ISSN: 2688-9323

The Development History of the Consensus Algorithm on the Public Chain

Yujie Huang

Department of Economic Management, North China Electric Power University, Baoding, 071000, China

220191060911@ncepu.edu.cn

Abstract

Consensus algorithms are the key to solving the consistency problem of distributed systems. Since Satoshi Nakamoto proposed proof-of-work, the consensus algorithm on the public chain has developed rapidly. Taking proof-of-work as the starting point, this thesis consisers 13 algorithms from the perspective of finding and solving problems, hoping to provide a reference for the development of consensus algorithms on the public chain.

Keywords

Public Chain; Consensus Algorithm; Advantages and Disadvantages.

1. Introduction

Consensus algorithms have a long history. In 1959, Edmund Eisenberg and David Gale from The Rand Corporation and Brown University first proposed the consensus problem[1]. In 1980, Marshall Pease, Robert Shostak, and Leslie Lamport presented the consensus problem in the field of computing[2]. The Byzantine Generals problem was first proposed by Leslie Lamport in the article "The Byzantine Generals Problem" [3] describing the problem of fault tolerance and consistency in distributed systems in 1982. The emergence of the Byzantine Generals problem divides the consensus problem into two types: the CFT scenario where only non-malicious faults such as network interruption and machine downtime are considered, and the BFT scenario where malicious nodes are considered[4].

For CFT scenarios, Paxos algorithm[5,6], Raft algorithm[7] and so on have been proposed since the 1990s. But for the BFT scenario, it was not until November 2008 when Satoshi Nakamoto proposed the concept of bitcoin that the use of proof-of-work was the first innovative solution to the Byzantine general problem of Internet scale. Proof-of-work is the prelude to the consensus algorithm of the public chain. Driven by the demand for decentralization, tamper-resistance, traceability and other functions in the financial, medical, public service, Internet of vehicles and many other neighborhoods, the consensus algorithm on the public chain is booming.

The history of consensus algorithms on the public chain can be summarized as the process of discovering and solving problems. Starting from the emergence of proof-of-work, problems are found in practice or simulation, and then new consensus algorithms are generated in the process of solving problems.

2. Beginning: Proof-of-Work

The basic assumption of proof-of-work is that controlling most of the computational forces in the system is harder than controlling most of the entities. The hash function is easy to solve forward and almost impossible to solve backward. Proof-of-work uses this property of the hash

ISSN: 2688-9323

function to select the decision makers. All people who use Bitcoin for transactions are called proposers. The proposer packages the transaction into blocks. The person who solves the hash function, that is, the person with the strongest arithmetic power, acts as the decision maker. The decision maker connects the packaged block to the longest chain to determine the validity of the transaction. Since bitcoin is a single chain, the longest chain prevails. If someone attempts to tamper with transaction records, use a single node to forge multiple identities to attack. To forge a blockchain longer than the longest chain, which would surely be the enemy of all other players and require a lot of computing power. This is almost impossible under the basic assumption of proof-of-work, and thus defends against witch attacks.

Proof-of-work exposes some flaws in the practical application of Bitcoin. First, it is not ecologically friendly. Solving the hash function requires a lot of physical hardware and the guarantee of electric power. The annual power consumption is around 134TWH, almost the power consumption of a medium-sized European country. Second, consensus is inefficient. It takes a long time to solve the hash function and generates a block in ten minutes. The consensus efficiency is low and it is not friendly to small transactions. Third, solving the hash function has no practical use, and computing power and electricity are wasted in vain. Fourth, it has the potential for 51% attacks. The proof-of-work encourages everyone to mine in the mining pool, which leads to the concentration of the computing power of the blockchain in the mining pool, laying a hidden danger of 51% computing power attack. Fifth, it has the post-era operation problem. There is a certain amount of bitcoin. When the bitcoin is issued, the miners lose the reward for maintaining the security of the bitcoin network. How to maintain the security of the Bitcoin network? Sixth, it has a high concurrency problem. When a large number of transactions occur at a moment, the system faces the risk of crash. Seventh, there are block interception attacks, where miners attack each other for their own interests and harm the interests of the system.

3. Solution to High Energy Consumption

The fundamental reason for the high energy consumption in the proof-of-work is to select the decision maker by computing power. However, the idea of selecting the decision maker by scarce resources is worth learning from. Therefore, consensus algorithms such as proof-of-stake, proof-of-capacity and proof-of-burn are produced by changing scarce resources.

3.1. Proof-of-Stake

In 2012, a netizen named Sunny King chose money itself as a scarce resource and launched Peercoin. The cryptocurrency uses a proof-of-work mechanism to issue new coins and a proof-of-stake mechanism to maintain cyber security, which is the first time the proof-of-stake mechanism has been used in cryptocurrency.

In proof-of-stake, computing power were replaced with the currency age, which refers to the product of the amount of money and the time of currency holding. The person with the largest coin age would pledge the money and gain resolution authority to determine the validity of the transaction. To ensure the fairness of the transaction, if the decision maker records an illegal transaction, part of the token will be lost to ensure the security of the blockchain. The age of the coin will be expended if selected as the resolution holder, eliminating the danger that the rich will get richer.

While proof-of-stake has the advantages of being efficient, energy-efficient and impervious to economies of scale, it also has many problems. First, there is a risk of currency hoarding. Given that the person with the most currency age is the decision maker, rational people hoard money and compete to be the decision maker. Second, nothing at stake may occur. When proof-of-stake

ISSN: 2688-9323

forks, rational people will bet on both chains to increase the likelihood of betting on a winning blockchain, making it harder to reach consensus.

3.1.1. Hoarding: Proof-of-Stake-Velocity

Proof-of-stake-velocity has made remarkable contributions to solving the problem of currency hoarding in Proof-of-stake. In April 2014, Larry Ren proposed proof-of-Stake-Velocity. The linear function relationship between coin age and time is modified to an exponential decay function, that is, the growth rate of coin age decreases with time and finally approaches zero. In this case, the age of money does not increase over time, giving hoarders little incentive.

3.1.2. Nothing at Stake: Chains-of-Activity

In order to solve the bifurcation problem in proof-of-stake, chains-of-activity was born. A random shareholder is selected to confirm a new block through an online lottery, and is penalized if both blocks are confirmed, thus preventing a "Nothing at stake" attack.

3.2. Proof-of-Capacity

In addition to proof-of-stake, Proof-of-capacity, proposed in 2014, also reduces energy consumption. Proof-of-capacity continues to uphold the concept of selecting decision makers for scarce resources. Proof-of-capacity replaces scarce resources with disk space. By caching data, it saves operation times and thus energy required for operation. It can be understood that it uses space to exchange for time.

3.3. Proof-of-Burn

Proof-of-burn was proposed in May 2014, which also chose coin itself as a scarce resource, but in a different way from proof-of-stake. Proof-of-burn sends coin to a verifiable unusable address, competing for resolution rights based on the amount of coin sent. The more coins you send, the more likely you are to get the right to decide. Burning certificates reduces the number of coins available and thus the liquidity of the market, but also increases the value of the coin.

4. Improving Consensus Efficiency

Proof-of-luck, proof-of-history and delegated-proof-of-stake can improve the efficiency of consensus. Proof-of-luck generates random numbers based on Trusted Execution Environments (TEE) platform to select decision makers, which shortens transaction verification time and improves consensus efficiency. To prevent double payments, networks need reliable systems to sort transactions, mostly by time. However, due to network delay, relative effect and time dilation, the entire network cannot accurately synchronize time. So, Proof of History used the time it took to calculate SHA-256 as a reference to rank trades, dramatically increasing the time it took to confirm trades. In April 2014, Dan Larimer (BM), lead developer of Bitshares, proposed the delegated-proof-of-stake, with representatives selected by polls of all network nodes and counted in turn, like democratic centralism. It has the advantages of fast and high efficiency. However, if the voting is not active in the delegate selection stage, or malicious nodes appear in the delegate, the security of the system will be reduced.

5. Solution to the Hash Function Without Practical Application

Since solving hash functions has no practical use, proof-of-useful-work was proposed in 2017. proof-of-useful-work transforms solving SHA256 hash operation into difficult but valuable operation in actual production and life, such as training machine learning model and 3SUM problem. It can not only solve practical problems, but also ensure the normal operation of the monetary system.

ISSN: 2688-9323

6. Solution to 51% Attack

The essence of the 51% attack is that someone controls more than 51% of the voting rights in the selection of the decision makers. In order to reduce the probability of 51% attack, the difficulty of controlling more than 51% voting rights can be increased.

The delayed- proof-of-work miners mined for wood and selected the decision makers based on the amount of wood burned. When there is a large amount of wood in the system, the consensus mechanism is like proof-of-stake. When wood stocks are small, the consensus mechanism is like Proof-of-work. To execute a 51% attack, you need 51% wood and 51% computing power. It's very difficult.

7. Solution to High Concurrency

In Bitcoin, a maximum of seven transactions can be recorded per second. When eight or more transactions occur at one time, how can it ensure that all of them are accurately recorded? To solve this problem, IOTA has emerged. IOTA Tangle is based on Directed Acyclic Graph, a new data structure.

When a new transaction is generated, two Tangle ends will be randomly selected at the Tangle ends and the new transaction will not conflict with the existing Tangle ends. If one of the Tangle ends is a fake transaction, it will be ignored and another new end will be selected. If all goes well, the new transaction will be connected to the two terminals selected. The newly added transaction validates the two end transactions that have been joined, and this transaction becomes part of the tangle.

The DAG structure determines extensibility. For every transaction that is added to the tangle, two other transactions will be validated. This means that when there are a lot of transactions, the network doesn't slow down, it speeds up.

8. Operation Mechanism of the System after the Issuance of Bitcoin

It is well known that the total number of bitcoins will reach a maximum of 2100 million in 2140. Miners in the proof-of-work scheme are motivated by financial incentives to keep the bitcoin network secure. After the amount of bitcoin reaches the maximum, stakeholders such as bitcoin owners can maintain the security of the network.

Under such an assumption, Bentov et al. proposed proof-of-activity in 2014, which distributed proof-of-work mining-generated coins to currency holders according to their activity level and encouraged them to maintain network security. The activity degree is related to the age of coin in proof-of-stake. The older the coin is, the more likely it is to get rewards. But at the same time, proof-of-activity makes it possible for people with large currency holdings to control the system, bringing the risk of decentralization.

9. Block Interception Attack

There are block interception attacks in Bitcoin. Miners can boost their earnings by attacking other miners. If all miners chose to attack each other, the sum of their gains would be less than if they had not attacked each other. This attack is an optimal strategy for the individual, but not for the system. This is analogous to the classic prisoner's dilemma in game theory.

The purpose of consensus algorithm is to allow multiple nodes to reach agreement without conflict. Under the public chain, each party has the right to choose the way to maximize its own interests. Therefore, the agreed state should be that each node's unilateral change of strategy will not increase individual benefits. So, this is the Nash equilibrium in economics. Therefore,

ISSN: 2688-9323

consensus algorithm can be improved from the perspective of Nash equilibrium to further improve fairness.

Aiming at the above problems, using Zero determinant (ZD) strategy to optimize the miner's strategic choice, limiting the profit of the miners who only consider personal interest and ignore the system profit, maximizing the system profit [8].

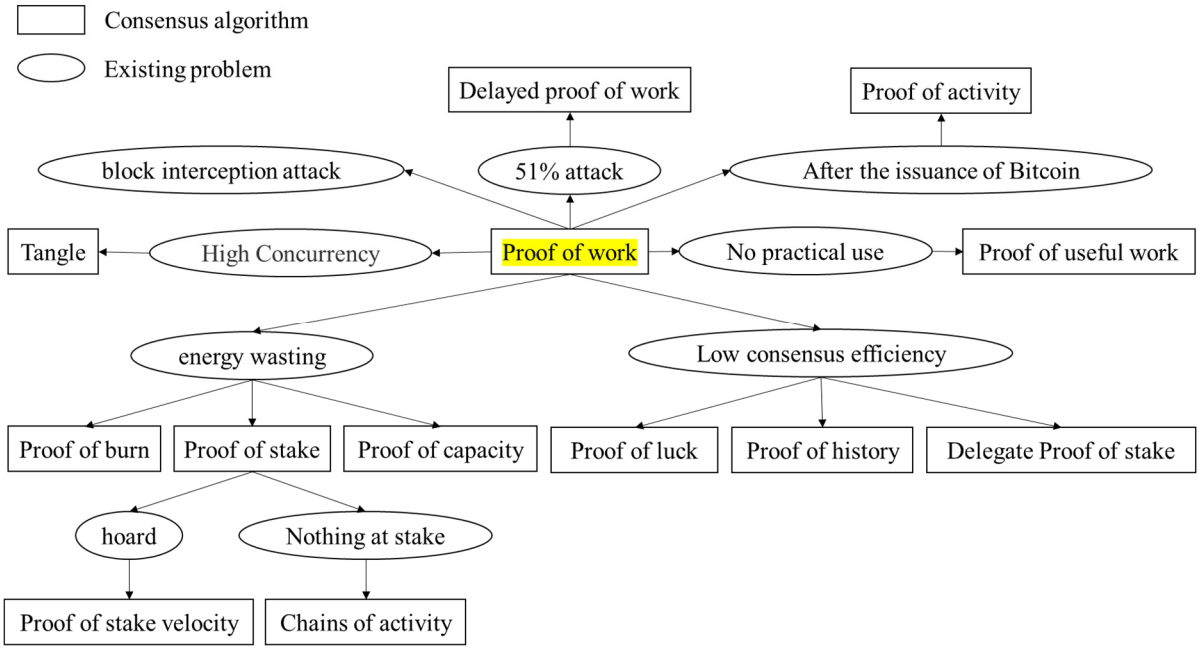

Fig 1. The evolutionary tree of consensus algorithms on public chains

10. Conclusion

The consensus algorithm on public chain starts from the proof of work and finds a series of problems in practice and simulation. Consensus algorithms such as proof of-stake, proof-of-capacity and proof-of-burn are produced to solve the problem of high energy consumption of proof-of-work. proof-of-luck and proof-of-history are generated to improve the efficiency of consensus. proof-of-useful-work is generated because solving hash functions has no practical significance. To avoid 51% attacks, delayed- proof-of-work are generated. Proof-of-activity was created to ensure the normal operation of the system after the issuance of Bitcoin. Tangle came into being to solve the problem of high concurrency in blockchain. Blockchain can also be viewed from the perspective of game theory to further improve the fairness and security of blockchain.

References

[1] Eisenberg E , Gale D . Consensus of Subjective Probabilities: The Pari-Mutuel Method[J]. Annals of Mathematical Statistics, 1959, 30(1):165-168.

[2] Pease M, Shostak R, Lamport L. Reaching agreement in the presence of faults. Journal of the ACM, 1980, 27(2): 228–234.

[3] Lamport L, Shostak R, Pease M. The Byzantine generals problem. ACM Trans. on Programming Languages and Systems, 1982, 4(3): 382–401.

[4] Liu Yihua, Chen Kang. New progress in blockchain consensus mechanism[J]. Application Research of Computers, 2020(S02):6.

[5] Lamport L.The part-time parliament[J]. ACM Trans on Computer Systems,1998,16(2):133-169.

[6] Lamport L. Paxos made simple[J]. ACM SIGACT News,2001,32(4):51-58.

ISSN: 2688-9323

[7] Ongaro D, Ousterhout J. In search of an understandable consensus algorithm. USENIX Association, 2014.

[8] Tang Chang-Bing, Yang Zhen, Zheng Zhong-Long, Chen Zhong-Yu, Li Xiang, Game dilemma analysis and optimization of PoW consensus algorithm. Acta Automatica Sinica, 2017, 43(9): 1520-1531.