I posted a design paper of a consensus protocol on Bitcointalk a few days ago, but didn't receive any response. I am not sure whether it was because the protocol is not valuable or because I didn't explain it well. I hope it was the second reason. So I'm going to explain it here in a simpler way: from the begining of my research. I hope there will be more people to be interested.

I decided to design a new protocol begining with some personal understanding with the debates of the existing protocols.

The first debate: PoW or PoS?

To my understanding, I prefer PoS, because the "work" of PoW is an uncontrolable external resource. Despite the energy consumption issue, using an uncontrolable external resource as a competitive material will probably cause large-scale collusions, which could impact the security. "mining pools" is an example of that. From this aspect, PoS uses an internal resource of the system. First of all, the total amount of stakes is fixed, which leads to a big advantage and I will introduce that later. Secondly, as a fundermental property of an account, stakes will not be shared among users in quantities. Therefore, large-scale collusions will be very difficult to occur.

The second debate: Chain, BFT,DPoS or DAG?

As far as I know, there are two ways to choose validators in all consensus systems: through competition (e.g. more hashpower or more stakes) or through cooperation (e.g. delegated by stakeholders). The former way results in wealth concentration (plenty of incentive) which causes user reductions, or lack of competitors (not much incentive) which causes security reductions; the latter results in the probability of large-scale collusions as PoW does (referring to this article), which causes security problems too. Security reductions are unacceptable for currency-functional chains and user reductions are hardly acceptable for public chains. Sharply reducing the validator nodes (BFT and DPOS) for perfomance will greatly aggravate those problems, so I think they should not be used in currency-functional public chains.

Is it possible to prevent all of those problems? It was the first question for me to find out. The answer is yes. The key is to let lower-ability users compete with higher-ability ones through a cooparative process (stake accumulating) using a chain-based mechanism, which ensures security and participation rate at the same time. From then on, I decided to design a new consensus protocol. I have thought DAG can do the same, but didn't find how to synchronize the state of stakes in an asynchronous system. So I chose the chain-based system in my design.

After determining the structure of chain-based PoS consensus

, the second goal was to compensate its defect (mostly NaS problems) and keep the advantage of PoW (e.g. efficient verification and objective boot straping) as far as possible. As a result, present NaS problems

are solved; objectivity

basically remains;verifications are still not as effecient as in PoW but it doesn't affect important functions such as cross-chain verifications.

My next goal was to consider what other features can be applied in my design.

First of all, scalability

. Although not being able to achieve the performance of BFT or DPoS and the scaling ability of DAG, chain-based consensus has its ways to expand the scale, which are multichain solutions such as side-chains or sharding. Multi-layer structures are actually better for safty factors, but to my understanding, a lack of clear and simple profit model keeps the expanding projects from being widely used (because the currency value is locked). Unpurposely, under the mechanism of accumulating stakes, the wallet applications will be involved in the mining process sothat their provider could directly profit from the system. It just solves the problem of profit model and will change many things. I hope it is helpful for scaling solutions.

Explicit finality

is an other important feature that brings many advantages such as fast confirmation and avoiding historical attacks. It's a feature that most (I'm not sure is it all) of the chain-based protocols don't have. Using the property of fixed amount of stakes that is mentioned above and a double voting method, the feature of explicit finalities is successfully applied.

That's all I have to explain. In short, PoAS is an optimized chain-based PoS

protocol. To my understanding, it should be valuable for the use of cryptocurrencies.

All suggestions and opinions are welcome! thank you for your time!

breif introduction and the full paper are here:

https://github.com/yj1190590/PoAS/