Multilevel Voting Based Consensus Algorithm for Private Blockchain

Dr. Kiran Manjappa
Information Technology
National Institute of Technology,
Karnataka, Surathkal, India 570025
kiranmanjappa@nitk.edu.in

Ayush Hemnani - 222IT012
Information Technology
National Institute of Technology,
Karnataka, Surathkal, India 570025
ayushhemnani13@gmail.com

Shubhranshu Dubey - 222IT031
Information Technology
National Institute of Technology,
Karnataka, Surathkal, India 570025
shubhranshudubey14@gmail.com

***Abstract*—Security and performance are consistently at danger due to the fast adoption of digital technology. Internet performance constraints and corrupted data are both possible. Blockchain technology is the answer to this issue. Since blockchain is a fairly new concept, it is currently one of the most competitive fields in the IT industry. Blockchain is structured on a peer-to-peer network topology with linked nodes. Each block includes a timestamp, necessary data, and an encryption hash of the following node to guarantee that no malicious or unauthorised nodes can access the linked nodes. Blockchain increases the resistance to unauthorised access to and change of the data, and since each node has its own processing power, it strengthens the security and effectiveness of the host network. The usage of private blockchain within companies or between organizations where private nodes are utilized and new nodes are required enabling authority to link nodes. This theory's objective is to reduce the number of messages exchanges between the nodes in a private environment to reach consensus phase. This theory's findings will lead to proposing a method which will have an algorithm that is based on voting for improved performance.**

Keywords: Blockchain, Voting based, Private Blockchain

I. Introduction

The computer networking sector is incorporating new technologies at the same time that the IT industry is growing. One of them being blockchain. One of the technologies that is being used more and more frequently is blockchain technology. The distributed peer-to-peer networking idea is the primary driver behind blockchain's increasing popularity. P2P empowers blockchain to spread work and resources over a network of computers known as nodes. By ensuring safety against DDoS and other attacks, this allows the use of blockchain in a diverse range of industries, particularly finance and healthcare, where the privacy of patient information is crucial.

As the name suggests, a private blockchain is one in which nodes must receive approval in the sense of a certificate of authorization in order to enter the network, these nodes are also private and well-known. The nodes of the private blockchain are deployed inside the organisation and are used inside or between organization. Because it offers stronger protections and is built on a permissioned blockchain, Hyperledger, this blockchain seems to be more reliable and safe. The consensus algorithms serve as the foundation for blockchain technology. The consensus algorithm is a method where nodes or groups of nodes agree to carry out a specific activity. Blockchains with and without permission adopt multiple consensus algorithms according to the needs and resources available. Algorithms used in permission-less blockchain are analogous to proof-based algorithms (Proof of Work, Proof of Stake). The majority of voting based consensus algorithms and Byzantine fault tolerance algorithm, as well as its variants, are used by private blockchain, in contrast. Smart contracts, which seem to be scripts or pieces of software that reside in the network and give agreement, are another feature of private blockchain.

The most widespread consensus technique, proof-of-work (PoW), involves solving challenging problems in order to reach consensus. The nodes with a larger share in the network are awarded consensus in a proof of stake procedure. However, these algorithms consume a lot of processing power. The Byzantine fault Tolerance algorithm and its variants are used in private blockchain to achieve agreement between the nodes even when there are network flaws, such as when one node is unable to interact with the server but others can. There seem to be numerous PBFT algorithms, each with its own features. The primary node chosen by voting based techniques like Raft controls the consensus among the remaining nodes, in contradiction to how other nodes are picked by the algorithms. The implementation of private blockchain technology is challenging.

The solution that this theory provides is based on voting algorithm which subsequently reduces the number of messages that has to be passed in order to reach the consensus phase. The theory also gives a solution which observes an subsequent increase in the fault tolerance of the environment. The motivation of this theory comes from the idea in which one mode of blockchain can cover the limitation of other mode of blockchain and vice versa, building a system which will have the main features of private blockchain.

II. Literature Survey

In contrast to public blockchain, which is permission less, in private blockchain nodes in the networks are known, thus the nature of consensus is different (miners also know each other). Scalability, throughput speed, and consistency are just a few of the variables that impact the permissioned blockchain's consensus process . Smart contracts are also implemented in permissioned blockchain to help the nodes come to an agreement. Additionally, not every node participates in the consensus process.

A. Variants of Practical Byzantine Fault Tolerance

The private blockchain employs this technique [**?**] to reach consensus even in the presence of fraudulent nodes or when certain nodes are unable to interact with the server whereas other can. This is handled by using a tolerance threshold and the replication process. BFT includes three phases: pre-prepare, prepare, and commit. The request message, which includes a digital signature and the message body, is given a sequence number by the primary node and sent to the nodes taking part in the consensus. The message is received by the nodes, who then check it using the digital signature prior to actually transmitting it to the other nodes for subsequent verification.

Demerits:

- The scalability of a network supporting BFT is severely constrained because the communication is multi casted to the members in phases.
- Attacks using Sybil can damage the network.

B. Raft

A voting based algorithm known as Raft is implemented in private blockchain. Candidates with good relationships to other nodes are evaluated as candidates. Voting is applied to select one candidate as the leader node. The rest of the nodes and candidates become the main node's followers. The leader conveys to the following. Each term, also known as a round, a leader is chosen. A new validator is nominated when a term is up. A leader puts forth a particular value for the followers to consider; if they agree, consensus is reached.

Demerits:

- Slow progress if the favored candidate has poor capabilities.
- Is inconsistent with Byzantine nodes.

Observation of different parameters in Raft and pbft can be seen in the figure 1 given below.

PARAMETER	RAFT	PBFT
Scalability	Low	Low
Energy Required	High	Lower then Proof of Stake
Decentralization	Not all nodes store ledger	Not all nodes store ledger
Reward for transaction performed	No	No
Implemented in	Hyperledger	Hyperledger

Fig. 1: Observation of existing methods

III. Methodology

The theory proposed in this methodology is based on voting algorithm for a private blockchain. First it randomly create groups of approximately equal sizes and then broadcasts the transaction. Voting is carried out among the group members and validator from each group is selected. After that, second level voting takes place where voting takes places between the validators selected from each group. The one who is elected mines the block and adds it to the main blockchain. The detail working of the theory can be observed from the flowchart given below:

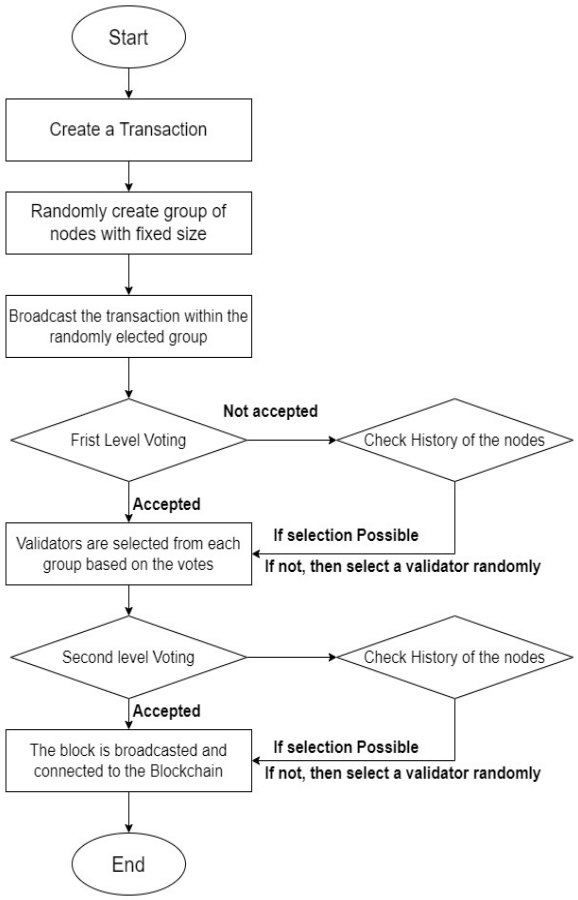

Fig. 2: Flowchart

Let us take an example to get a proper understanding, Suppose there are 25 nodes in a private blockchain environment. Since it is a voting based algorithm every node will send its vote to all the present nodes. For e.g. A votes for B, then A will broadcast this information to all other nodes. Hence, if n=25 there will be 25*25 = 625 messages passed in the network. And same 25*25 = 625 for acknowledgement that the information sent by the node A has been received by all other nodes and so on. Hence $n^2 + n^2 = 2n^2 = 1250$ messages will be passed in the network for 25 nodes.

It is important to note that, if suppose A votes for D and broadcasts this information to all other nodes. Every node in the network now knows that A has voted for D. Suppose if any node bluffs for e.g. if B bluffs that A has voted for it, the other nodes can easily make out the fraud and hence the node or the miner that catches the malicious node will be rewarded and the fradulent node will be penalized.

Now, according to theory proposed we will first group the available nodes randomly. Let us say that 5 groups each consisting of 5 nodes is created to keep a balance between number of groups and number of group members within them.

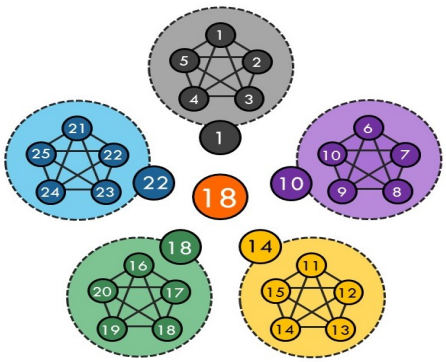

Fig. 3: Randomly generated group

In every randomly generated group first level voting takes place. 5 nodes will send their voting information to each and every node. Hence, 5*5 = 25 messages will be passed in the in first level voting in each group. Five such groups are generated here and hence 25*5 = 125 messages and same 125 for acknowledgement that the nodes information has been received by the other nodes. Hence 125+125 = 250 messages are passed in the network for first level voting.

Now from each group one validator is selected. In our case 5 validators has been selected. Now second level voting will take place, hence 5*5 = 25 and the same 25 for acknowledgement. Hence 50 messages will be passed in the network for second level voting. In total only 300 messages were generated instead of 1250 by using the proposed theory.

To generalize the example considered in the first level voting (5*5)*5 messages were generated which is $(\sqrt{n} * \sqrt{n}) * \sqrt{n} = n\sqrt{n}$ and the same for the acknowledgement. In total, $2n\sqrt{n}$ messages were generated in the first level voting. In the second level voting 5*5 i.e. $\sqrt{n} * \sqrt{n} = n$ messages were generated and the same for acknowledgement. In total for both the level of voting $2n(\sqrt{n} + 1)$ messages were generated. Comparing the traditional approach with the theory proposed we can say that: Hence, from the above derivation we can say that the

$$\frac{\boldsymbol{n}}{\sqrt{\boldsymbol{n}} + \mathbf{1}}$$

proposed theory reduces the number of messages passed in the network by O($\sqrt{n}$) times i.e. the proposed theory is $\sqrt{n}$ times better than the traditional approach.
There can arise many situations that the all the members of the group votes for exactly one each person distinctly. For e.g if 5 nodes are there in a group, it can so happen that everyone receives one vote. The probability of happening this is very less, but yet certain unexpected cases can occur. Hence, if the voting takes place in a circular permutation we can say that $(n-1)!$ cases will occur where everyone will receive one vote. And the total number of possible voting permutations will be $(n-1)^n$. Thus, we can say that the probability will be very less for this case to occur. Probability : Apart from that as we

$$\frac{(\boldsymbol{n} - \mathbf{1})!}{(\boldsymbol{n} - \mathbf{1})^{\boldsymbol{n}}}$$

keep on increasing the number of the nodes in a group or in a network the probability keeps on decreasing. This observation can be seen from the figure given below. We can observe that the graph that comes is logarithmically decreasing when the number of nodes increases from 1 to 50.

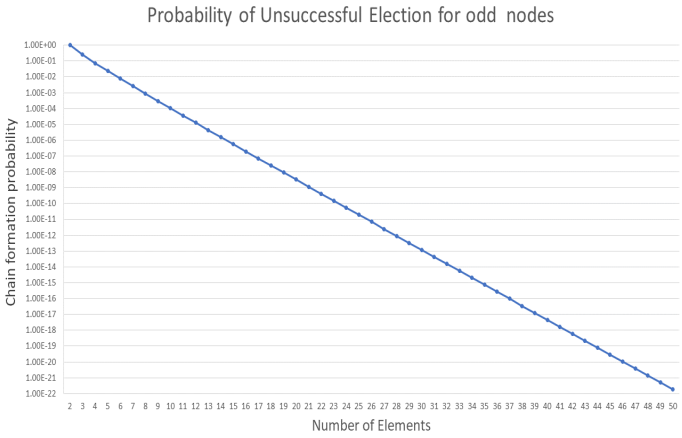

Fig. 4: Probability of unsuccessful election for odd nodes

Few of the values can be seen in the table given below that when the number of nodes increases the probability decreases subsequently.

Number of Nodes	Unsuccessful Probability in 2LV for odd nodes $\frac{(n-1)!}{(n-1)^n}$
2	1
3	0.25
4	0.0740741
5	0.0234375
6	0.00768

Fig. 5: Probability of unsuccessful election for odd nodes

This remains the same for even number of nodes. For even number of nodes the generalised formula depends upon number of non trivial factors of total number of nodes in the voting process.

$$\frac{\left((n-1)! + \sum \frac{(n-1)!}{fi!}\right)}{(n-1)^n}$$

Values for some of the nodes can be observed from the table given below:

Number of Nodes	Unsuccessful Probability in 2LV for even nodes $\left((n-1)! + \sum \frac{(n-1)!}{fi!}\right)/(n-1)^n$ (for every non trivial factor(fi) of n)
3	0.22223
4	0.01728
5	0.00117
6	0.00011
7	1.293e-05

Fig. 6: Probability of unsuccessful election for even nodes

Reduction in probability was seen when the number of nodes were increased subsequently which can be seen in the graph given below.

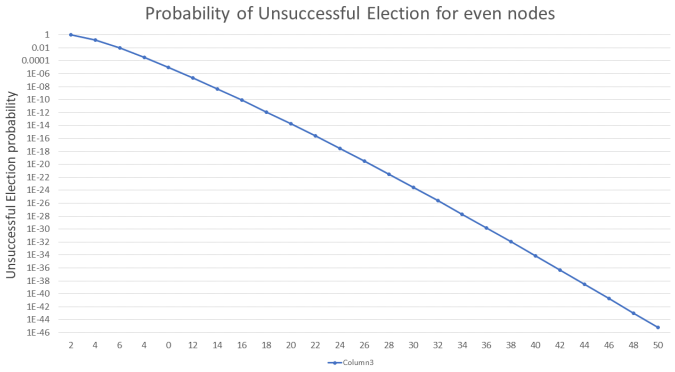

Fig. 7: Probability of unsuccessful election for even nodes

Until now all the observations were made for the first iteration where we elect a validator from each group. What if some nodes behave maliciously? What weightage should be given to them in the next iteration of voting? How to differ them from other nodes? To resolve such issues we maintain history of each and every variable of the node present in the private blockchain network. The history stores the ratio of how correctly the node is behaving to how many voting has been conducted. Initially the history of all the nodes will remain null. As the iterations keeps on occuring the ratio is calculated of how faithful the node is behaving. In the next iterations when the voting takes place, every node will contribute their vote proportional to their weight. And thus the voting iterations goes on. Methodology example can be seen in the figure 8 given below.

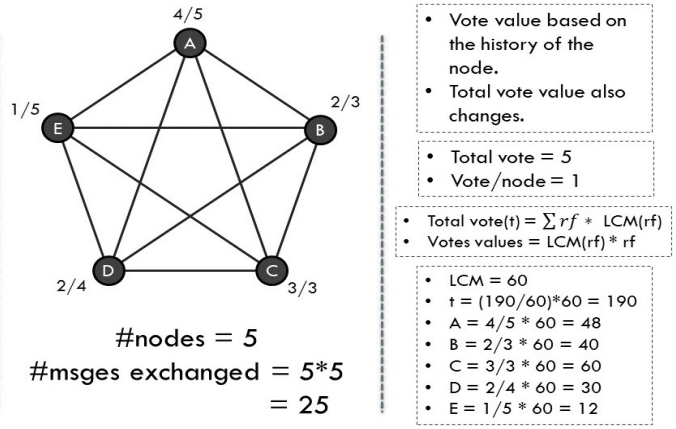

Fig. 8: Methodology example

IV. Expected Results

A. Reduction In Number of Messageges

As we observed above in the methodology, that the proposed theory shows a reduction in the number of messages passed in the network by $O(\sqrt{n})$. Comparison of the theory proposed with the existing solutions can be observed in the table given below.

Number of Nodes	Number of messages in RAFT (n^2)	Number of messages in MLV $(n\sqrt{n}+n)$	Reduction Factor $\frac{n}{\sqrt{n}+1}$
4	16	12	1.3334
9	81	36	2.25
16	256	80	3.2
25	625	150	4.1667
36	1296	252	5.1428

Fig. 9: Message Reduction factor

As we keep on increasing the number of nodes, we observe a subsequent increase in the reduction factor rate compared to the existing traditional solutions which can be observed from the table given below. Graphical comparison can be observed from the graph below.

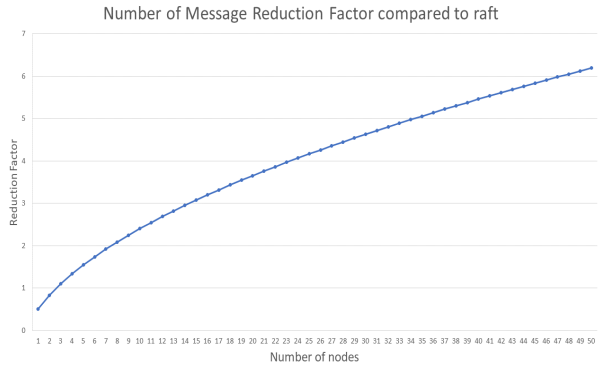

Fig. 10: Message Reduction factor w.r.t existing solutions

B. Fault Tolerance

Let us suppose that we have n nodes in a private blockchain network. A traditional method can handle $((n-1)/2)$ number of faulty nodes. Suppose for the same network our proposed theory after random shuffling of the nodes we got m = $\sqrt{n}$ groups with m members each.

In the theory proposed the number of malicious node that can be handled in a network increases subsequently. Suppose out of m, $((m-1)/2)$ elective nodes are faulty nodes and at worst case all the members of these faulty elected nodes are also faulty which makes a total of $m*((m-1)/2)$ group behave maliciously. Yet this case can be handled by the theory proposed. Moreover, even if in the remaining $(m-((m-1)/2))$ groups $((m-1)/2)$ nodes of each remaining group becomes faulty, the consensus phase will take care of such situation. Hence we can say that the theory proposed has more fault tolerance as compared to the existing techniques. The generalized form of how many malicious node the network

$$m\left\lfloor\frac{m-1}{2}\right\rfloor+(m-\left\lfloor\frac{m-1}{2}\right\rfloor)\left\lfloor\frac{m-1}{2}\right\rfloor$$

can handle can be given by :
The fault tolerance of PBFT is (n/3) whereas the fault tolerance of Raft is (n-1)/2. The compariosn of fault tolerance of Raft and PBFT with the 2 level Voting can be observed in the figure 13 given below:

Number of Nodes	Fault Tolerance		
	PBFT	RAFT	2LV
9	3	4	5
100	51	50	75
400	133	200	300
1600	533	800	1200
2500	833	1250	1875

Fig. 11: Fault Tolerance comparison

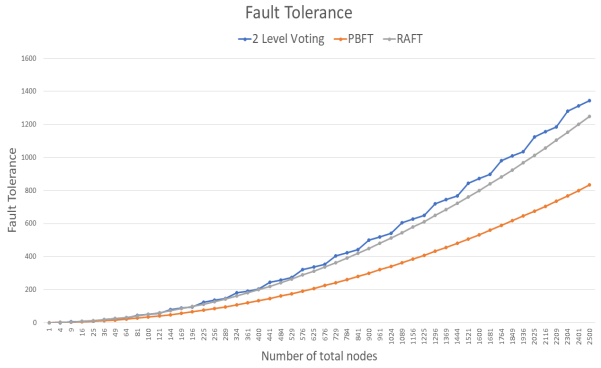

Fig. 12: Fault Tolerance comparison existing solutions

Fault tolerance factor of two level voting technique with respect to PBFT and Raft has been calculated in the table given below and we can observe that fault tolerance factor become constant as we increase the number of nodes in a private blockchain network.

Number of Nodes	Fault Tolerance Factor	
	2LV/PBFT	2LV/RAFT
9	1.667	1.25
100	2.2727	1.5
400	2.2556	1.5
1600	2.2514	1.5
2500	2.2509	1.5

Fig. 13: Fault Tolerance Factor

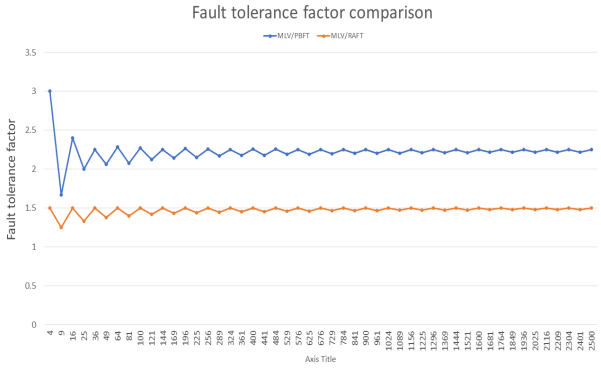

Fig. 14: Fault Tolearnce factor w.r.t existing solutions

V. Conclusion

One of the fundamental blockchain consensus protocols, which is employed in private blockchain, has been outlined in this suggested theory. In order to make it easier to select the consensus as per the business requirements that directly affect its performance, we have given a theory on voting-based consensus approaches utilised in private blockchains. The summary of this consensus process will be valuable for further investigation by academics. With the use of this consensus method for private blockchains, more research may be done in contrast with different parameters. In order to determine the true performance indicator of the consensus employed, alter the amount of loads and peers and analyse it using certain benchmarks.