CS6290 55323707 Reading Summary 2

Yu SU
Dept. of Computer Science
ID:55323707

I. Summary of Paper [1]

A. Problem statement

In the distributed computer system, it is harder to handle each nodes correctly, so the problem of this paper is that how to make the computer system maintain the reliability with coping failed component.

And the paper abstracts the problem to the Byzantine Generals Problem. The each one in the army need to get a right consensus in the case that there are some traitors in the army, which is just similar to the distributed system problem. The paper tries to get the regulation of this problem and figure out some algorithm to solve it.

B. Problem Significance

This paper is very meaningful in both distributed system and fault tolerance problems.

There are many problems and restricts in central system. So it is necessary to research on distributed system. Then because of the nature of distributed system, we need to make sure it could work reliabily. To achieve this, the first is try find a standard to evaluate how a system could work well. And this paper is a research about it. So this research is very meaningful.

What's more, besides the aim to improve the distributed system, it is also meaningful for the research of logic. To realize consensus without trusty center in practice is very hard. So try to figure out a protocol to solve this problem is also meaningful a lot.

C. State of the Art

The paper was written in 1982. At that time, the distributed system is a hot topic of research. Before this paper there is no reliable standard of fault tolerance which could make sure a distributed system could work well or not. The reason why this problem was not solved before this paper is that the distributed system research was in the starting stage.

The most of the fault tolerance systems before this paper is trying to use the redundancy to fix the bad part, which is inefficient.

The consensus in distributed system was also in the starting stage before this paper. Although there are some attempts , no one could really solve this problem in practice.

D. Contributions

This paper proposed an impossibility results to realize the stable distributed system and also proposed some solution under different assumptions.

The paper proposed and proved that the impossibility results for the Byzantine General Problem. It proposed a consensus theory, using fault tolerance instead of consensus of all members, which is imposed by many distributed system in the future, lick bitcoin.

To solve this problem, the author figure two algorithm under different assumption. One is oral messages, under this assumption, there are no certificate of their received message. The paper proposed an algorithm OM(m), which could solve this problem if there are 3m+1 or more generals in the presence of at most m traitors. What's more, he also proposed an algorithm SM(m). Under the assumption of this algorithm, the signature could guarantee the authentication, which is more like the real computer system. Finally, Under the scenario of signed messages, he proposed and proved the algorithm could solve the problem no matter there are how many traitors only if there is one more generals. And these algorithms are developed to some other advanced algorithm like practical Byzanine fault tolerance.

To make the problem is more similar to the distributed system problem, he also considered the lost messages problem. The author imposed the scenario where communication paths may be missing. He proposed and proved two more theorems which could solve Byzantine generals problems with the missing communication paths.

At last, the author proposed an idea to realize the reliable system, which is using several different processors to compute the same result with the majority voting, which idea is still imposed in many distributed system today. The theory of Byzantine fault tolerance is meaningful. Even today, some aircraft system use the Byzantine fault tolerance to ensure it works well.

E. Remaining Questions

There are some problems for Byzantine generals problem implementation.

On the one hand, even though the theory of the solution in this paper is perfect, it is really hard to realize. For example, the signed message algorithm assumption A4 is hard to achieve without a center authenticity certificate and the paper did not give the solution to implement these.

On the other hand, the largest problem for this theory is that all the solution based on the communication among the nodes. Even though there are some optimization in the later algorithm based on this could still not solve the expense of communication problem. This is the really reason why the soluton in this paper is not popular today.

II. Summary of Paper [2]

A. Problem Statement

With the development of the network, there are more and more transactions on the internet, which gives a lot of pressure for the bitcoin. As the largest concurrency on the internet, bitcoin could only generate one block in 10 minutes, which is definitely not enough today. The problem is bitcoin has bad scalability.

There are two reasons, first is speeding up the growth of block is not good for the bitcoin. On the one hand, is will cause a lot of waste because there will be more useless computation on orphan block. On the other hand, it will also cause some security problem because they disperse the honst power. The second reason is any change in bitcoin need a hard folk, which is not only a simple yes or no problem.

B. Problem Significance

From 2009, bitcoin become more and more popular and the general interest in the currency and its use has been slowly increasing. And the researchers want to make it fit to the present circumstance.

However, the main obstacle of bitcoin is the lack of scalability, which is hard to fit the higher transaction rate and quicker transactions process.

One of the cause of restricting the scalability of bitcoin is the security problem. To improve the efficiency of bitcoin and guarantee the security, it is very meaningful to research on it.

Only if overcome this challenge, the blockchain could fit nowadays society and have a longer development.

C. State of the Art

Before the paper, the GHOST has been adopted and a variant of it has been implemented as part of the Ethereum project.

At that time, the consensus that implemented by Satoshi Nakamoto looks not fit this era anymore. 10 minutes one block with 1 MB size is not efficient enough. So There were a lot of researchers trying to improve this, such as bitcoin-NG.

The problem was not solved before, one of the reason is the use of bitcoin was not as much as today, the original efficiency was enough. There was no intense requirement for bitcoin to update. The other reason is there was no a fair evaluation on the performance of it to prove the new algorithm could do better. What's more, besides the technology factor, bitcoin is a system contains a lot of wealth.So each choice of bitcoin is driven by the behind capital. It is harder to implement a new technology than to propose.

D. Contributions

The paper imported the math model and define the primary measure of bitcoin's scalability as the number of transactions per second(TPS). Under this model, he find there two ways to increase the TPS, increasing the block size or block rate. The paper combined the security problem with the protocol and find no matter which ways the security of the system always drops.

To solve this problem, the paper proposed the new algorithm, The Greedy Heaviest-Observed Sub-Tree(GHOST), which could maintain the weight of the block off the longest chain to make the security will not be influenced by the increase of throughput. This algorithm change the bitcoin consensus from the longest-chain to heaviest-chain.

To evaluate the performance of it compared with the longest-chain selection rule, the paper imported math model to evaluate their performance. Because the system performance is dependent on network. He use the lower and upper bound with simulation to evaluate the growth rate of the system.

Besides, the paper also focused on the weak attackers problem compared with two system with different selection method. To avoid this problem, the paper proposed and proved the new acceptance policy to fit its new selection rule.

Last the author gave more details about the implementation, such as the hard folk for deployment, keeping the total rate of block creation, fees for uncle block and preventing amplified denial of service attack. All are very meaningful not only in this system but the concurrency. Some of the ideas are used in the other concurrency today.

I think the best contribution is the paper used a lot of math theory to represent its idea and also give all his theory math proof, which is what most researcher on blockchian lacked at that time. Because of the incomplete theory of bitcoin in Satoshi Nakamoto's bitcoin paper, there need more theory not just need test in practice, which is costly and inefficient.

E. Remaining Questions

The largest remaining question of GHOST is the uncle mining problem. [3] To make the system secure, the off-chain block is still weighted, in other word, mining them could still get little reward. And because of this nature of GHOST, there is an uncle mining problem.

As the name implies, the whole blockchain is seen as a family tree. The main chain is the immediate family, and the off-chain block is the uncle block. The attacker try to mine the uncle block instead of the main chain, which means he could get more reward than he should get because there are less competition of working on uncle block. This problem eventually will cause the damage of the concurrency ecology.

Besides the uncle mining problem, implementing GHOST on bitcoin still need a hard fork, which means it is practicable only if the majority of the computers accept this. It is hard to implement because there are more factor beyond technology. And from the result of today we know, this approach did not get a success on bitcoin.

References

[1] L. Leslie, S. Robert, and P. Marshell, "The byzantine generals problem," 1982.

[2] Y. Sompolinsky and A. Zohar, Eds., *Secure high-rate transaction processing in bitcoin*.

[3] F. Ritz and A. Zugenmaier, "The impact of uncle rewards on selfish mining in ethereum."