Disposition 12: Nakamoto Style Consensus

Mathias Ravn Tversted

January 14, 2020

Table of contents

Nakamoto Style Consensus

Nakamoto Style Consensus

- Uses Totally-ordered broadcast with Lotteries
- Can be considered a kind of synchronous protocol

It should have the following properties:

Properties of Nakamoto Style Consensus

- **Autonomy**: System should not be controllable by single entity. Should not have *single point of attack*. Anyone should be able to join and leave, and the system should not be impacted.
- **Self-incentivization**: Running servers and maintaining the system costs money. The system should incentivise people to run and maintain the system. If the currency is secure and valuable, it will incentivise people into running the system.

Properties of Nakamoto Style Consensus

- **Peer-To-Peer Friendly**: Peer to peer networks are autonomous and unpermissioned. These are ideal implementing the blockchain. Communication is done by flooding messages to neighbours.
- **Denial of service attack resistant**: Peer-to-peer networks are open, so peers cannot hide behind firewalls. Individual peers are vulnerable to denial of service attacks. Instead of everyones IP being known, peers simply know their neighbours and flood messages. This means that denial of service attacks on individual peers do not impact the stability of the system

Three Components:

Nakamoto Style Consensus has 3 components:

- Distributed Lottery
- Tree of Blocks
- Longest Chain Rule

Distributed Lotteries

Distributed Lottery

The lottery is a distributed leader election. Synchronous TOB has a leader in each round sending out a new block. Round robin does not work in systems like blockchains.

Local computation only: participants only need to know the current state of TOB to participate. They can participate without communicating with others. There can be multiple winners or even no winners.

Non-predictability: No one should be able to predict *who* wins, or they could be a target of a ddos attack for example.

Public Verifiability: Once you won, the peers should be able to verifiy that this is true

Distributed Lottery

One message: When you win, you should only send one message along with the next block. Proof should be associated with the block, so that it cannot be misused Verification key is embedded into the block.

Sybil Attack Resistant: It should cost something to enter the lottery, so that an adversary cannot enter enough to be extremely likely to enter

There are two different ways of deciding winners generally speaking. They are *Proof of work* and *Proof of stake*.

Proof of work

Proof of work allows people to do computation in order to get tickets for the lottery.

- Let A be last block, H a hash function, then $a = H(A)$ be the block we want to extend. a is a pointer to A.
- To win the right to extend the chain at A, compute $y_c = H(a, vk, c)$ for Verification key vk.
- c is a counter. Using SHA256, this gives 256-bit y_1.
- Good hash functions behave like "random" functions, therefore y_1 can determine if vk won the lottery.

Proof of work

- Hardness level h. If first h bits are 0, you win. If y_c is random, then probability is 2^{-h}
- Giving each vk one try to compute y_1 is not sybil resistant.
- Set hardness higher, such as 30 and let everyone go nuts. This prevents Sybil attacks

Proof of work is environmentally problematic, because it uses an enourmous amount of power.

Proof of stake

Amount of currency determines lottery chances.

- Each account vk has a stake a, which could the amount of cryptucurrency.
- All parties agree on when slots will be run. To check if vk won the lottery, it computes $Draw = Sig_{sk}(LOTTERY, slot)$. Where LOTTERY is the string "lottery"
- Value of the draw $Val = a \cdot H(Draw)$ where $H(Draw)$ is a random value. If $Val \geq Hardness$, congratulations you win the lottery.
- Send along $Sig_{sk}(B)$ so that the draw is bound to the block. Others can now verify that you won the lottery.

Tree of Blocks and longest chain rule

Tree of blocks

When the leader sends out a new block. It contains a proof that it won the lottery, along with what it thinks is the previous block. This creates a chain of blocks, but when several people win the lottery, this creates a tree. This is fixed by the *longest chain rule*, where the longest chain available is extended. Where there is a tie, a deterministic heuristic is used.

Properties of the blockchain

Because a rollback of a branch is possible, a transaction cannot be considered final until it is no longer able to be rolled back. We now introduce the following

- **Super block**: A super block is a block produced by an honest party which won a timely lottery with no other winner. This can not result in branching, so they sit at different heights in an honest tree.

Properties