Consensus Algorithms in Blockchain

Eric Adhikari

In a blockchain network, it is essential that everybody agrees on the state of the network. Every running node should have the same state of logs and chain with them. Consensus is a way of agreeing on everything that happens in the network. For example, if I have 10 ethers and I received 5 ethers from another person, I now have a total of 15 ethers. However, another node could have an old log or chain, which can make it think I have 10 ethers, creating a problem. Thus, we need consensus to establish a single truth in the network and agree upon it. We also need consensus to deal with malicious nodes, network latency, server crashes, and network partitions, which are common in distributed systems.

Proof of Authority

In Proof of authority(POA) a trusted set of node are selected which acts as validators and only they can propose the block which is added to the chain. this effectively removes the problem of malicious nodes as validators are trusted entities.

Also the there is a high reliability of producing blocks at a fixed interval, unlike proof of work where we particularly don't know the hash rates of nodes or pool of nodes which may result in unpredictable block generation, POA ensures a fixed and known time interval for block production which makes the network more reliable and perform better.

POA in Etherium

In Etherium There is a test net named Ropsten which was based on proof of work , test nets are created to mimic the main net conditions. All the ethers mined on the test net is of no value, even if people spent real money for GPU's and Electricity for mining the outcome is of no economic insentive this problem leads to really low hash rate on the test net. And if someone tries to point its asic miners (miners with hardware specific for mining) at it then network is kind of screwed. This happened in early 2017 where spam attackers raised the block gas price by submitting transactions with very high gras prices. which lead to higher transaction cost, network congestion and lower gas transactions were left in delay as they were of little revenue for miners.

So an alternative test net was set up after the attack in ropsten which were :

1. KORAN
2. RINKEBY

0.1 KORAN - Authority Round (Aura)

In aura we have a set of validators and we go through them sequentially. the time in aura is divided into descrete steps where,

$$\text{step} = \frac{\text{Unix time}}{\text{length of time}}$$

$$\text{step } 1 = \frac{5}{5}$$

$$\text{step } 20 = \frac{100}{5}$$

These should be synchronization of clock among the validators, if different validators have different steps the that could be a problem.

we can figure which validator to choose in each step :

$$index \equiv s \pmod{n}$$

where s is step and n is the no. of validators this produces a number within the range [0, n-1] which is taken to choose the validator by index for that step

$$\text{validator}[1 \pmod 5)] = \text{validator}[1]$$

Aura provides a concept of finality, Once a block is finalized and majority of n validators have stacked their block on top of it the block is finalized and cannot be reverted.

0.2 RINKEBY - Clique Engine

The goal of clique was to standarize what a POA algorithm looks like so that it can be implemented in multiple etherium clients and used in private and public blockchains. It is used in Rinkeby and Goerli Test net.

There are a set of validators group which are in predefined rotation schedule to propose a new block, they are called inturn validators where as others are called outturn validators, both can propose a block but if i am the in turn validator then my block weighs more. Block have higher difficulty depending on who made them :

- Difficulty-inturn = 2
- Difficulty-outturn = 1

the choice of block for clique is block with highest difficulty thus it favors chain with block produced by inturn validators.