Prof. Lorenzo Naranjo Spring 2026

Proof of Stake Economics

Why Proof of Stake?

Main idea:

- PoW security is mainly a flow-cost mechanism.
- PoS security is mainly a slashable-capital (stock-cost) mechanism.
- Both systems are incentive systems designed to make attacks privately unprofitable.

Logic of the notebook:

1. Show why blockchain immutability requires hash linking, fork choice, and slashing-backed finality.
2. Link rewards and required returns to equilibrium total stake.
3. Translate stake into an attack-deterrence inequality.
4. Compare PoW vs PoS by the type of cost an attacker must bear.

The Blockchain Structure

PoS immutability rests on three complementary layers:

Hash linking. Each block header commits to the previous block's hash. Any alteration to a past block immediately changes its hash and breaks all forward references — detectable without any consensus mechanism.

Fork choice. When competing forks exist, PoS nodes follow the branch with greatest accumulated attestation weight. An attacker controlling stake fraction q can only displace the canonical chain if $q > \frac{1}{2}$; below that threshold the honest chain accumulates weight faster in every slot.

Finality. Once a checkpoint block attracts attestations from at least $\frac{2}{3}$ of total stake, it is finalized and cannot be reverted. Reverting a finalized checkpoint requires at least $\phi \cdot \frac{1}{3} \cdot S$ of capital to equivocate and be slashed.

```
import hashlib
import json
import numpy as np
import pandas as pd

def block_hash(prev_hash: str, slot: int, proposer: str, transactions: list) -> str:
    header = json.dumps(
        {"prev_hash": prev_hash, "slot": slot, "proposer": proposer, "transactions": transact
        sort_keys=True,
    )
    return hashlib.sha256(header.encode()).hexdigest()

genesis_hash = "0" * 64
chain = [{"slot": 0, "proposer": "genesis", "transactions": [], "hash": genesis_hash}]

for slot, proposer, txs in [
    (1, "A", ["tx1", "tx2"]),
    (2, "C", ["tx3"]),
    (3, "B", ["tx4", "tx5"]),
]:
    h = block_hash(chain[-1]["hash"], slot, proposer, txs)
    chain.append({"slot": slot, "proposer": proposer, "transactions": txs, "hash": h})

pd.DataFrame(chain)[["slot", "proposer", "hash"]].assign(
    hash=lambda df: df["hash"].str[:20] + "..."
)
```

	slot	proposer	hash
0	0	genesis	0000000000000000000...
1	1	A	ae1db86e5248d0e66482...
2	2	C	a31ebda99474b5436770...
3	3	B	83d4284091036a0bc353...

Tamper test:

	description	first_20_chars	matches_block2
0	block 1 original hash	ae1db86e5248d0e66482…	True
1	block 1 tampered hash	365d9e1cd52e8a77d2be…	False
2	block 2 prev_hash pointer	ae1db86e5248d0e66482…	True

Key result:

- Changing block 1's transactions produces a completely different hash, instantly breaking the pointer stored in block 2.

Staking Economics and Security as a Stock Cost

The mechanism is: higher net return attracts stake, higher stake raises slashable collateral, and higher slashable collateral raises the cost of attack.

If rewards R are allocated proportionally to stake, the common gross reward rate is:

$$r_i \equiv \frac{R_i}{s_i} \approx \frac{R}{S}.$$

Net staking yield subtracts average costs δ:

$$y \approx \frac{R}{S} - \delta.$$

	total_staked	net_yield
0	100000000.0	0.110
1	150000000.0	0.070
2	250000000.0	0.038
3	400000000.0	0.020

A key insight in Saleh (2021) is that PoS security depends on a stock of slashable capital, not only on a current flow of spending.

In equilibrium, entry drives yield to the required return, so:

$$S \approx \frac{R}{\rho + \delta}.$$

The Blockchain Structure section established that breaking finality requires controlling at least $\alpha = \frac{1}{3}$ of stake. Combining with equilibrium S gives the full deterrence condition:

$$G < \phi\alpha S = \phi\alpha \frac{R}{\rho + \delta}.$$

	slash_fraction_phi	max_attack_gain_G
0	0.10	5.714286e+06
1	0.25	1.428571e+07
2	0.40	2.285714e+07
3	0.60	3.428571e+07
4	0.80	4.571429e+07

Interpretation:

- Rewards R and required return $(\rho + \delta)$ determine equilibrium stake S.
- Security increases with slashable stake S, slash fraction ϕ, and control threshold α.
- The deterrence condition says attacks are unattractive when private gain is below expected slashable loss.

PoW vs. PoS Cost Type

John et al. (2025) emphasizes the economic difference between flow-based and stock-based security. Using $\phi = 0.40$ as a mid-range slash fraction, the compact cost comparison is:

$$C_{\text{PoW}}(H) \approx c_{\text{flow}} H, \qquad C_{\text{PoS}} \approx \phi\alpha S,$$

where H is attack horizon. PoW cost grows with the duration of the attack; PoS cost is a fixed balance-sheet exposure independent of time.

	horizon_H	pow_attack_cost	pos_attack_cost
0	1	2000000	2.285714e+07
1	2	4000000	2.285714e+07
2	4	8000000	2.285714e+07
3	8	16000000	2.285714e+07
4	16	32000000	2.285714e+07

This distinction helps explain why policy levers differ. In PoW, security is tightly linked to sustainable reward flow and operating margins. In PoS, security is tightly linked to stake value, slashability, and credible finality enforcement.

Takeaways

- PoS uses economic stake and slashing instead of computational burn.
- Immutability rests on three layers: hash linking, fork choice, and slashing-backed finality.
- Security is best viewed through incentive constraints, not just protocol syntax.
- Relative to PoW, PoS shifts security from flow costs to stock costs.
- Concentration remains the main systemic risk in either mechanism.

John, Kose, Thomas J. Rivera, and Fahad Saleh. 2025. “Proof-of-Work Versus Proof-of-Stake: A Comparative Economic Analysis.” *Review of Financial Studies* 38 (3): 861–907.

Saleh, Fahad. 2021. “Blockchain Without Waste: Proof-of-Stake.” *Review of Financial Studies* 34 (3): 1156–90.