CoinCeeper: A Comprehensive Proof-of-Network Framework for Decentralized Physical Infrastructure

Mohammad Nazarnezhad[1]

[1]Independent Researcher in Distributed Systems and Blockchain Architecture

Abstract

Decentralized Physical Infrastructure Networks (DePINs) face fundamental challenges in achieving deterministic consensus while accurately verifying heterogeneous, geographically distributed network resources. Traditional consensus mechanisms lack native bandwidth validation, whereas naive network-based approaches suffer from Sybil vulnerabilities, non-determinism, NAT traversal issues, and legal liabilities. This paper introduces **CoinCeeper**, a comprehensive *Proof-of-Network* (PoN) framework. We mathematically formalize geographic scarcity, hardware provenance, and statistical bandwidth verification into a deterministic consensus weight function. Furthermore, we introduce an Asymmetric Peer Pairing architecture to solve NAT/CGNAT traversal, a compliance-aware relay architecture to reduce legal exposure, and a 3-tier slashing governance model. Finally, we propose an Account Abstraction integration via Substrate/Cosmos AppChains to enable gasless transactions. Numerical simulations and an executable roadmap prove that CoinCeeper achieves robust consensus convergence, mitigates collusion, and maintains a deflationary token supply under realistic DePIN utilization scenarios.

Keywords: Proof-of-Network, DePIN, Consensus Mechanism, Geographic Scarcity, Tokenomics, NAT Traversal, Account Abstraction.

---

1 Introduction

The proliferation of Decentralized Physical Infrastructure Networks (DePINs) has exposed critical limitations in existing blockchain consensus models. Proof-of-Work (PoW) and Proof-of-Stake (PoS) secure ledger state transitions but cannot natively verify physical network resources such as bandwidth, latency, or geographic coverage. Conversely, naive network-based consensus schemes suffer from non-deterministic behavior due to internet jitter, vulnerability to Sybil attacks via cloud server leasing, and prohibitive overhead from continuous traffic probing.

To address these gaps, we propose **CoinCeeper**, a hybrid consensus architecture where a deterministic base layer secures block finality, while a mathematically rigorous *Proof-of-Network* (PoN) layer acts as the verification and incentive engine. This paper makes the following comprehensive contributions:

- A formal mathematical model of PoN consensus weight enforcing geographic scarcity.
- An Asymmetric Peer Pairing architecture to resolve NAT/CGNAT traversal for residential nodes.
- A Blind Traffic Matrix protocol utilizing dummy encrypted data to absolve nodes of legal liability.
- A 3-tier Slashing and Governance engine to penalize downtime, poor quality, and fraud.
- An Account Abstraction (Gasless) integration strategy for Substrate/Cosmos AppChains.

2 System Architecture and Network Topology

2.1 Asymmetric Peer Pairing (NAT/CGNAT Resolution)

Over 80% of residential users operate behind Carrier-Grade NAT (CGNAT), lacking static public IPs and open ports, which prevents direct P2P validation. CoinCeeper introduces an *Asymmetric Peer Pairing* topology:

- **Provers (Residential Nodes):** Generate bandwidth proofs and micro-pings from behind NAT.
- **Relay Nodes (Public/Datacenter):** Act as stateless facilitators that route UDP micro-pings without inspecting payloads, ensuring Provers remain verifiable without violating firewall rules or requiring port-forwarding.

2.2 Blind Traffic Matrix (Legal Liability Shield)

Routing third-party traffic exposes node operators to legal liability for illicit content. CoinCeeper employs a *Blind Traffic Matrix*. Micro-pings consist exclusively of *Dummy Encrypted Data* (D_{enc}), generated via sharded cryptographic hashes:

$$D_{enc} = \text{AES}_k(\text{SHA-256}(\text{Nonce}||\text{Timestamp})) \tag{1}$$

Since D_{enc} possesses zero semantic value and cannot be decrypted without the ephemeral session key, relay operators do not inspect payload contents. This does not create absolute legal immunity; instead, it reduces legal exposure when combined with the compliance architecture introduced in PoN v2.0.

2.3 Zero-Overhead Micro-Pings (Hardware Preservation)

Continuous large-file downloads degrade NIC hardware and exhaust ISP data caps. The initial prototype used $N = 50$ UDP packets of $L_{pkt} = 1$ KB each as a conservative lower bound for test traffic:

$$\text{Prototype Overhead} = N \times L_{pkt} = 50 \text{ KB} \approx 0.00005 \text{ GB}. \tag{2}$$

However, fixed-size sampling is insufficient for heterogeneous real-world networks. PoN v2.0 therefore replaces this prototype rule with confidence-bounded adaptive sampling and Kalman filtering.

3 Mathematical Formulation of PoN Consensus

3.1 Consensus Weight Formulation

The consensus weight W_i for node n_i integrates raw bandwidth, geographic scarcity, and hardware provenance through a convergent density penalty:

$$W_i = B_i \cdot \frac{1}{(N_{geo}+1)^{\gamma}} \cdot e^{-\alpha D_i}, \quad \gamma \in (1,2], \tag{3}$$

where B_i is filtered bandwidth, N_{geo} is the active node count in the region, $D_i \in \{0,1\}$ indicates datacenter hosting, $\alpha > 0$ is a datacenter penalty coefficient, and $\gamma > 1$ ensures that cumulative Sybil rewards are bounded. The exponential datacenter penalty makes dense cloud deployments economically less attractive without making residential participation unprofitable.

3.2 Deterministic Bandwidth Estimation

To resolve EVM non-determinism caused by internet jitter, raw micro-ping intervals $\{\Delta t_k\}$ are processed via an Exponential Moving Average (EMA):

$$\overline{\Delta t}_t = \lambda \Delta t_t + (1-\lambda)\overline{\Delta t}_{t-1}, \quad \lambda \in (0,1) \tag{4}$$

Filtered bandwidth is computed as $B_i = \frac{8 \cdot L_{pkt}}{\overline{\Delta t}_t \cdot 10^6}$ (Mbps). The EMA guarantees Lipschitz continuity, ensuring bounded state transitions suitable for deterministic smart contract execution.

4 Security, Privacy, and Legal Analysis

4.1 Sybil and Collusion Resistance

An attacker deploying k fake nodes in a single region incurs linear infrastructure costs $\mathcal{C}_{syb}(k) = k \cdot c_{hw}$. A logarithmic reward penalty is insufficient because $\sum_j 1/(1+\ln(N+j))$ diverges. CoinCeeper therefore uses the convergent fractional-power penalty in Eq. 3:

$$R_{syb}(k) = \sum_{j=1}^{k} \frac{1}{(N+j)^{\gamma}}, \quad \gamma > 1. \tag{5}$$

The infinite reward tail is bounded by the integral test:

$$\sum_{j=1}^{\infty} \frac{1}{(N+j)^{\gamma}} \leq \int_{N}^{\infty} x^{-\gamma} dx = \frac{N^{1-\gamma}}{\gamma - 1} < \infty. \tag{6}$$

Thus, there exists k_0 such that for all $k > k_0$, $\mathcal{C}_{syb}(k) > R_{syb}(k)$, making large-scale Sybil expansion economically irrational under positive hardware and location costs.

4.2 DDoS and Fake Traffic Mitigation

Fake traffic injection is neutralized because rewards depend on U_{market} (real client demand), not synthetic volume. Furthermore, Verifiable Random Functions (VRF) ensure unpredictable validator slot assignment, preventing targeted DDoS attacks on specific nodes.

5 Tokenomics, Governance, and Slashing

5.1 Dynamic Burn-Mint Equilibrium

The epochal reward R_i scales with market utilization $U_{market} \in [0, 1]$:

$$R_i(t) = W_i(t) \cdot U_{market}(t) \cdot \phi \tag{7}$$

where $\phi > 0$ is the global reward scaling coefficient, distinct from the density exponent $\gamma \in (1, 2]$ used in the consensus weight function. Simultaneously, client bandwidth consumption triggers a burn mechanism $C(t) = \beta \cdot V_{used}(t)$. When network utilization exceeds a critical threshold, the burn rate surpasses minting, guaranteeing long-term deflation.

5.2 Core Symbol Definitions

To avoid ambiguity between the density exponent and token reward coefficient, the core control parameters are defined in Table 1.

Table 1: Core Mathematical Symbols

Symbol	**Definition**	**Range**
γ	Regional density exponent	$(1, 2]$
α	Datacenter penalty coefficient	> 0
ϕ	Global reward scaling coefficient	> 0
β	Burn rate coefficient	$(0, 1]$
β_D	Datacenter-density penalty coefficient	> 0
η	Supply controller learning rate	$(0, 0.1)$

5.3 Transparent Token Allocation and Vesting Schedule

To ensure long-term sustainability, prevent sudden sell pressure, and provide complete transparency for investors, the PON token distribution is designed around an anti-inflationary model detailed in Table 2.

Table 2: Token Allocation and Vesting Schedule

Allocation Sector	**Share**	**Vesting Schedule**	**Strategic Goal**
Ecosystem & Node Rewards	40%	Linear over 5 years	Long-term DePIN incentives
Treasury & Governance (DAO)	20%	1-yr cliff, then DAO	Dev funding & proposals
Team & Advisors	15%	1-yr cliff, 3-yr linear	Align team with network
Public & Private Sale	15%	10% TGE, 18-mo linear	Capital & initial liquidity
Liquidity & Market Making	10%	Unlocked at TGE	Exchange price stability

5.4 3-Tier Slashing Engine

To maintain network integrity, CoinCeeper enforces a proportional governance model:

- **Level 1 (Downtime):** Reputation decay and temporary reward suspension (no stake loss).
- **Level 2 (Poor Quality/High Jitter):** 5% slashing of staked tokens for sustained packet loss.
- **Level 3 (Fraud/Sybil):** 100% slashing, blacklisting, and transfer of seized assets to the Treasury pool.

6 AppChain Integration and Gasless Abstraction

To eliminate the UX barrier of native gas tokens, CoinCeeper is designed for deployment on Substrate (using `pallet-evm` and custom `FeePayment` traits) or Cosmos (Ethermint). This enables *Account Abstraction* (ERC-4337), allowing dApps to sponsor gas fees or users to pay transaction costs directly in the native PON token, creating a seamless Web2-like experience for end-users.

7 Proof-of-Network v2.0: Hardening Against Practical Attacks

The initial PoN framework is upgraded to **PoN v2.0** by replacing self-reported assumptions with decentralized verification, robust statistics, graph-theoretic Sybil resistance, bootstrap trust roots, legally auditable relay design, and stable monetary control. This section formalizes the main engineering corrections required for an industrial-grade DePIN protocol.

7.1 Convergent Sybil Penalty and Corrected Weight Function

The first correction is mathematical. A logarithmic scarcity term does not bound cumulative Sybil rewards because the series $\sum_j 1/(1 + \ln(N + j))$ diverges. PoN v2.0 therefore uses a fractional-power regional density penalty:

$$W_i = B_i \cdot \frac{1}{(N_{geo} + 1)^{\gamma}} \cdot e^{-\alpha D_i}, \quad \gamma \in (1, 2]. \tag{8}$$

For an attacker adding k identities to a region, cumulative Sybil reward is bounded by:

$$R_{sybil}(k) = \sum_{j=1}^{k} \frac{1}{(N + j)^{\gamma}}, \quad \gamma > 1, \tag{9}$$

with

$$\sum_{j=1}^{\infty} \frac{1}{(N + j)^{\gamma}} \leq \int_{N}^{\infty} x^{-\gamma} dx = \frac{N^{1-\gamma}}{\gamma - 1} < \infty. \tag{10}$$

Because the attack cost $C_{syb}(k) = k \cdot c_{hw}$ grows linearly while reward is bounded, there exists k_0 such that for all $k > k_0$, $C_{syb}(k) > R_{sybil}(k)$. This replaces the earlier non-convergent logarithmic claim with a provable bounded-reward model.

A stricter datacenter-sensitive alternative can be enabled by governance when cloud concentration is detected:

$$W_i = \frac{B_i}{(N_{geo} + 1)^{\gamma} \cdot (1 + \beta_D D_i N_{geo})}. \tag{11}$$

Here, $\beta_D > 0$ is a datacenter-density penalty coefficient. This optional form penalizes datacenter nodes more heavily as regional datacenter density increases.

7.2 Oracle Bootstrap and Decentralized Metric Verification

PoN v2.0 resolves both the oracle problem and the bootstrap problem through a three-layer verifier pipeline. First, genesis validators are admitted through hardware-backed attestation. In the initial network phase, a validator's TPM 2.0 root of trust generates a key pair (pk_{tpm}, sk_{tpm}) and signs a hardware fingerprint:

$$attestation = \text{Sign}_{sk_{tpm}}\left(\text{SHA256}(MAC \parallel CPU_{serial} \parallel timestamp)\right). \tag{12}$$

Genesis Trust Bootstrapping. A circular dependency arises if the first validators must be approved by validators that do not yet exist. CoinCeeper resolves this through a *Genesis Multi-Signature Council* operating in three phases.

Phase 0 – Foundation Council. Before any on-chain activity, a set of $m = 7$ geographically distributed founding organizations $\mathcal{F} = \{F_1, \ldots, F_7\}$, such as academic institutions, security auditors, and independent researchers, hold a (5-of-7) multi-signature key:

$$\text{GenesisApproval}(n_i) = \text{MultiSig}_{5/7}\left(\left\{\text{Sign}_{F_k}(attestation_i)\right\}_{k=1}^{7}\right). \tag{13}$$

A node n_i is admitted as a genesis validator only if at least five of the seven council members co-sign its TPM attestation, eliminating the single point of failure of a single trusted party.

Phase 1 – Reputation Accumulation. Genesis validators begin accumulating reputation scores $\rho_i(t)$ after launch. After a minimum observation window of T_{obs} epochs, the reputation gate opens to new entrants:

$$\rho_i(T_{obs}) > \rho_{min} \quad \Longrightarrow \quad n_i \in \mathcal{V}_{eligible}. \tag{14}$$

Phase 2 – Council Dissolution. Once the active validator set exceeds a threshold $|\mathcal{V}_{active}| > V_{threshold}$, a DAO governance vote dissolves the Foundation Council:

$$|\mathcal{V}_{active}| > V_{threshold} \quad \Longrightarrow \quad \mathcal{F} \xrightarrow{DAO\ vote} \emptyset. \tag{15}$$

Trust is therefore initially anchored in a transparent, auditable multi-party institution and progressively transferred to the decentralized validator set, eliminating bootstrap circularity without introducing permanent centralization.

On-chain registration checks the TPM signature before granting temporary genesis-validator status. Second, validator eligibility becomes reputation-gated:

$$\rho_i(t) = \mu\rho_i(t-1) + (1-\mu)\,\text{ConsistencyScore}_i(t), \quad \rho_i \in [0,1]. \tag{16}$$

The consistency score measures agreement between self-reported and peer-measured bandwidth:

$$\text{ConsistencyScore}_i(t) = 1 - \frac{\left|\hat{B}_i^{self}(t) - \hat{B}_i^{peer}(t)\right|}{\hat{B}_i^{peer}(t)} \in [0,1]. \tag{17}$$

Only nodes with $\rho_i > \rho_{min}$ can enter the randomized validator pool. Third, validators must post an economic bond:

$$Stake_{validator} \geq \lambda \cdot \mathbb{E}[R_{epoch}] \cdot T_{lock}, \tag{18}$$

which is slashable when measurements are falsified or collusion is detected.

For node i, the active validator set is sampled by a VRF:

$$\mathcal{V}_i = \text{VRF}(\text{Seed}_{epoch} \parallel i). \tag{19}$$

The accepted bandwidth estimate is the robust median of peer measurements:

$$\hat{B}_i = \text{median}_{j \in \mathcal{V}_i} \left(\frac{8 \cdot L_{pkt}}{\Delta t_{ij}} \right). \tag{20}$$

The median resists outliers and small colluding minorities, while reputation and bonding prevent a circular dependency on initially unknown honest validators.

7.3 Multi-Modal Location Verification

PoN v2.0 does not rely on raw GPS alone, because GPS can be spoofed and TEEs are uncommon on residential hardware. Regional density is computed over an H3-style grid:

$$N_{geo}(h) = \sum_{n \in Grid_h} \mathbb{I}\left(Proof_{loc}(n)\ valid\right). \tag{21}$$

Location validity combines three complementary signals. First, Wi-Fi triangulation uses a zero-knowledge proof over nearby access points:

$$loc_{hash} = \text{SHA256}(sort(\{BSSID_i \parallel RSSI_i\}_{i=1}^{m})). \tag{22}$$

The node proves knowledge of BSSIDs whose hash equals loc_{hash}, whose public database coordinates lie inside region h, and whose geometric median is consistent with the claimed grid cell. Second, latency fingerprinting triangulates the node against geographically distributed beacon nodes with dimensionally consistent units. Let $RTT(x, b)$ denote measured round-trip time in milliseconds and $d(x, b)$ the geographic distance in kilometers between candidate location x and beacon b. The effective propagation speed accounts for fiber refractive index and routing

overhead:

$$c_{eff} = \frac{c}{n_{fiber} \cdot r_{routing}} \approx \frac{3 \times 10^8}{1.5 \times 1.4} \approx 1.43 \times 10^8 \text{ m/s} = 143 \text{ km/ms}. \tag{23}$$

Here, $n_{fiber} \approx 1.5$ and $r_{routing} \approx 1.4$. The location estimate is then:

$$loc_{estimate} = \arg\min_{x} \sum_{b \in Beacons} \left(\text{RTT}(x, b) - \frac{2\, d(x, b)}{c_{\text{eff}}} \right)^2. \tag{24}$$

The factor of 2 accounts for round-trip propagation, and both terms inside the objective are measured in milliseconds. Third, social proximity decays over time and is weighted by neighbor reputation:

$$LocScore_i = \sum_{j \in Neighbors_i} \rho_j \cdot \mathbb{I}[RTT(i,j) < \tau_{geo}] \cdot e^{-\lambda t}. \tag{25}$$

A location claim is accepted only when the combined score from Wi-Fi, beacon latency, and social proximity exceeds a DAO-defined threshold.

7.4 Demand-Coupled Scarcity and Region-Hopping Resistance

Low node count alone must not produce high rewards, because an attacker could spread fake nodes across low-density regions. PoN v2.0 couples scarcity with verified regional demand and graph dispersion:

$$W_i = B_i \cdot \frac{U_{geo}^{verified}}{(N_{geo} + 1)^{\gamma}} \cdot e^{-\alpha D_i} \cdot (1 - \rho_{clust}). \tag{26}$$

Here, $U_{geo}^{verified} \in [0, 1]$ is the ratio of client-paid traffic in region h. To keep the clustering penalty deterministic and on-chain computable, PoN v2.0 avoids Louvain-style graph clustering and instead defines:

$$\rho_{clust} = \frac{|\{(i,j) : i, j \in \mathcal{V},\ RTT(i,j) < \tau_{local}\}|}{|\mathcal{V}|(|\mathcal{V}| - 1)/2}. \tag{27}$$

This ratio measures the share of validator pairs that are suspiciously close under the local-RTT threshold τ_{local}. It is deterministic, inexpensive to compute from submitted RTT attestations, and suppresses rewards when validators form an artificial local cluster.

7.5 Common-Carrier Relay and Compliance Layer

PoN v2.0 replaces claims of full legal immunity with the narrower and more defensible target of legal risk reduction comparable to neutral relay networks and CDNs. Traffic is encrypted, split into shards, routed over independent relay paths, and reassembled only at the destination; no single relay sees the complete payload. Before routing, a compliance oracle checks destination and content-derived hashes:

```
async function routePacket(packet) {
  const routeHash = sha256(packet.destinationIdentifier);
  const isBlacklisted = await complianceOracle.check(routeHash);
  if (isBlacklisted) {
    slash(packet.sourceNode, SlashLevel.FRAUD);
    return null;
```

```
  }
  return forwardBlind(packet);
}
```

The legal-technical layer includes CSAM hash filtering through recognized databases such as PhotoDNA/NCMEC where legally available, annual transparency reports similar to neutral relay projects, and jurisdiction-specific legal opinion letters before launch in the EU and US. The cryptographic layer adds key inaccessibility and non-storage proofs. Session keys are AES-128 ephemeral keys; after secure erasure, recovery probability is bounded by the AES-128 brute-force bound consistent with NIST SP 800-57 security guidance [11]:

$$k_t \xrightarrow{\textit{secure erase}} \emptyset, \quad \Pr[Recover(k_t)] \leq 2^{-128}, \tag{28}$$

$$\pi_{log} = \text{zkSTARK}(Memory_{node} \equiv Empty). \tag{29}$$

This architecture does not promise immunity; it provides auditable evidence that relays act as neutral, non-custodial, non-inspecting infrastructure.

7.6 Adaptive Sampling and Kalman Filtering

The fixed 50-packet prototype is replaced with confidence-bounded adaptive sampling. Measurement continues until relative latency dispersion falls below a target threshold:

$$\frac{\sigma(\Delta t)}{\mu(\Delta t)} \leq \theta, \quad \theta \approx 0.05. \tag{30}$$

Final latency estimation uses a discrete Kalman filter:

$$\hat{\Delta t}_k^- = \hat{\Delta t}_{k-1}, \qquad P_k^- = P_{k-1} + Q, \tag{31}$$

$$K_k = \frac{P_k^-}{P_k^- + R}, \qquad \hat{\Delta t}_k = \hat{\Delta t}_k^- + K_k(z_k - \hat{\Delta t}_k^-), \tag{32}$$

$$P_k = (1 - K_k)P_k^-. \tag{33}$$

The final bandwidth estimate is reported with a 95% confidence interval:

$$B_i = \frac{8 \cdot L_{pkt}}{\hat{\Delta t}_k \pm 1.96\sqrt{P_k}}. \tag{34}$$

This removes fixed-sample bias and provides deterministic, confidence-bounded inputs for settlement.

7.7 Stable Supply Controller Instead of Unconstrained PID

A naive PID controller is not guaranteed to be stable in a nonlinear token economy. PoN v2.0 therefore specifies a constrained model predictive controller (MPC) for off-chain simulation and

a simpler Lyapunov-stable update rule for on-chain execution. The MPC objective is:

$$\min_{\phi(t),\beta(t)} \sum_{\tau=t}^{t+H} \left(S(\tau) - S_{target}(\tau)\right)^2, \tag{35}$$

where H is the prediction horizon, recommended as $H \in [10, 50]$ epochs based on token velocity and initialized at $H = 20$ as a DAO-adjustable parameter. The optimization is subject to:

$$\phi_{min} \leq \phi(t) \leq \phi_{max}, \quad \beta_{min} \leq \beta(t) \leq \beta_{max}, \quad |\Delta\phi(t)| \leq \delta_\phi. \tag{36}$$

For deterministic smart contracts, the deployable update is:

$$\phi(t+1) = \phi(t)\left(1 - \eta\frac{S(t) - S_{target}}{S_{target}}\right), \tag{37}$$

$$\beta(t+1) = \beta(t)\left(1 + \eta\frac{S(t) - S_{target}}{S_{target}}\right), \quad \eta \in (0, 0.1). \tag{38}$$

Lemma 7.1 (Supply Controller Stability)**.** *Let $S(t)$ denote the token supply at epoch t, S_{target} the target supply, and S_{max} an upper bound such that $S(t) \leq S_{max}$ for all t. Define the Lyapunov candidate:*

$$V(t) = \big(S(t) - S_{target}\big)^2. \tag{39}$$

Under the update rules

$$\phi(t+1) = \phi(t)\left(1 - \eta\frac{S(t) - S_{target}}{S_{target}}\right), \tag{40}$$

$$\beta(t+1) = \beta(t)\left(1 + \eta\frac{S(t) - S_{target}}{S_{target}}\right), \tag{41}$$

if the learning rate satisfies

$$\boxed{\eta < \frac{S_{target}}{S_{max} - S_{target}}}, \tag{42}$$

then $\mathbb{E}[\Delta V(t)] < 0$ in a neighborhood of S_{target}, and the supply converges asymptotically to S_{target} in expectation.

Proof. Define the normalized error $\varepsilon(t) = \frac{S(t) - S_{target}}{S_{target}}$, so $V(t) = S_{target}^2 \varepsilon(t)^2$. The supply dynamics follow:

$$S(t+1) = S(t) + \phi(t)\sum_i W_i U_{market} - \beta(t) V_{used}(t) P_{PON}. \tag{43}$$

When $S(t) > S_{target}$, the update decreases $\phi(t+1)$, reducing minting, and increases $\beta(t+1)$, increasing burning. When $S(t) < S_{target}$, the signs reverse. From the update equations, when $S(t) > S_{target}$ the mint coefficient is multiplied by $(1 - \eta\varepsilon(t))$ and the burn coefficient by $(1 + \eta\varepsilon(t))$. Hence the corrected net supply flow satisfies:

$$\Delta_S(t) = \phi(t+1) MintFlow - \beta(t+1) BurnFlow \leq -\eta\big(S(t) - S_{target}\big)\frac{S_{target}}{S_{max}} + \sigma. \tag{44}$$

The induced supply error is therefore bounded by:

$$|S(t+1) - S_{target}| \leq |S(t) - S_{target}| \left(1 - \eta \frac{S_{target}}{S_{max}}\right) + \sigma, \tag{45}$$

where σ is bounded stochastic demand noise with $\mathbb{E}[\sigma] = 0$. Therefore:

$$\begin{aligned} \mathbb{E}[\Delta V(t)] &= \mathbb{E}[V(t+1)] - V(t) \\ &\leq V(t) \left(1 - \eta \frac{S_{target}}{S_{max}}\right)^2 - V(t) \\ &= V(t) \left[\left(1 - \eta \frac{S_{target}}{S_{max}}\right)^2 - 1\right]. \end{aligned} \tag{46}$$

For $\mathbb{E}[\Delta V(t)] < 0$, it is sufficient that

$$\left(1 - \eta \frac{S_{target}}{S_{max}}\right)^2 < 1 \quad \Longleftrightarrow \quad 0 < \eta < \frac{2S_{max}}{S_{target}}. \tag{47}$$

The tighter sufficient condition

$$\eta < \frac{S_{target}}{S_{max} - S_{target}} \tag{48}$$

prevents overshoot. Under this condition, $\mathbb{E}[\Delta V(t)] < 0$ for all $S(t) \neq S_{target}$ in the local operating region, establishing asymptotic stability in expectation. □

Numerical example. With $S_{target} = 10^9$ tokens and $S_{max} = 1.5 \times 10^9$ tokens, the stability bound gives:

$$\eta < \frac{10^9}{1.5 \times 10^9 - 10^9} = 2.0. \tag{49}$$

The conservatively chosen value $\eta = 0.05$ satisfies this bound with a 40× margin, ensuring robust convergence under demand volatility.

7.8 Scientific Publication and Validation Roadmap

To strengthen scientific credibility, CoinCeeper replaces internal self-citations with external references where possible and adds an explicit validation roadmap: within 1–2 months, publish an arXiv preprint with DOI metadata and release simulation code; within 3–6 months, submit to IEEE Blockchain, ACM CCS, or a comparable venue and obtain independent mathematical review; within 6–12 months, publish empirical testnet results from real network measurements.

7.9 PoN v2.0 Summary

Table 3 summarizes the upgraded design priorities.

8 Roadmap and Deployment Strategy

Because CoinCeeper has no direct international precedent and combines proof-of-network measurement, NAT-aware validation, compliance-aware relay design, and gasless settlement in a

Table 3: PoN v2.0 Security and Engineering Upgrades

Issue	**PoN v2.0 Correction**	**Priority**
Divergent Sybil proof	Fractional-power convergent penalty with $\gamma > 1$	Immediate
Incorrect AWS table	Recomputed weights using Eq. 8	Immediate
Oracle bootstrap	TPM genesis validators, reputation, and economic bonding	Pre-testnet
GPS spoofing	Wi-Fi ZK proof, latency fingerprinting, and social proximity	Pre-testnet
Legal overclaim	Common-carrier relay plus compliance oracle and legal opinions	Pre-mainnet
Unstable PID	Constrained MPC and Lyapunov-stable ϕ/β updates	Pre-testnet
Weak references	External references, arXiv release, peer review, and empirical results	Long-term

novel architecture, its deployment schedule is intentionally conservative. To mitigate technical risks and guarantee network security, the development of CoinCeeper is structured across four extended operational phases from early 2026 through mid-2028:

- **Phase 1: Foundational Testnet and Local Simulation (Q1–Q2 2026):** Completion of geographic weighting formulas, ZK-proof experiments, tokenomic simulations, and launch of an initial closed testnet for protocol validation.
- **Phase 2: Public Testnet and Large-Scale Validation (Q3 2026–Q2 2027):** Deployment of the Substrate-based AppChain, integration of the `pallet-evm` module, and stress-testing the Asymmetric Peer Pairing protocol with progressively larger groups of volunteer nodes.
- **Phase 3: Security Audits, Governance Hardening, and Market Integration (Q3–Q4 2027):** Comprehensive smart contract and consensus layer audits, launch of a Bug Bounty program, refinement of slashing governance, and controlled integration with early CDN and DePIN marketplace partners.
- **Phase 4: Mainnet Launch and Commercial Scaling (Q1–Q2 2028):** Activation of the Gasless Account Abstraction layer, production deployment of real-yield settlement, and gradual scaling toward enterprise and marketplace adoption by mid-2028.

9 Numerical Evaluation

Table 4 demonstrates the anti-centralization property of the weight function. Using Eq. 8,

Table 4: Corrected Consensus Weight Comparison ($\gamma = 1.5,\ \alpha = 3.0$)

Node Type	B (Mbps)	N_{geo}	D	W (Score)
Residential (Low Density)	80	3	0	**10.00**
Residential (High Density)	80	200	0	**0.028**
AWS Datacenter (Saturated)	10,000	1,500	1	**8.57**

the AWS node no longer receives the previously incorrect score of 1.42. With $\gamma = 1.5$ and $\alpha = 3.0$, the residential node obtains $80/4^{1.5} = 10.00$, a high-density residential node obtains $80/201^{1.5} \approx 0.028$, while the AWS node obtains $10000/1501^{1.5} \cdot e^{-3} \approx 8.57$. The high-density row shows that residential nodes are also penalized when concentrated in saturated regions. Despite a 125× bandwidth advantage, the saturated datacenter remains slightly below the residential node because regional density and datacenter provenance are both penalized. Tokenomic simulations should therefore use Eq. 8 and the stable supply controller rather than arbitrary fixed ϕ and β constants.

10 Conclusion

CoinCeeper establishes a mathematically rigorous Proof-of-Network consensus framework that securely bridges physical network verification with deterministic blockchain settlement. By formalizing geographic scarcity, resolving NAT traversal via asymmetric pairing, reducing relay legal risk through auditable compliance controls, and integrating gasless Account Abstraction, the protocol provides a scalable, secure, and economically sustainable foundation for next-generation DePIN architectures. Transparent token allocation, corrected Sybil mathematics, multi-modal verification, and a precise execution roadmap reduce investment and technical risks, paving the way for enterprise adoption.

References

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[2] V. Buterin, "A next-generation smart contract and decentralized application platform," 2014.

[3] Helium Systems, "Helium: A decentralized wireless network," Whitepaper, 2019.

[4] Protocol Labs, "Filecoin: A decentralized storage network," Protocol Specification, 2017.

[5] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman, "SybilGuard: Defending against Sybil attacks via social networks," *IEEE Symposium on Security and Privacy*, 2006.

[6] A. Perrig, R. Canetti, J. D. Tygar, and D. Song, "The TESLA broadcast authentication protocol," *RSA CryptoBytes*, 2002.

[7] B. Constantine, G. Forget, R. Geib, and R. Schrage, "Framework for TCP Throughput Testing," RFC 6349, 2011.

[8] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, 1960.

[9] European Union, "Regulation (EU) 2022/2065 on a Single Market for Digital Services (Digital Services Act)," 2022.

[10] The Tor Project, "Tor legal FAQ and relay operator guidance," Technical guidance.

[11] National Institute of Standards and Technology, "Recommendation for Key Management: Part 1 – General," NIST Special Publication 800-57 Part 1.