Lecture 7: Financial Aspects of Proof of Stake

Tarun Chitra

Gauntlet Networks

July 5, 2022

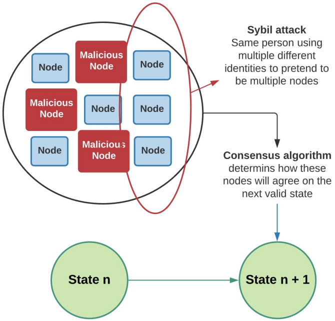
Outline

Sybil Resistance Mechanisms

- In pseudonymous environments, the most important safety mechanism is *Sybil Resistance* (SR)
- SR: user cannot split their resources R, distribute R to multiple identities, and earn more rewards vs. not splitting R
  - As a miner, I can't split my mining resources and get more block rewards or transaction fees
- The most popular SR mechanism is *Proof of Work*

What is Proof of Work?

- Recall: Blockchain = miners competing to add txn blocks
- Why do we need Sybil Resistance?
  - User splits into clones, unfairly increasing prob. of winning
- Proof of Work: Miner submits a block B with a hash(B) matching a pattern (*e.g.* ends in d zeros) wins that block



How is PoW a Sybil Resistance mechanism?

tl;dr: Hashing superpolynomially difficult to improve via splitting

- Main Assumption in PoW: Hash function $h : \{0,1\}^* \rightarrow \{0,1\}^n$ ensures $\mathbf{Prob}[h(x) \text{ ends in d zeros}] \approx \frac{1}{2^d}$
- $d \in \mathbf{N}$ is the *difficulty*
  - Adjusted as fn. of how fast blocks are produced
- Need $\Omega(2^{d/2})$ parallel hashes (splitting of resources) to have appreciable probability of finding block faster
- Difficulty adjustment: Bitcoin adjusts d based on how much hashpower is present so that $2^{d/2}$ is economically unfeasible

Proof of Work: Pros and Cons

- PoW: most common consensus yet controversial mechanism
- **Pros**
  1. Makes decentralized network creation easy for new participants (just need electricity)
  2. Identities of miners never have to be committed to
- **Cons**
  1. Uses a lot of energy, especially relative to centralized systems (but there's nuance here!)
  2. Limitations to speed, bandwidth that can be processed by PoW
- **Can we do better?**

Outline

Sybil Resistance and Sampling

Probabilistic formulation for PoW:

- n players w/ *hash power* $h_i(k) \geq 0$ at block height $k \in \mathbf{N}$
- Collision resistant hash function guarantees that player i is chosen for block k with probability

$$p_i(k) \approx \frac{h_i(k)}{\sum_{i=1}^{n} h_i(k)}$$

- If i splits into i_1, i_2 $h_{i_1} + h_{i_2} = h_i$, i's probability of winning is the same (*proportional allocation*)

Sybil Resistance and Markov Sampling

- This is a Markov process that draws block producer $i \sim \mathbf{p}(k)$
  - Conditional on $h_i(k)$, there's no history dependence
- Suggests replacing hash power sampling with a Markov Chain sampling the same distribution
- *Idea*: Can we *simulate* the hash power lottery by replace hash power (= energy) with other resources?
  - *e.g.* Markov Chain Monte Carlo
  - What if it was a *digital resource* like a token instead of energy or hard disk space?

Simulating Proof of Work

Can we cryptographically sample $\mathbf{p}(k)$ w/o using physical resources?

- Need to know two things:
  1. Initial resource distribution: $\boldsymbol{\pi}(0) \in \mathbf{R}_+^n$
  2. Rewards distribution: $R : \mathbf{N} \to \mathbf{R}_+$
     $R(k) =$ block reward at height k

- How do we sample the kth block producer?
  - Let $F_k^{-1} : [0,1] \to [n]$ be the inverse CDF of $\mathbf{p}(k) = \frac{\boldsymbol{\pi}(k)}{\|\boldsymbol{\pi}(k)\|}$
  - Given uniformly random $u_i \in [0,1]$, choose kth producer i_k as

$$i_k = F_k^{-1}(u_i)$$

Given $\boldsymbol{\pi}(0)$ and a way to sample i, we can simulate PoW:

Initialize $\boldsymbol{\pi}(0), R(k)\ \forall k \in \mathbf{N},\ \forall i,\ \pi_i(0) > 0$
for $k = 0$ to N **do**
  $\boldsymbol{\pi}(k+1) \leftarrow \boldsymbol{\pi}(k)$ — Initial next stake distribution
  $i_k \sim \frac{\boldsymbol{\pi}(k)}{\|\boldsymbol{\pi}(k)\|_1}$ — Sample block producer via inverse CDF
  $\pi(k+1)_{i_k} \leftarrow \pi(k+1)_{i_k} + R(k)$ — Reward winning producer
end for

Proof of Stake: Simulating PoW (sort of)

- 2011: Proof of Stake first proposed in the BitcoinTalk forums (w/o mechanism for sampling)
- 2015: Use **Verifiable Random Functions (VRF)** for cryptographic sampling of π
- VRF allows n parties to sample $u_i \sim \text{Unif}([0, 1])$
  - Verifiable, private, non-manipulable
  - *e.g.* use private key as the seed to a PRNG; generate ZK-like commitment that anyone can verify using my public key
- Stake distribution π is public in the ledger $\implies$ we can use u_i and inverse CDF to simulate PoW

Proof of stake instead of proof of work
July 11, 2011, 04:12:45 AM
Merited by ETFbitcoin (3), Vod (2), webtricks (2), d5000 (1), drays (1)

What is the stake distribution?

- Take a step back: What is $\boldsymbol{\pi}$?
  - Distribution of *coins* in the system
  - π_i is the number of coins held by the ith address
- Fundamentally different than PoW:
  - PoW samples hash power distribution to generate new coins
  - But coin and hash power (resource) distribution can diverge
    - *e.g.* Selling coins doesn't impact hash power distribution but changes coin distribution
  - Not true in PoS by construction

Proof of Stake: Pros and Cons

- Benefits of a single coin and resource distribution (PoS)
  - Lower latency, higher throughput
  - Easier to add finality (e.g. Tendermint, HotStuff)
- Negatives that don't apply to PoW
  - Financial properties make PoS less secure
  - Distribution $\mathbf{p}(k)$ must be public and known to all users in PoS

PoS v. PoW

Let's summarize the differences between PoS and PoW

- PoW is partial information (you don't know $\{h_i\}$ for all players), PoS is full information[1]
- PoS relies on *Adpativity* (*e.g.* resources need to be live at all times) — [LPR20] show an impossibility theorem for safety, liveness, and adaptivity
- Financials outcomes are different because coin and resource distribution are different

[1] Except w/ homomorphic encryption [BEHG20]

Outline

Notable Financial Differences between PoS and PoW

Three main financial distinctions between PoS and PoW,

1. **Concentration of Wealth**: PoS currencies have more extreme wealth concentration than PoW
2. **DeFi cannibalizes security**: Yields from protocols built *on* of a PoS chain can cannibalize security from the base protocol
3. **Derivative assets provide easier access to returns**: PoS derivatives, while dangerous, allow for a level playing field (extra material)

How does one compound wealth in PoW?

Compounding of wealth in PoW: Hash power h_i earns $p(h_i)$ coins which buys $H(p(h_i))$ units of hash power

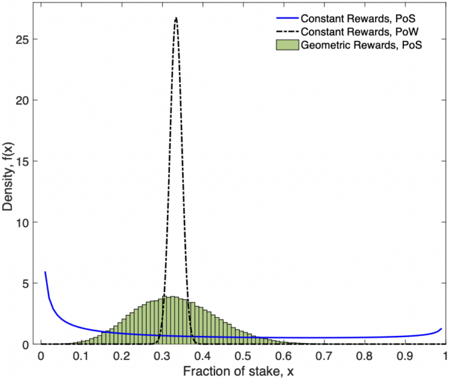
- *Risky process*: H, p are random variables of market prices
- *No Instant Compounding*: Only compound by selling coins for hash power (non-zero latency)
- *Expected Earnings Distribution*: $\text{Binom}(T, h_i / \sum_i h_i)$

How does one compound wealth in PoS?

- *Zero Risk*: Coins earned can immediately be used to increase future rewards (e.g. $H \circ p$ is deterministic)
- *Instant Compounding*: Earned coins can be immediately used to compound wealth
- *Expected Earnings Distribution*: $\text{Beta}(\pi_i, 1 - \pi_i)$

Compounding, compared

Can lead to severe wealth inequality:



Reducing Compounding of Wealth

Simple model of [FKO+19] assumes

- No addition or removal to stake / hash power distribution
- Single leader per block

Define *equitability*: $E_i(T) = \frac{\text{Var}[\pi_i(T)]}{\pi_i(0)(1-\pi_i(0))}$ = variance at time T / variance at time 0

Main result: Only sufficiently non-constant, inflationary block rewards can ensure that $E_i(T) \ll E_i(0)$ as $T \to \infty$

Comparison of Equitable, Inequitable Rewards

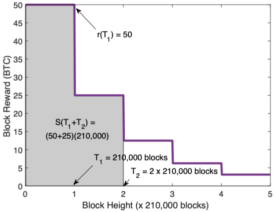


Fig. 2: Bitcoin block rewards as a function of block height. The area of the shaded region gives the total stake after $T_1 + T_2$ time.

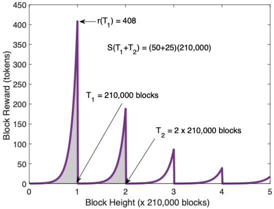


Fig. 3: Geometric block rewards as a function of block height, using Bitcoin-based T_i and R_i values from Figure 2.

Rational Staking Actors

- Most cryptography/DS proofs assume 2 types of agents: honest, Byzantine
- But what about rational agents with complex strategies?
- Suppose there are two coin yields, $\gamma_1(k), \gamma_2(k) \in \mathbf{R}_+$
  - $\gamma_1(k)$ is the yield for staking, $\gamma_2(k)$ is from on-chain lending at block height $k \in \mathbf{N}$
- How does a rational agent allocate their coins?

Modeling Rational Stakers

Rational Agent i state at block height k

- *Resource Distribution*: $\pi_i(k) \in [0,1]$
- *Wealth*: $W_i(k) \in \mathbf{R}_+$

Model of [Chi21] assumes each agent is *Markowitz*, *e.g.* updates their allocation by solving the convex program

$$\pi_i(k+1) = \operatorname*{argmin}_{\pi} \boldsymbol{\pi}(k)^T \boldsymbol{\gamma} + \boldsymbol{\pi}(k)^T \boldsymbol{\Sigma} \boldsymbol{\pi}(k)$$

where

- $\boldsymbol{\pi}(k) = [\pi(k), 1 - \pi(k)]$
- $\boldsymbol{\gamma} = [\gamma_1, \gamma_2]$
- $\boldsymbol{\Sigma} \in \mathbf{R}^{2\times 2}$ is a PSD covariance matrix

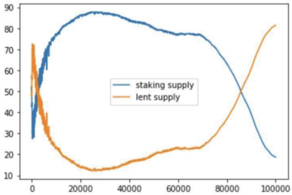
Competitive Equilibria Between Staking and Lending

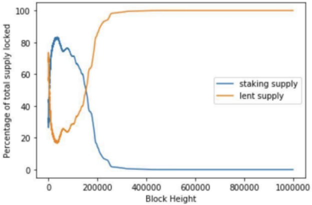
Main Results:

1. Unless the inflation is increasing exponentially, $\lim_{k\to\infty} \pi(k) = 0$ a.s.

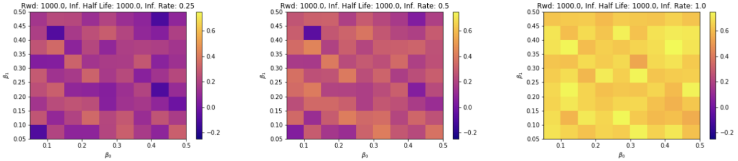
2. Galton-Watson phase transition between $\lim_{k\to\infty} \pi(k) \in \{0, 1\}$, $\pi(k) \to c \in (1/4, 3/4)$ as a function of lending demand distribution moments

Simulation of Galton-Watson Phase Transition





- Left: Oscillatory behavior in relative percentage of supply in stake (blue) and lending (orange)
- Right: 0-1 law where everything ends up lent



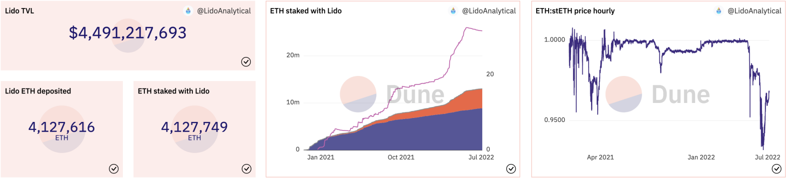
- Heatmap of Coins Staked - Coins Lent (negative = blue/red, positive = yellow)
- Inflation schedule is $R_h \propto e^{\lambda h}$
- Blue heatmaps have $\lambda < 1$, Yellow heatmap has $\lambda \geq 1$ (phase transition)

Outline

Staking Derivatives

- Proof of Stake can be *capital inefficient* for stakers
  - Network only secure if capital locked for a long time
- *Idea* (Manian, Aggarwal, et. al): What if we did overcollateralized lending against stake?
  - *e.g.* I lock $1,000,000 of staked assets, network lets me borrow $200,000 against it
  - Protocol can execute its own liquidations and manage liquidity in a CFMM
  - Similar to a 'perpetual' mortage-backed security
- Clearly reduces the security of the network — but by how much?

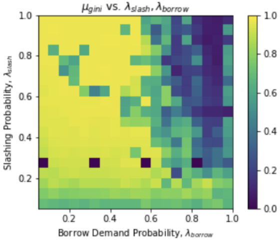
Staking Derivatives Today

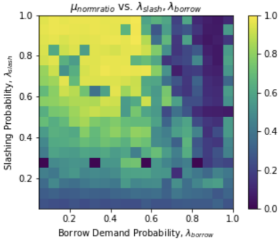


- Largest staking derivative today is Lido stETH
- Borrowing against *locked* ETH2 stake
  - Will only be unlocked once the ETH2 merge occurs
- Deposit ETH, receive stETH (which you can use in DeFi)
- stETH/ETH tends to stay near 1, although recently had a liquidity crisis!

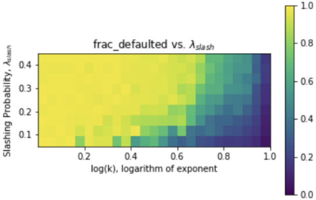
Main Results

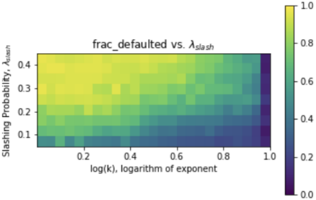
- Lower concentration of wealth in systems with staking derivatives
- Protocol controlled parameters (e.g. margin requirements, liquidation thresholds) can be adjusted dynamically to avoid ruin scenarios
- Qualitatively different phase transition that staking and lending (measure-valued Pólya urn process)





- Expected Gini Coefficient (left) and L^1 to L^2 norm ratio
- Phase transition — Gini coefficient goes down (higher equality) when there's enough borrow demand
- Can show formally this is always *less* than the (expected) Gini coefficient for staking and lending





- x-axis is a notion of *curvature* of the CFMM used for liquidations
- More aggressive price impact (*e.g.* $\log k \approx 1$) has no defaults — trade-off risk vs. return by tuning the CFMM

References I

- Dan Boneh, Saba Eskandarian, Lucjan Hanzlik, and Nicola Greco, *Single secret leader election*, Proceedings of the 2nd ACM Conference on Advances in Financial Technologies, 2020, pp. 12–24.
- Tarun Chitra, *Competitive Equilibria Between Staking and On-chain Lending*, Cryptoeconomic Systems **0** (2021), no. 1, https://cryptoeconomicsystems.pubpub.org/pub/chitra-staking-lending-equilibria.
- Giulia Fanti, Leonid Kogan, Sewoong Oh, Kathleen Ruan, Pramod Viswanath, and Gerui Wang, *Compounding of wealth in proof-of-stake cryptocurrencies*, International conference on financial cryptography and data security, Springer, 2019, pp. 42–61.

References II

Andrew Lewis-Pye and Tim Roughgarden, *Resource pools and the cap theorem*, arXiv preprint arXiv:2006.10698 (2020).