# Analysing Algorand: Possible directions for quantitative analysis

Ivan Malakhov

PRIN NiRvAna project meeting 2024

# What is Algorand?

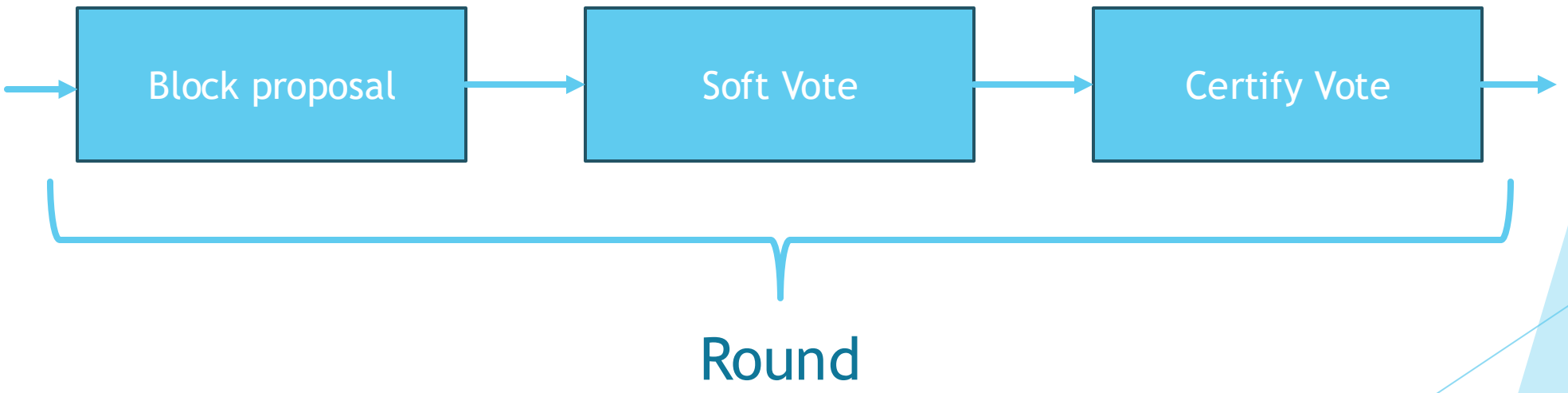
Key features:

- **Pure PoS consensus**
- VRF
- Unconditional reward
- Forkless
- Instant finality
- Smart contracts
- Delegation
- FIFO Mempool policy

# Verifiable Random Function (VRF)

- Core mechanism in Algorand's PPoS
- Eliptic curve
- Cryptographic sortition
- Determines randomly if a token is the "winner"

# Pure Proof-of-Stake



# Pure PoS

- Scalability
  - Superfast execution of VRF for the committee
  - The committee size is apprx. 1000 members
- Security
  - Introduction of primary and ephemeral keys
  - VRF randomness
- Decentralisation
  - Total involvement of participants

# Forks absence

- One block to pass the threshold of committee votes
- $10^{-18}$ probability of fork
- All appeared blocks are final (consequence)

# Smart Contracts

- Transaction Execution Approval Language (TEAL)
- Assembly-like language
- Struggle to compete with other SC languages

# Algorand incentivisation

Standard implementation:

- All addresses contain a minimum balance or more will receive rewards
- Determine the *Reward Pool* based on number of blocks in *Reward Period* and fixed per block reward
- Split the per block reward amount across all token holders based on the amount of stake
- Block reward depends on effective balance

# Algorand incentivisation

## Algorand Consensus Incentivisation

AN ALGORAND FOUNDATION DISCUSSION PAPER

John Woods
john@algorand.foundation
john@postquantum.dev

Michele Treccani
michele@algorand.foundation

John Jannotti
jj@algorand.com

Naveed Ihsanullah
naveed@algorand.foundation
naveed@jamsni.com

Version 1.0, 14th December 2023

### 1 Purpose

The goal of this project is to engineer a native solution at layer 1, modifying the Algorand proto-

# Algorand incentivisation

Proposal of Deflating block reward mechanism

$$R(\eta) = (1 - \frac{\eta}{\mathcal{N}})$$

**then, adding a normalisation factor:**

$$\mathcal{K} = \sum_{\eta=1}^{\mathcal{N}} R(\eta) = (\mathcal{N} - \frac{\mathcal{N}(\mathcal{N}+1)}{2 \times \mathcal{N}})$$

**∋ the payout function is defined as:**

$$\mathcal{R}(\eta) = \mathcal{M} \times R(\eta) \times \frac{1}{\mathcal{K}}$$

linear

$$R'(\eta) = e^{-\rho \frac{\eta}{\mathcal{N}}}$$

**then, adding a normalisation factor:**

$$\mathcal{K} = \sum_{\eta=1}^{\mathcal{N}} R'(\eta) = \frac{1 - e^{-\rho}}{1 - e^{-\frac{\rho}{\mathcal{N}}}}$$

**∋ the payout function is defined as:**

$$\mathcal{R}'(\eta) = \mathcal{M} \times e^{-\rho \frac{\eta}{\mathcal{N}}} \frac{1}{\mathcal{K}}$$

exponential

Where $\eta$ : number of a starting block

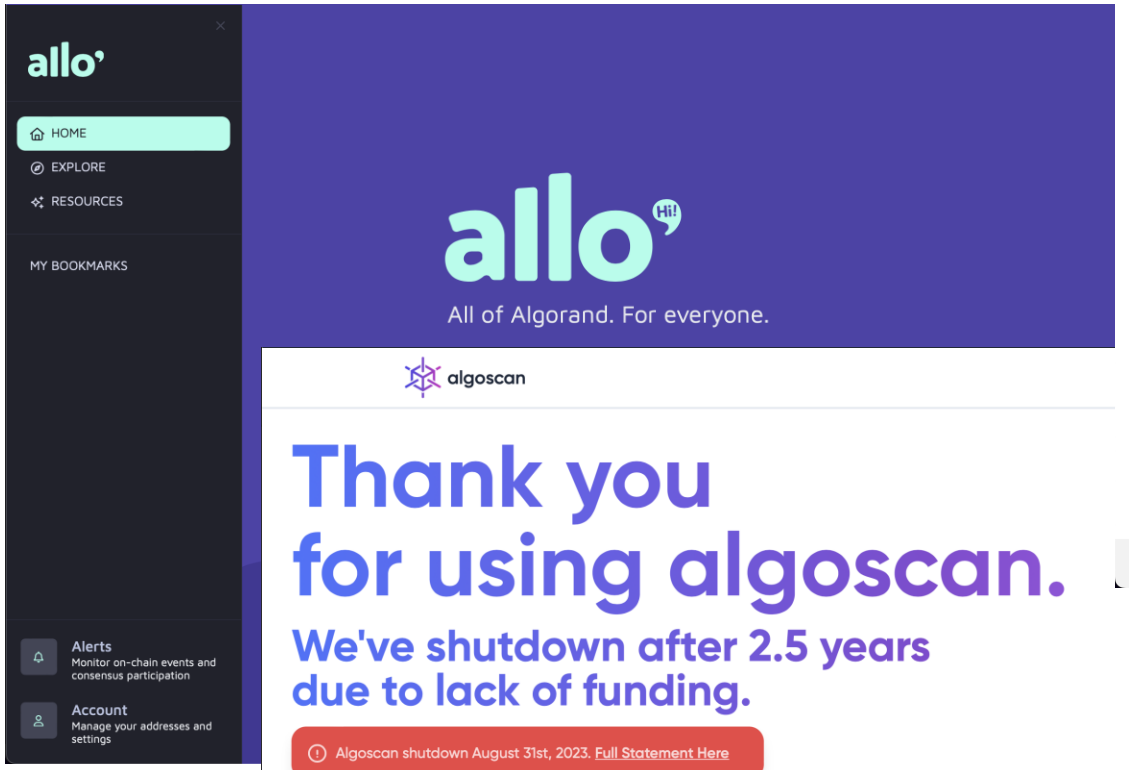
- M : Total units for incentive (e.g., 200 million Algo)
- B : Block-time (e.g., 3 seconds)
- T : Total duration for the payout, expressed in seconds (e.g., 3 years)
- N : Total number of blocks over the payout period
- $\rho$ : Rate of decay

# Algorand vulnerabilities

Attack vectors and their potential mitigations:

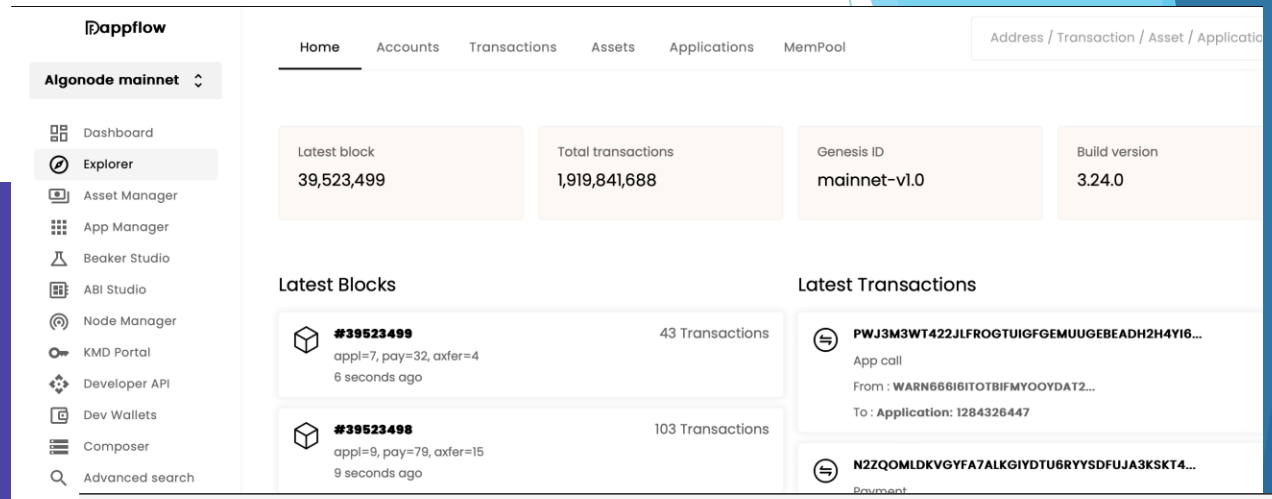
- Absenteeism -> Suspension from the stake
- Pooling -> Participation Key expiration
- Protocol deviation -> “negligible”?

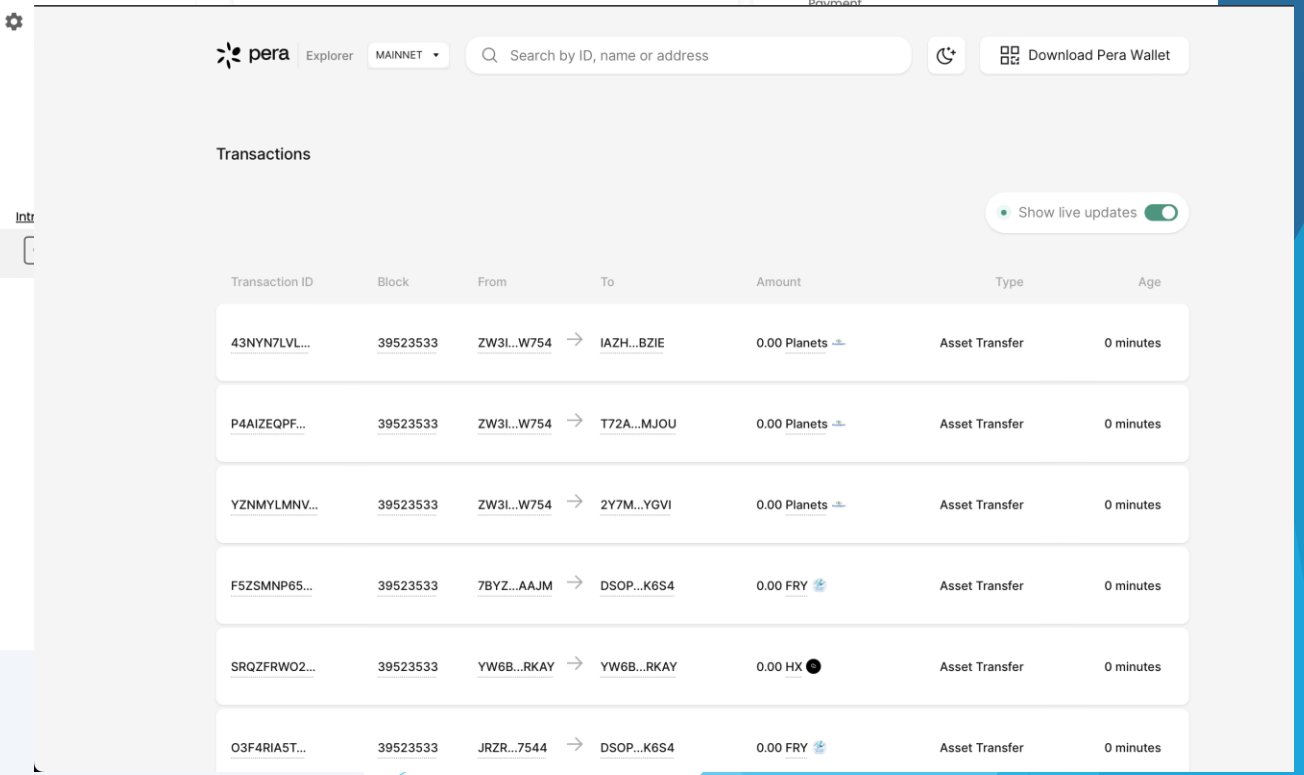
# Algorand analysis









# Conclusion

Promising PoS-based blockchain with solid academic background

- Lacks substantial incentivisation mechanism
- Validators' absenteeism
- Pooling and delegation
- Protocol Deviation