A Consensus Protocol Based on the Ability of Network Dispersity

Yj1190590[*†]

Abstract. This article describes a blockchain consensus protocol that based on network terminals and transmission latency. According to this protocol, we could build blockchains with Pow-like consensus mechanism but without high energy consumptions. The situation of all the consensus protocols in this article is about fully decentralized systems, protocols such as “PBFT”, “DPOS”, “Ripple” are not involved.

KEY WORDS

1. Proof of Network Dispersity(PoND). 2. Vote. 3. Reward distribution. 4. Self-interest.

1. Introduction

The “Proof of Work” (PoW) protocol is being criticized because of the energy consumptions. But PoW is still the most essential and effective method of the blockchain system regardless of the energy issue. With the logic of “able people should get more pay”, PoW is the first consensus protocol that makes a total decentralized system work. All the other protocols are following the same logic, like Proof of Activity, Proof of Burn, Proof of Storage, Proof of Elapsed Time, and so on. “Proof of Stake” (PoS) protocol is the most successful one of them because it doesn’t need any external resources and consumes low energy by expanding the main logic from “able people should get more pay” to “rich people should get more pay”. Which makes the mining competition only among static inner states and there is no need for high power consumptions. But PoS has its own flaws such as “nothing at stake” which can cause double voting problem, “stake grinding attack” and so on. An inevitable problem of PoS is caused by the logic: “rich people should get more pay”. Whenever the miners spend the same amount of time, the richer ones earn more, so that the rich miners tend to spend more time to work than others. In that case, the rich miners will gradually become richer and richer and this process will be ever-accelerated until there are only some richest users left in the system. By the way, some PoS-like protocols offer “interests” to every stakeholder to replace the mining process. It seems that they resolved the problem above but those protocols are vulnerable for lack of incentive to maintain the network.

The purpose of this study is to find another low energy consumption proof of ability to replace PoS. Considering the following fact: the more dispersed online terminals work together, the faster they pick up randomly distributed information on the network because of the network latency. And this is a kind of “ability” that I called “Network Dispersity”. Which can’t be improved by enhancing the performance of single machines or using more electric power. The “stake” property plays an important role as a measurement in the protocol because of the feature of unduplicatable. Users can freely choose to participate in the maintenance of network with their abilities or with their stakes. The power of mining will be balanced by a dynamic-adjustment of the reward distribution to maximize the fairness and safety.

[†]Yj1190590 (3171228@qq.com)

2. Scenario and Characters

The whole network can be imagined as a kind of canvassing scenario. There are three types of characters according to the node's functions.

Voter—Each node that broadcasts a transaction could be a Voter which is mainly responsible for answering the "canvass" requests from the Workers and publishing transactions with its votes packed into them. The stake that a Voter holds is seen as "votes" in this scene and the number of "votes" will determine the mining competition. Any user could be a Voter.

Miner—Miners have their own Workers canvassing "votes" for them throughout the network and they use those "votes" to compete for generating blocks. Any user could be a Miner.

Worker—Workers work for the Miners to detect the Voters nearby and send a "canvass" request as soon as possible. Workers are mostly running on the network terminals by being embedded into client Apps or web sources.

The network scheme is shown as below:

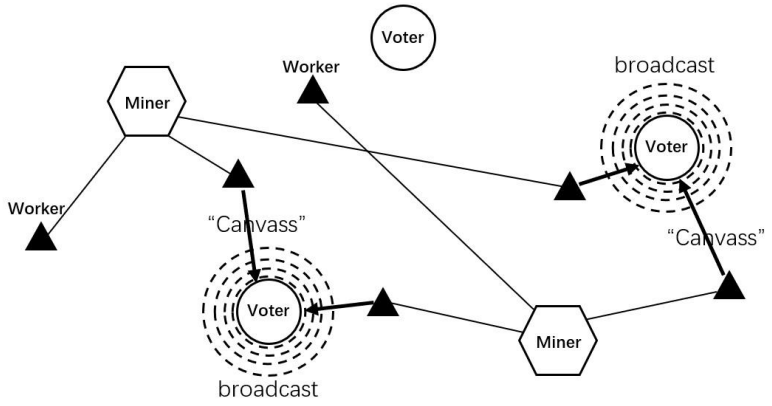

Fig. 1. nodes & network scheme illustration.

3. Consensus Process

(1) A Voter broadcasts a signal to preannounce it before publishing a transaction. The Workers nearby send "canvass" requests back when they detect the signal.

(2) The Voter packs the main chain's[1] tail-block "b" and the Worker's owner(Miner)'s account "m" into the transaction structure after receiving the first request from a Worker, and then broadcasts the transaction.

(3) This transaction is validated and packed into a new block by another miner, and then the block is published on the network.

(4) When a Miner receives a new block, he begins to validate it. And the Miner checks the field "b" and "m" of each transaction at the same time. Mark the transactions with "x" when their "m" fields point at the Miner himself and mark them with "y" when their "b" fields match the current chain's tail-block.

(5) Equally divide the stakes of each Voter among all of its transactions in that block.

(6) Add up the stakes divided at the previous step of all the transactions marked with both "x" and "y" block by block until the Miner meets the target of generating a block. Denote the result by "X" and the max number of accumulated blocks is the voting cycle (*e.g.* 6000 blocks).[2]

(7) Add up the divided stakes of the transactions marked with "y" in the current block and denote the result by "Y".

(8) Miners are trying periodically to meet the target of generating a block with a mathematical operation based on constants such as timestamps and private signatures.

That is denoted by: `hashProof()<target*d*X*Y` ("d" means the difficulty adjustment parameter).[3]

(9) The Miner packs all the transactions received during the period above into a new block and broadcasts it after meeting the target of generating a block. All of the referred transactions (coded within 6000 blocks to save some space), profits of all nodes and the other parameters should also be packed for verification. Mining rewards will be distributed between the Miner and all the participant Voters.[4]

For better understanding, the steps can be briefed as: "Every time they publish a transaction, stakeholders vote with their stakes on miners and on branches. The more stake a block or a miner get, the higher chance they win the competition." [5,6]

The following diagram shows the mining competition based on the network latency:

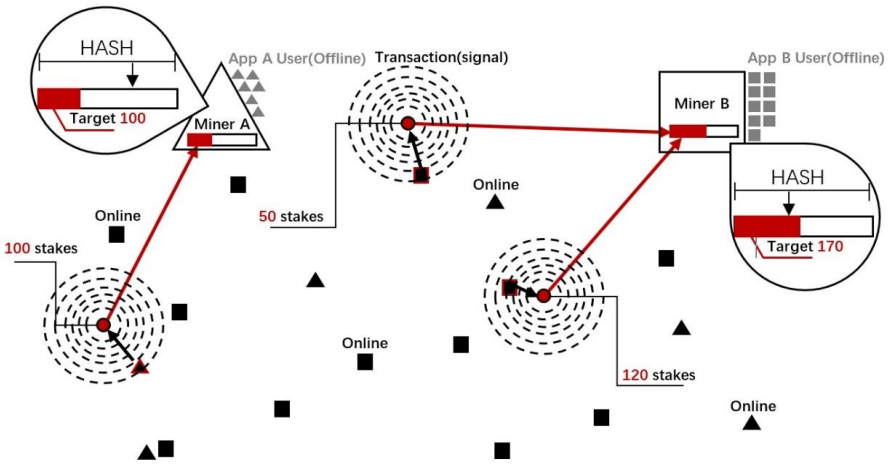

Fig. 2. Process of competition for the Voters

As shown in the scheme above, the more dispersed online user an App has, the higher possibility it has of winning the votes(stakes).[7] Which will help the miner win the mining competition later.

The votes will be recorded in blocks. The following diagrams show the process of competing for block generation (x-vote) and the main chain (y-vote) respectively:

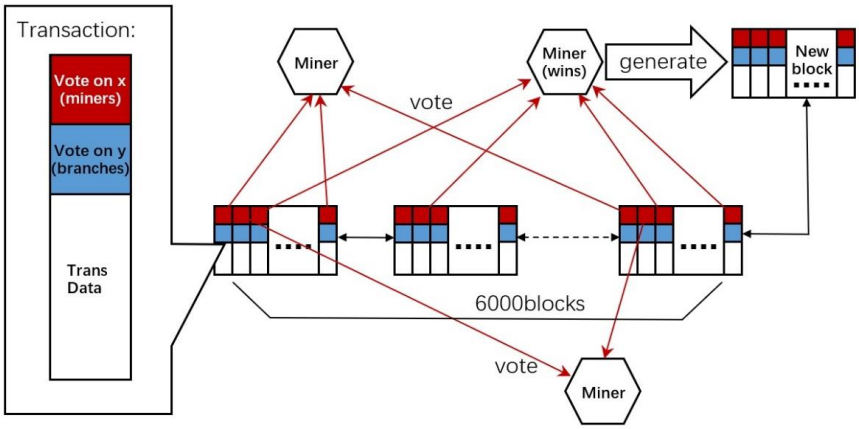

Fig. 3. The process of x-votes taking effect

As shown in the scheme above, the system will count votes in the existing blocks and the Miner with maximum votes has the highest chance to win the competition.

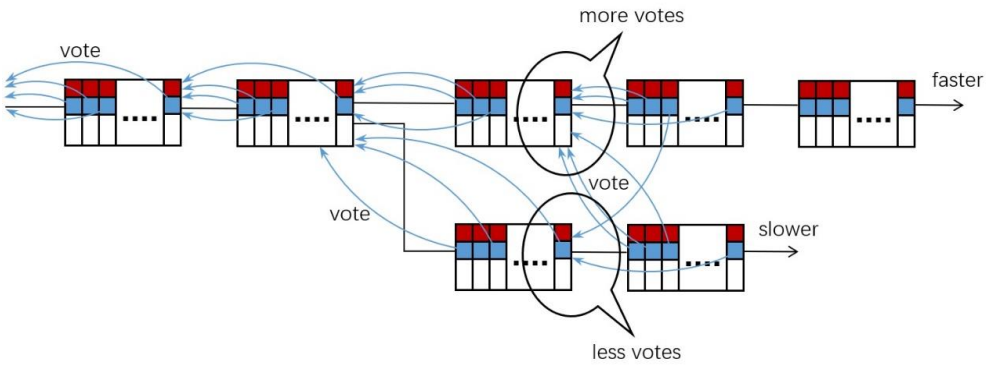

Fig. 4. The process of y-votes taking effect

As shown in the scheme above, when there comes a fork, Voters will vote between the branches and the votes will be recorded in the next block. Which determines the number of votes (stakes) for both branches in the next block generation (*i.e.* the amount of block generation difficulty parameter "Y" is determined). Thus, the more votes one branch can get, the faster it will be to generate the next block, and the sooner the next block is broadcasted, the more votes a branch will get from the next round. This process will increase the speed gap between these two branches and determine the main chain in short time.

The reason why separating the voting on mainchain and on block generation is because it could make "stake" decide the branch individually. Because Miners will only compete for the votes of generating blocks, and the votes of mainchain has none effect on their incomes. That makes it not necessary to influence objective of the voting. Besides, it is difficult to do that because the Voters only need to vote their desired mainchain according to the blockchain data. So even if the users are quite centralized, it also can protect the attacks against the main chain effectively.

According to the consensus process above, Miners have to earn more dispersed workers to catch up the randomly distributed signals of Voters on the network if they would like to get more advantages in the competition. Thus, the competition ability is impossible to be simulated in single computers, which avoids the competition by infinitely increasing the performance of mining machines and ensures the fairness.

Like PoS, this protocol doesn't need any external resources. The difference is that when the votes take effect, they are already sealed in the existing blocks. That makes the common problem of PoS such as "nothing at stake" or "stake grinding" resolved.

Another difference from PoS is: Stakeholders could hand over their works to Miners through the voting process. The Miners can help to complete the competition even though the users with low stakes are not so positive. The stakeholders only need to vote once within a voting cycle to make sure that their holding stakes would not miss the voting activities, which can basically avoid the unequal wealth distribution issue as discussed above.

Meanwhile, the form of Miner-Workers can be embedded in the App and website development. Hopefully it helps some developers to get out of the dilemma in which they only can get income from advertisement, or makes a contribution to better user experiences of some Apps and websites. And there are a big number of developers could be the potential miners, that will reduce the threshold for starting and helps the sustainability of blockchains.

4. Compete for Voters

Miners will try the best to win as many as possible Voters to achieve the ability of faster block generation. Some of the Miners may reach agreements with some Voters to cooperate mining other than using more Workers as canvassers. In this case, the protocol provides some alternatives in the transaction structure, with which the Voters can choose type *A*: accept vote canvassing from Workers with the form of broadcast-request; or type *B*: designate one Miner and make him win the voting; or type *C*: designate themselves and work on mining individually. The regular Miners will compete with the Miners with cooperating Voters and the individual ones. It is a competition of the ability of gathering users between type *A* and type *B* Miners, and it is a competition of stake between type *C* and the other two types. Essentially, the three types of Miners compete with each other using the stakes gathered with three different abilities. Dynamic-adjustment mechanisms are introduced through the process of reward distribution to balance the mining power of the three different abilities. The risk that attackers control the network by dominating one kind of ability will be effectively reduced by the dynamic balance.

4.1 Wallet application

Wallet applications can guide or even control the Voters to choose the transaction type and voting targets because of the special user group. However, considering their own benefit, wallet applications would try the best to make the Voters vote to the Miners belong to their suppliers. Which voids the chances for other ordinary Miners to participate in the competition. Therefore, there needs a mechanism to encourage wallet applications to select type *A* to provide a fair competition environment. In this case, the field of "wallet account" is added into type *A* transactions and distribute some of the profits to the suppliers of the wallet applications, but type *B* and *C* don't have this kind of rewards. Any behavior that influences the fairness of the competition in type *A* should be considered as malicious since the wallet applications have the legitimate income. There is extra benefit if the wallet suppliers profit from the system. It encourages the developers to make better wallet applications or, more importantly, provides an incentive to build sidechain projects which may help to build an extendable open system with multi-chains.

4.2 PoS variant

We can also derivate a variant for this protocol when there is only type *C* mining: every Miner mines with his own stakes and it becomes a pure PoS consensus. This variant system also can ensure the fairness without the several common problems under the traditional PoS protocol, except the wealth distribution issue.

5. Reward distribution

Users may make different behaviors and that will affect the network structure if the distribution strategy of mining rewards is different. The distribution strategy is only for the profits from A type mining, because the profits of type *B* are distributed by the Miners, and there is no distribution requirement for type *C* mining. Considering the questions as following:

(1) Miners should achieve rewards according to the fixed proportion of total amount to avoid that the Miners divide themselves to get more benefit.

(2) The weight of stake while distributing will influence the Voters' choice, and absolute fair distribution may not be enough to attract more ordinary Voters if we follow exactly the stake proportion. But we would lose too many high-stake Voters if the stake weight is too low.

(3) It is necessary to make transaction fee as reference to avoid the Voters dividing their stakes to get more rewards after decreasing the stake weight, and it also can provide the Voters with more strategies to choose.

Therefore, the best solution for distribution plan of type *A* mining is to be affected by both parameters of stake and transaction fee, *e.g.* the Miner shares 20% of the total rewards, and the rest will be divided into two parts, 1/3 of which is distributed according to the proportion of transaction fee, and the other 2/3 is distributed according to the proportion of stake. 25% of the rewards from the Voters (*i.e.* 20% of total amount) will be distributed to the wallet account.

However, it is determined by the Miner according to the agreement between the Miner and the cooperating Voters as for the distribution plan of type *B* mining. It is relatively freer, but there are also some restrictions: B-type Miners' rewards must be restricted under the wallet income out of type *A* mining to ensure the benefit of the wallet applications with type *A* is always higher than that with type *B*, in this case, under 20%.

The reward proportion of type *A* Miners should be dynamically adjusted according to the sum of stakes of each type of transactions to balance the power of mining: decrease the proportion if the sum of stakes of type *A* is too high to reduce the number of A-type Miners; if otherwise increase it to attract more Miners of type *A*.

The default transaction fee of different voting plans could be dynamically adjusted according to the average number of the Miners in type *A* or type *B*. If there are more Miners joined, reduce the transaction fee to attract more Voters to match the number of the Miners. Actually, even if there is no adjustment above, the economic principle would also help people to make such balance, just a little slower.

6. Frauds and attacks

(1) Filtered transaction: Miners only pack the transactions which are beneficial to them.

Miners may want to gain more advantages through choosing transactions because the transactions included in each block could influence the competition environment.

First of all, transactions in each block only take up a small percentage of the total amount, so there will be little influence on the statistic results if some changes are made in one block. To solve this kind of problem better, we need to count the sum of stakes of all the Voters in this transaction, which will affect the next block generation interval. The higher stake it has, the shorter calculation period it will have. *e.g.* if we make the calculation period between 0.9-1.1 secs, it would directly affect next block generation speed. In this case, it would be better choice for packing as many transactions as possible, which could disincentive the Miner frauds. This parameter should be adjusted per the average value every once in a while.

(2) It's also filtered transaction, but the purpose is to influence the branch growth speed.

Because the transactions included in each block will also decide the speed of next block generation in the same branch by influencing the parameter "Y", and attackers could decrease the transactions deliberately in a certain branch to affect determination of the mainchain.

Solution: In step 4 of the consensus process, when checking the transaction field "b", other than marking the transactions when they match the tail of the current chain as "y", we also need to mark the transaction that point to the current chain's countdown second, third (quantity adjustable) block as "y1". Both "y" and "y1" marks shall be counted when calculating the value of "Y". In this case, it will compensate the deliberate missing transactions if the latter miners are honest even though the attacker reduces the speed of a certain block generation. And all those missing transactions' "b" fields will point to the countdown second block and backwards.

(3) 51% attack.

The system will be vulnerable under his attack if anyone controls over 50% stakes of the network, which is the same as the PoS protocol. But PoS system can't keep a high participation rate because users have to run full nodes and keep them online to join the competition. Therefore, lower proportion of online stake reduces the security level. In PoND system, stakeholders could join the network maintenance with only one vote during a voting cycle. With a lower requirement of participation, we can keep a higher online proportion of stake and that improves the security of the system.

(4) Simulate Workers: trying to create large number of Worker nodes by simulation and add those virtual nodes to the p2p network to increase the probability of successful canvass.

To deal with this situation, we can make some control when creating the p2p links, *e.g.* each node only needs to build connection with a certain number of nodes with the fastest response speed.

7. Conclusion

Compared with "PoW" and "PoS", this model has the following benefit:

(1) No hashpower competition and no high-energy-consumption;
(2) No "nothing at stake" or "stake grinding" attack;
(3) Provide enough incentives to maintain the network without the wealth concentration problem;
(4) Provide an incentive to develop extend projects which ensure the sustainability and extendibility of the system.
(5) The new way of mining competition helps change the environment of App or website development and improve the user experience of them.

Conflict of Interest

Patent NO. : CN2018102289423.

Notes

[1] The main chain is selected as the "heaviest" branch according to the GHOST protocol.

[2] The "x" mark means the vote of generating blocks, but the "y" mark also need to be checked, by doing so nobody will get any reward if they select a wrong branch, which facilitate users to be more careful in the branch selection.

[3] If there are any blocks from the competitive branches received during the mining process, they should be parallel processed. Miners should not stop working on the competitive branches if the weight of the main chain is less than the competitive chain's weight plus eight in order not to weaken the main chain. And it actually follows their own benefit (possibility to become the main chain).

[4] In terms of the extra storage space for validation parameters and reward distribution, it could be 4 more bytes in the ID and 4 more bytes in the profit of each transaction at maximum, which is fully acceptable; additional work of block validation is going through the last 6000 blocks, and it will not cause a bottleneck with a proper design.

[5] It is necessary to count the block interval between the last transaction and the current block in each account, in unit of a certain granularity, such as 10 blocks, as the adjustment coefficient of stake to control the voting frequency. From zero, every unit time, the coefficient increases proportionally, and only after a voting cycle, such as 6000 blocks, around 100 hours, coefficient can be restored to the maximum; also, the same adjustment coefficient will be added to each UTXO, a nd transaction frequency can adjust the number of stakes with the same approach and parameters as the former.

[6] We could add transactions with pure voting and users should be able to decide whether they want to vote with the current transaction to improve efficiency.

[7] Theoretically the probability of wining is proportional to the number of online terminals.

References

[1] Yj1190590 (/yj1190590). “PoND(Proof of Network Dispersity) BlockChain Project.” Github (accessed 29 April 2018) https://github.com/yj1190590/PoND/blob/master/README_eng.md