. To extend the Raft algorithm's applicability and adapt it to different application scenarios, many scholars have made modifications to the algorithm's existing basis. For instance, Fu et al. [8] invited equivalent nodes to take part in log information distribution at the log copy stage in order to lower the communication complexity of leader nodes. Xu et al. [9] analyzed the security performance of the algorithm in transaction success rate by mapping the blockchain transaction and constructing a wireless network.

The aforesaid works conduct modification from the log copy module or security, but this paper focuses on improving the leader election module and proposes a random forest-based priority Authorization Node Raft algorithm (AN-raft) with the view to solving problems concerning election failure and low consensus efficiency. Contributions of this paper are as follows:

- Authorization property is added into the original Raft algorithm to specify only authorized nodes are qualified for being candidates so that the Leader could succeed in the election and election failure could be well avoided.
- Random forest algorithm is employed to divide a cluster of nodes into preferred nodes and ordinary nodes and group the preferred nodes to form a consensus group and lower communication frequency.
- Credit-based authorization mechanism is introduced to establish preferred nodes with the highest credit as authorized nodes so that there is only one authorized node in the consensus group.

II. MATERIALS AND METHODS

A. Selection of preferred nodes

Random forest is to set up several decision-tree classifiers with various data subsets. It is a new method to randomly sample data and features, which could better improve the model's precision and stability than a single decision tree [10]. The random forest model makes independent trees with internet clusters on the server. The decision tree in the model could judge or predict the samples first and then selects the optimal classification as per the voting result. The schematic diagram of this process is shown in Fig. 1.

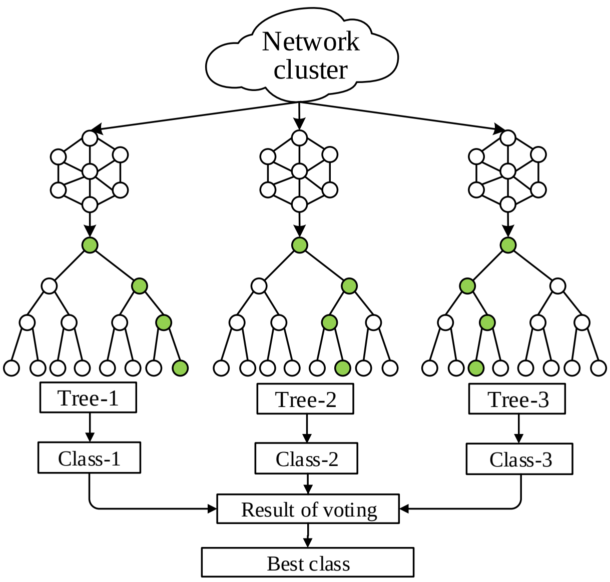

Fig. 1. Network cluster-based random forest structure

As shown in Fig. 2, network cluster nodes collected undergo a linear regression algorithm for data pretreatment in order to rule out abnormal and unrelated data. Subsequently, those network cluster nodes are grouped (n) to form sample datasets. Each training set is to generate a decision tree in accordance with the corresponding algorithm.

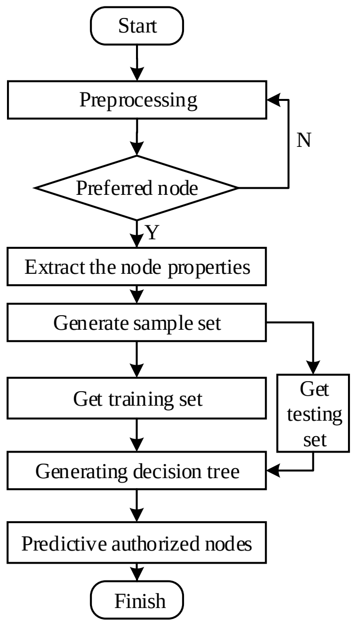

Fig. 2. Random forest-based authorized node judging process

B. Random forest-based Raft algorithm applied to authorization of preferred nodes

According to Fig. 3, for the traditional Raft algorithm, every node in the cluster is qualified to be converted from Follower to Candidate and ask voters for support at the start-up stage or during the downtime of the Leader node. Node that gets supported by over half of the voters will be elected as the new Leader. Should more than one node become Candidates within one cycle, votes may be scrambled for among those nodes so that none of them succeeds in obtaining over half of the votes. Such a situation may cause election failure and a new round of election would become necessary.

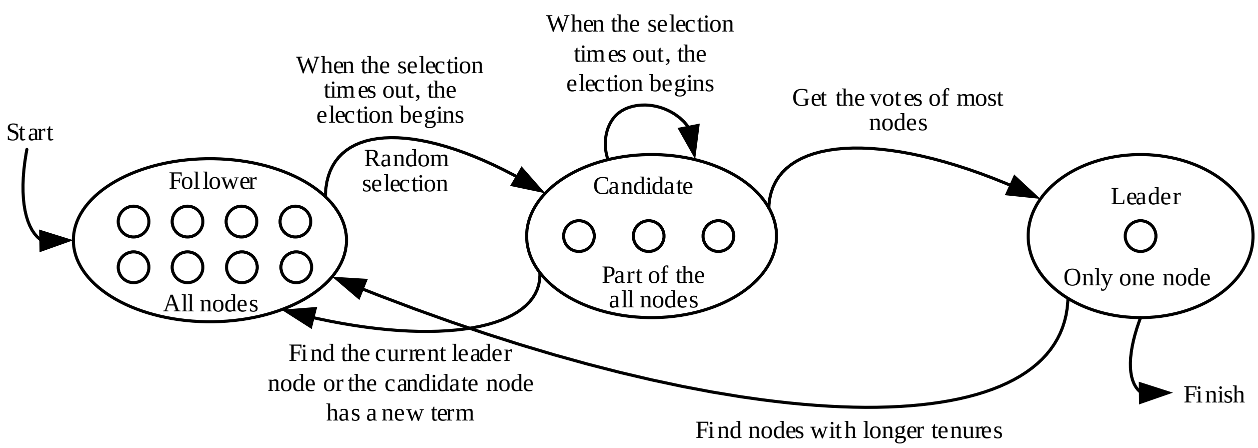

Fig. 3. Election based on traditional Raft algorithm

The random forest-based Raft algorithm for authorizing preferred nodes is illustrated in Fig. 4. Its authorized node setting prevents several nodes from being selected as candidates at the same time so that the Leader could be elected in each round. A decision tree algorithm is first applied to assess and classify network cluster nodes available for the election and then such nodes are tagged and classified as per such features as term, broadcasting duration and

This work was financially supported by the National Key Research and Development Program of China (Grant No. 2020YFD1100600) and the Natural Science Foundation of Jiangxi Province (Grant No. 20212BAB202015).

timeout duration. Nodes are classified into preferred and ordinary ones. The former ones selected based on the random forest would have their credits accumulated according to the credit authorization mechanism. The node with the highest accumulated credit would be authorized to become a Candidate.

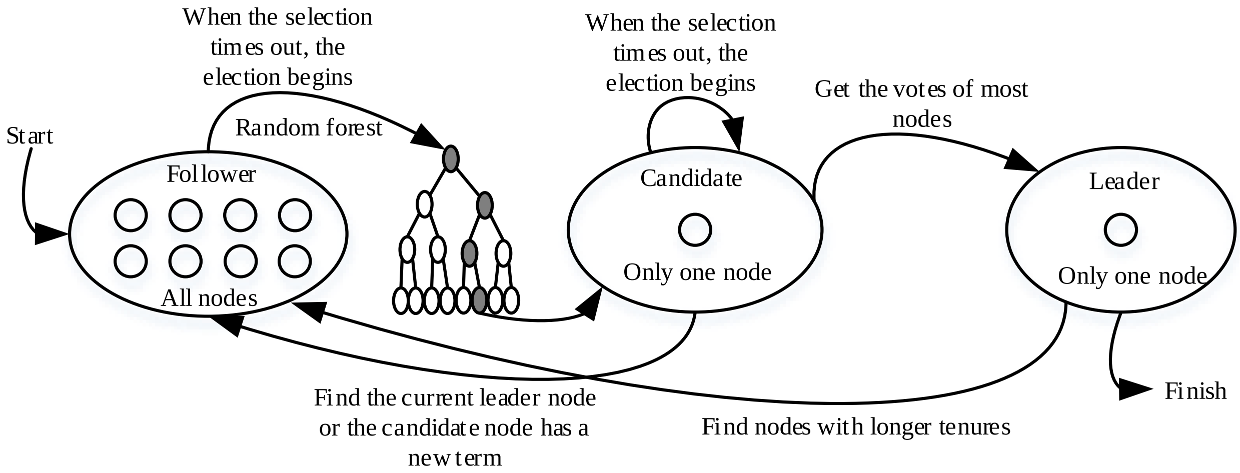

Fig. 4. Random forest-based Raft algorithm election for node prioritization and authorization

On the other hand, as the number of nodes increases, preferred nodes may be chosen to form a consensus group, and each preferred node may be grouped together with a proper number of ordinary nodes to form a consensus subgroup. A Leader node should be elected in the consensus group. As shown in Fig. 5, the Leader node would send a heartbeat message to other preferred nodes in order to prove it is working and prevent a new Leader from being elected. Upon receiving a heartbeat message from the Leader node, other preferred nodes in the consensus group would broadcast the message in their subgroups to inform ordinary nodes in those subgroups about the election of the Leader node and keeping the follower state. Later, the Leader node will send additional log entries through RPC to all the Follower nodes until all followers keep those entries. Group-based election and consensus could reduce communication frequency, improve performance against multi-node synchronization, and elevate consensus efficiency.

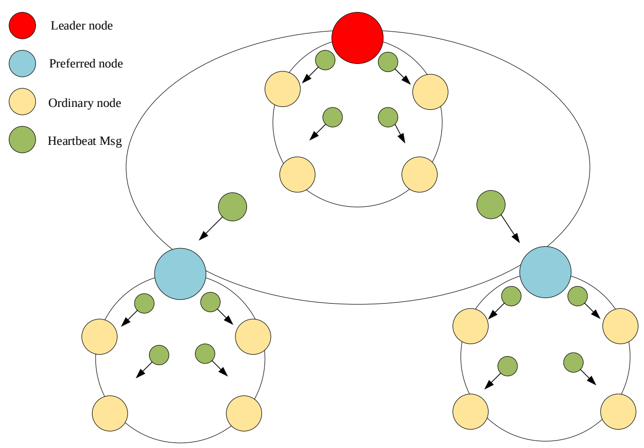

Fig. 5. Grouping consensus model

C. *Credit authorization mechanism for preferred nodes*

In actual application, as the number of cluster nodes grows rapidly, preferred nodes are not entirely the same in qualification so that more than one node may be authorized at the same time in one network cluster. This may lead to an abnormal election. When there are several authorized nodes within the consensus group, a credit authorization mechanism may be implemented to screen the network nodes and cut down the probability of an abnormal election.

Once the system initialization is done, the network nodes labeled as preferred ones would be offered 50 initial credits. After that, whenever such nodes are elected as Leader and send additional entries to other servers, other nodes would offer feedback. If entries are received by over half of the ordinary nodes, the Leader would execute the entries in the state machine and return the execution result to the client. As long as the whole log is successfully executed, the system would reward the Leader node with 10 credits and other followers with 5 credits.

III. EXPERIMENT AND DISCUSSION

A. *Experimental environment*

In the experiment, the AliCloud server is employed as the participant's online node in detection. Its software and hardware details are as follows: system image: Ubuntu 16.04 64; supporting environment go1.15.3 linux/amd64; supporting software: Docker version 18.09.7; hardware environment: 1-core CPU, 2GB memory, 40GB hard disk drive, 1M bandwidth. Several terminals are enabled to simulate the distributed nodes in the consortium blockchain and manual node connection/disconnection is used to simulate downtime and reconnection of nodes.

B. *Experimental analysis*

In the first place, Raft and AN-Raft are repetitively tested through multi-node simulation to figure out the variance between average election duration and multiple election durations. As shown by experimental findings in Fig. 6 and 7, the Raft algorithm generates significant variance between average election duration and multiple election durations and the resulting data fluctuates evidently. By contrast, AN-Raft has a short and stable average election duration and only one election is needed in spite of growing nodes. Thus, the AN-Raft algorithm shows a better and more stable performance when applied to consortium blockchain. Furthermore, its strengths become even more evident with the growing number of nodes.

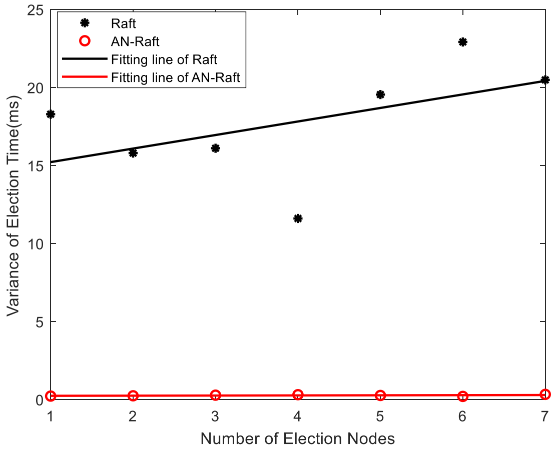

Fig. 6. Comparison of Raft with AN-Raft in average election duration

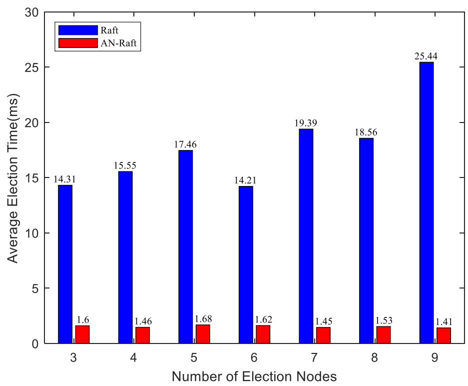

Fig. 7. Comparison of Raft with AN-Raft in election duration variance

Then, the experiment simulates the existence of several authorized nodes at one time. More authorized nodes are added to test on AN-Raft repetitively to obtain the average election duration. As shown by the experimental findings in Fig. 8 and 9, the algorithm performs well when coping with the election of a single authorized node but displays vertical fluctuation in performance and appears particularly unstable when applied to the election of multiple authorized nodes. As the number of authorized nodes grows, the standard deviation increases accordingly, presenting a dramatic rising tendency. It could be concluded that in view of the strengths of the credit-based authorization mechanism, only one authorized node should be allowed in each consensus group to obtain the best performance and stability.

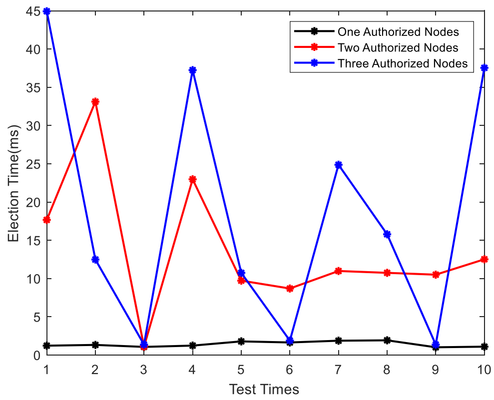

Fig. 8. Comparison of AN-Raft's election stability in 1-3 authorized nodes

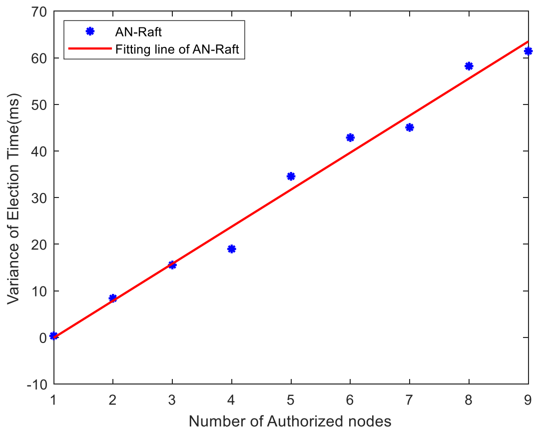

Fig. 9. Comparison of AN-Raft's average election time in multiple authorized nodes

IV. Conclusion

A random forest-based Raft algorithm for authorizing preferred nodes could avoid an abnormal election. Its performance is proven to be better than the original Raft algorithm in terms of election efficiency and stability. Furthermore, it groups the nodes to form consensus, lower communication frequency and improve multi-node synchronization performance. The proposed method may offer a well-organized node operating mechanism for consortium blockchain.

References

[1] W. Yi, X. Huang, H. Yin, and S. Dai, "Blockchain-based approach to achieve credible traceability of agricultural product transactions," Journal of Physics: Conference Series, vol. 1864, no. 1, p. 012115, May 2021.

[2] W. Yi, Q. Li, H. Yin, H. Tang, and Y. Zhao, "Efficient User Inspection Algorithm Based on Dual Bloom Filters Oriented for Blockchain Data Management Systems," 2021 XXIV International Conference on Soft Computing and Measurements (SCM), May 2021.

[3] Q. Li, W. Yi, X. Zhao, Y. Zhao, H. Yin, and Y. Xu, "Design and Evaluation of a High-performance Support System for Credibility Tracing of Agricultural Products," 2021 IV International Conference on Control in Technical Systems (CTS), Sep. 2021.

[4] B. Lashkari and P. Musilek, "A Comprehensive Review of Blockchain Consensus Mechanisms," IEEE Access, vol. 9, pp. 43620–43652, 2021.

[5] X. Zheng and W. Feng, "Research on Practical Byzantine Fault Tolerant Consensus Algorithm Based on Blockchain," Journal of Physics: Conference Series, vol. 1802, no. 3, p. 032022, Mar. 2021.

[6] J. Hu and K. Liu, “Raft consensus mechanism and the applications,” Journal of Physics: Conference Series, vol. 1544, no. 1, p. 012079, May 2020.

[7] A. Alexandridis, G. Al-Sumaidaee, R. Alkhudary, and Z. Zilic, “Making Case for Using RAFT in Healthcare Through Hyperledger Fabric,” 2021 IEEE International Conference on Big Data (Big Data), Dec. 2021.

[8] W. Fu, X. Wei, and S. Tong, “An Improved Blockchain Consensus Algorithm Based on Raft,” Arabian Journal for Science and Engineering, vol. 46, no. 9, pp. 8137–8149, Feb. 2021.

[9] H. Xu, L. Zhang, Y. Liu, and B. Cao, “RAFT Based Wireless Blockchain Networks in the Presence of Malicious Jamming,” IEEE Wireless Communications Letters, vol. 9, no. 6, pp. 817–821, Jun. 2020.

[10] D. Song, Y. Wang, and M. Yuan, “An Improved Method of Blockchain Consortium Chain Consensus Mechanism Based on Random Forest Model,” Communications in Computer and Information Science, pp. 148–157, 2021.