POS Models and Simulations

POS – Proof of Stake

The most common form of POS uses the number of coins a miner owns as his stake. One miner is stochastically selected at a time to add one block to the chain. The probability that a miner gets selected should be proportional to the amount of his coins or stakes. A simplified POS miner selection process can be expressed as

Hash(Prev Block Hash, Miner's Public Key, Current Time in Seconds) < Difficulty Factory * Coins

Each second the left hand side produces a number. If the number is smaller than the right hand side the miner is selected to create a block. The Difficulty Factor can be adjusted so that one miner is selected every 10 minutes on average like Bitcoin. Like POW of Bitcoin, the longest chain is normally the consensus.

A Simple Model

The POS system described above can be simulated by a simple model.

- There are total M miners in the system.
- All miners have equal amount of stakes.
- Selection of miners is by running a random number generator. Each second every miner generates a random number between 0 and 1. One block is added to the chain if any miner has generated a number smaller than the difficulty factory = 1/M/10. On average 1 block is created every 10 second. If more than one miners have numbers smaller than the difficulty factor (there is a fork in this case) then only 1 block is added instead of creating a fork.
- Multiple voting is allowed.

The T-SQL code of this model can be downloaded from
https://github.com/txngrid/posmodels/blob/master/simple_model.sql.

Simulation Results

1. The results of 2 runs of 100 (M = 100) miners for 10,000 seconds are presented in Fig. 1. Observations:

    - The growth of the chain is not smooth.

- The growth rate is about the same in the long run if the number of miners is the same in the system.
- There are about 40 forks in 10,000 seconds simulation. It is shown in one blockchain in Fig. 1. This is close to theoretical result based on Poisson distribution. For 10% success rate 2 miners generate a number smaller than the difficult factor at the same time is 0.45%, which is 45 in 10000 seconds. This means forks naturally happen due the stochastic process.

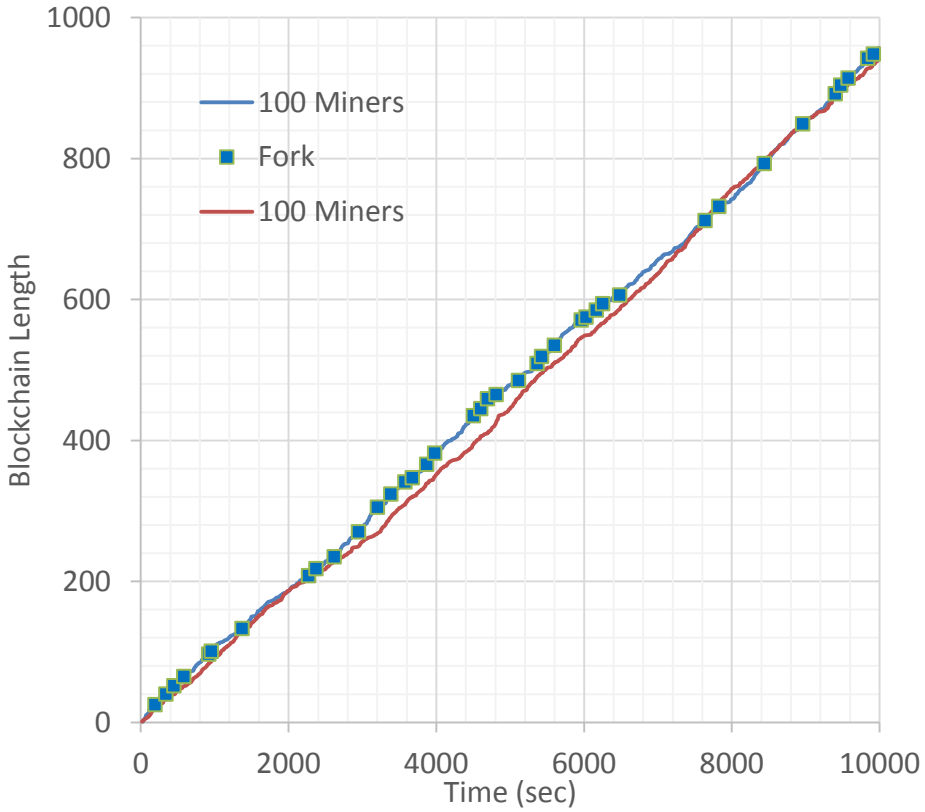

Fig. 1. 2 runs of 100 miners (M=100) for 10, 000 seconds.

2. Fig. 2 is the results of 100 vs 90 miners. 100 miners work on one chain and 90 miners work on another chain at the same time. Observations:

   - Branches will diverge eventually (there will be a winning chain) if there is a small miner percentage difference between the branches.
   - In short term (a few hundred blocks) it is difficult to select the winning branch if there is only 10% difference on miner voting between branches (or stakes working on the branches). Chain reorganization can be frequent (each crossover in Fig. 2 can be a reorganization). This will make it harder for miner not to double voting since there is no clear winning branch in the short term.

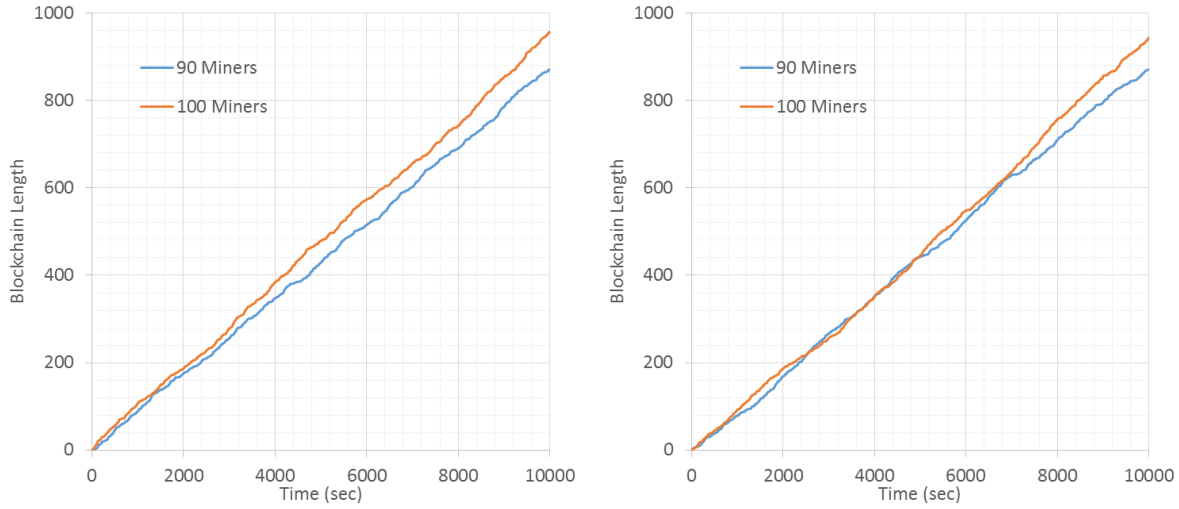

Fig. 2. 100 miners (M=100) vs 90 miners (M=90).

3. Increasing the number of miners (M=1000 and M=900) does not change the simulation results in any significant way.

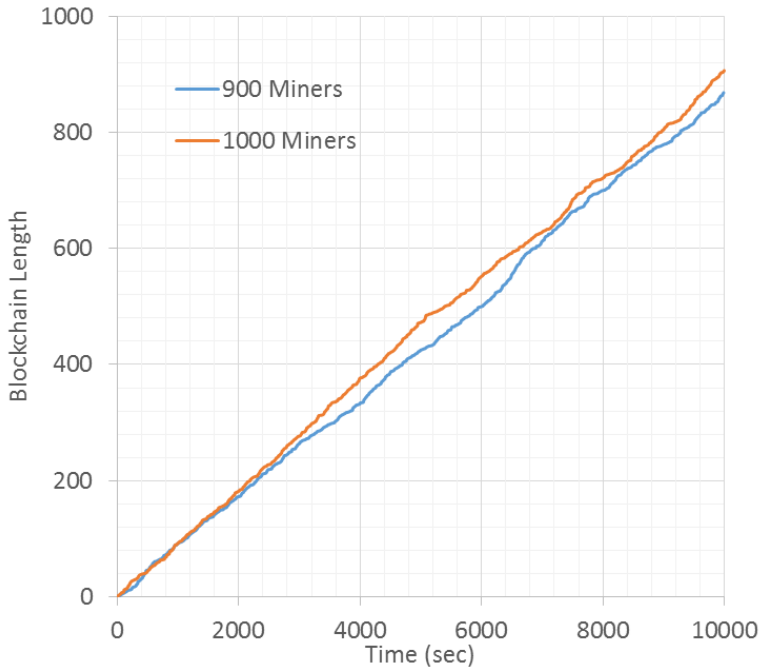

Fig. 3. 1000 vs 900 miners.

4. Summary. Not only does multiple voting in POS lower the security but also it makes it hard to distinguish branches that have small miner percentage difference. In short term it is difficult to tell which branch is the winner and chain reorganization can happen frequently. It is not easy for miners to select a chain to work on even if they want to be honest, and also they will be forced to switch chains if the percentage of multiple voting miners is high in the system. Considering the forks that occur naturally as show in result 1, the ending result of a POS system will be more like a block tree than a chain as in Fig. 4. It will be hard to differentiate the branches unless there is a way to control multiple voting.

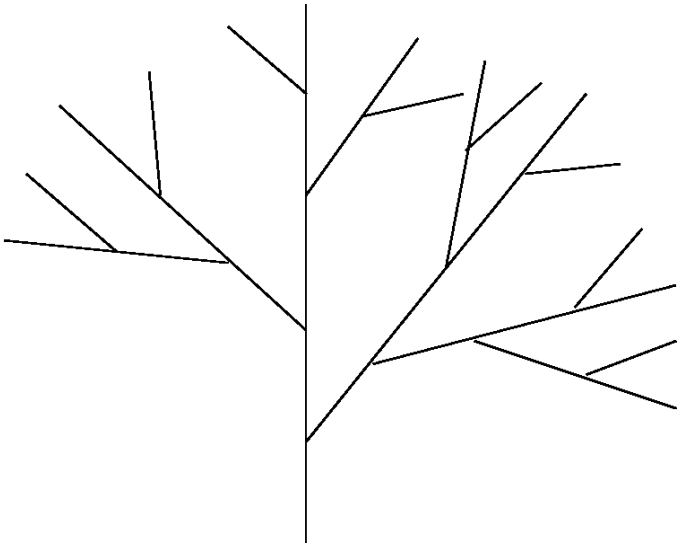

Fig. 4. The block tree created by POS stochastic process and multiple-voting.

None of the results is surprising given the nature of the stochastic processes. The simple model and simulations just make it easier to envision how blockchain is growing.

Get Rid of Stochasticity

Here is one way to smooth out the fluctuation in POS generated by the stochastic process. Instead of one miner mining a block at a time, a fixed number (N) of miners are selected to sign each new block. The miners are selected based on his "distance" to the previous block hash

Distance = hash (prev block hash + miner's public key) / (number of miner's coins)

The first N miners with the shortest distances are selected. The smaller the sum(N distances) the greater the number of miners (total miner's coins, to be precise) it can represent - the more miners participate the smaller the sum of the N smallest distances. At the end of every 10 minutes the block

with the smallest sum is the winner. In case of a fork, the chain that has the smallest sum(block distances) is the consensus, since it represents the most miners.

A Model with no Stochasticity

This POS system can be modeled by this model.

- There are total M miners in the system.
- All miners have equal amount of stakes.
- Selection of miners is by running a random number generator. Each time every miner generates a random number between 0 and 1 – this is the miner's distance. The first N miners who have the smallest numbers are selected to create a block.
- The block distance = sum(N smallest numbers).
- The chain distance = sum(block distances).
- Multiple voting is allowed.

The T-SQL code of this model can be downloaded from https://github.com/txngrid/posmodels/blob/master/multisig_model.sql.

Simulation Results

The simulation results of (M = 100, N = 20) and (M = 90, N = 20) are presented in Fig. 5. It can be seen that

- The growth of blockchain is much smoother without stochastic process. The total number of blocks in Fig. 5 is the same as in the simple model which is 1000 blocks.
- Branches diverge much faster with the same percentage of miner difference (10%) as in Fig. 2. It is easier for miners to pick a winning branch to work on.
- There are fewer crossovers between branches even in short term. This means fewer blockchain reorganizations.

With the improvements above honest miners with a 10% percentage advantage will have little difficulty to identify and work on the winning branch and secure the blockchain. A small percentage of more miners can be a deciding factor in a POS system like this with no stochasticity. On the other hand it works the same way for attackers. A small stake percentage over the honest miners can undermine the security of POS. Consider this situation, 50% of all miners are double-voting, an attacker controls 30% of the stakes, and only 20% are honest miners. There will be 80% (50% + 30%) miners working on the attacking branch and only 70% (50% + 20%) on the honest branch. The attack will take control of the whole blockchain in this case. Smoothing out the fluctuation of the chain growth does not solve the double-voting attack problem. We still need a way to get rid of double-voting in POS to make it as secure as POW.

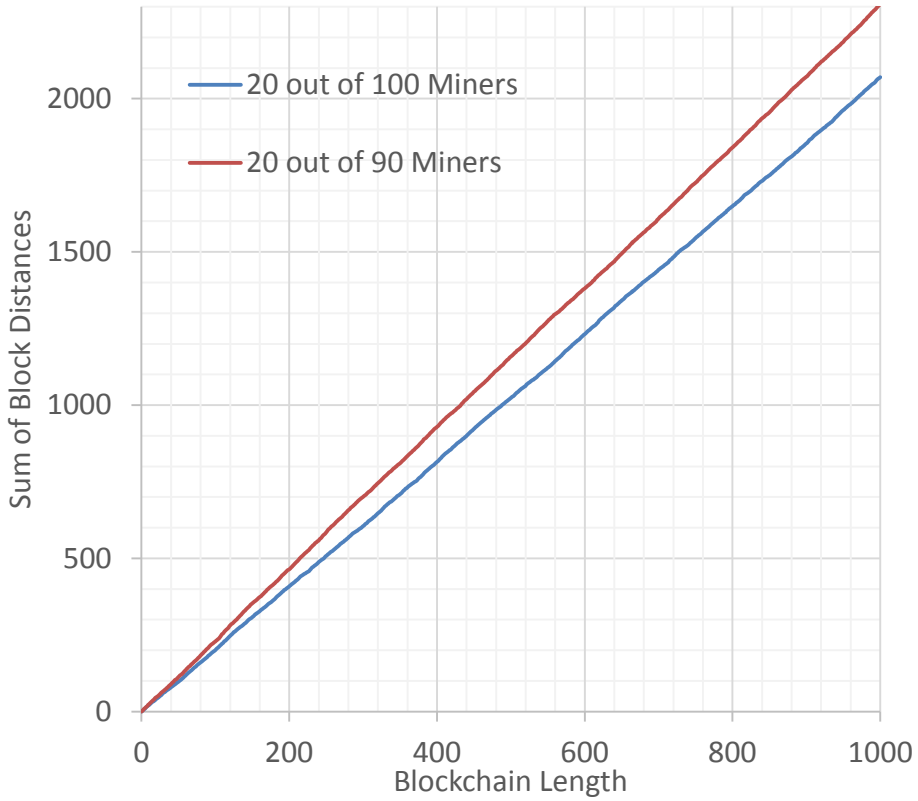

Fig. 5. Simulation of 2 blockchains created by 100 miners (M=100, N=20) vs 90 miners (M=90, N=20). 20 miners are selected to create blocks in both cases.