Astatine Chain: Artificial Proof-of-Work on a 3-Node Network

Emyr

Jecta Labs Research & Development

November 2025

Abstract

Astatine Chain is an Artificial Proof-of-Work (PoW) experiment operating across three autonomous smart-contract nodes. Each node begins with a distinct 256-bit seed and an equal share of the total supply, allowing users to mine independently on any endpoint. This paper outlines Astatine's decentralized architecture, mining mathematics, halving logic, retargeting adaptation, and fairness model that rewards the best (not the first) proof. The chain serves as a thought experiment on equilibrium, entropy, and cooperation within distributed computation.

1 Introduction

Conventional Proof-of-Work blockchains rely on a single global chain where the first valid hash wins. Astatine Chain instead distributes its logic across three smart-contract nodes, each acting as a self-contained mini-chain with identical rules but distinct starting seeds. Every node holds one-third of the total token supply and independently validates proofs, allowing parallel mining competitions under the same economic conditions.

This design explores how fairness and stability emerge when work is decentralized not only among miners but also among validation endpoints. By dividing supply, applying same halving logic for each node, and dynamically retargeting difficulty, Astatine demonstrates a system that converges toward equilibrium without centralized scheduling. The following sections explain this mechanism mathematically and visually.

2 Nodes

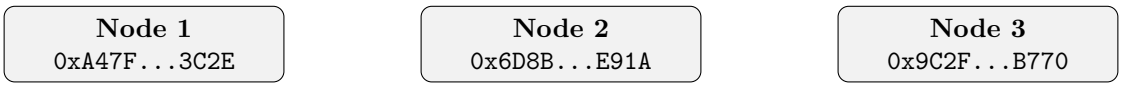

Figure 1: Three-node structure of Astatine Chain. Each node holds $S_{\text{node}} = S_{\text{total}}/3$ and starts from a unique 256-bit seed.

Each node maintains:

- its own seed $seed_i$ and block state,
- a local mining block B_i for proof collection,
- and an equal token supply share $S_{\text{node}} = S_{\text{total}}/3$.

Nodes operate asynchronously. They all have their own chains itself but working on a same goal on mining state. Users may submit proofs to any or all nodes. Global behavior emerges from shared parameters but independent randomness.

3 Blocks

A block on node N_i at height h captures the winning proof for that round and the context required to advance the local chain state. Conceptually we denote it as $B_{i,h}$.

Block $B_{i,h}$	
height	h
prev_hash	SHA256($B_{i,h-1}$)
seed	$seed_i$ (256-bit)
target	$target_i = \lfloor C/D_i \rfloor$
difficulty	D_i
best_hash	min(h_m) over valid submissions
best_miner	winner address (bech32)
nonce	$nonce_*$ of winning proof
header	miner-supplied header bytes

$$B_{i,h}.\text{id} = \text{SHA256}(seed_i \,\|\, addr_{canon} \,\|\, nonce \,\|\, header) \quad \text{(consensus identifier)}$$

Figure 2: Canonical block contents on node N_i. Each block finalizes the lowest valid hash found within the node's mining block B_i.

This structure keeps each node self-contained while remaining compatible with Astatine's fairness rule (*best hash wins*), halving thresholds , and per-node retargeting.

4 Mining

The mining algorithm uses double SHA-256 hashing:

$$h_i = H(H(seed_i \,\|\, addr_{canon} \,\|\, nonce \,\|\, header)). \tag{1}$$

A hash is valid if $h_i < target_i$, where

$$target_i = \left\lfloor \frac{C}{D_i} \right\rfloor, \tag{2}$$

and D_i is the node's current difficulty, $C = 2^{256}$.

The expected probability of success after k attempts is

$$p = 1 - (1 - \frac{1}{D_i})^k.$$

Each valid submission is stored temporarily within W_i until finalization selects the lowest hash.

5 Halving

Each node mints tokens from its local supply S_{node}. Halving occurs whenever the minted amount reaches half of the node's remaining supply, n is the current halving index and M_i is the next halving threshold as supply amount :

$$M_i = S_{node} - \frac{S_{\text{node}}}{2^n}, \tag{3}$$

and block rewards follow:

$$R_{i,n} = R_0 \times 2^{-n}. \tag{4}$$

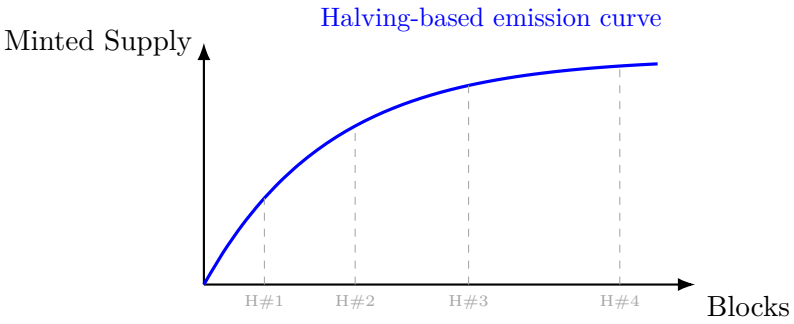

Figure 3: Each node's minted supply approaches S_{node} asymptotically through successive halvings.

Over infinite epochs, M_i converges to S_{node}, guaranteeing predictable scarcity and finite issuance.

6 Retargeting

Difficulty adjusts every $K = 100$ blocks. EMA (Exponential Moving Average) is being calculated for the each block mining to get the result of the difficulty retargeting. Let T_{actual} denote the measured time for those 100 blocks and $T_{\text{target}} = 100 \times \tau$, where $\tau = 300\,\text{s}$.

$$D_{new} = D_{old} \times \frac{T_{\text{actual}}}{T_{\text{target}}}. \tag{5}$$

To avoid volatility:

$$1 \leq \frac{D_{new}}{D_{old}} \leq 4.$$

If blocks are produced too quickly ($T_{\text{actual}} < T_{\text{target}}$), D increases. If slower, D decreases. Each node performs this independently, keeping its mining pace close to the desired interval.

7 Mining Race and Fairness

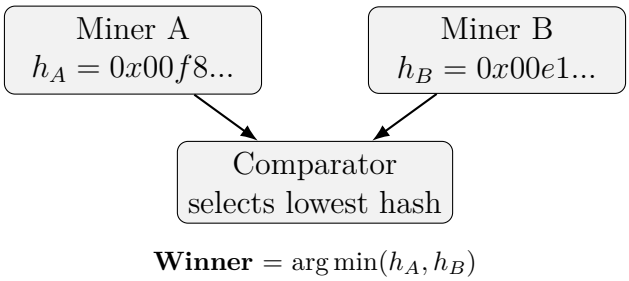

Figure 4: Fair competition: the lowest valid hash wins, eliminating front-running advantages.

Astatine replaces "first-wins" logic with "best-wins." If Miner A finds a valid hash h_A but Miner B later finds $h_B < h_A$, the system selects Miner B. Formally:

$$\text{Winner} = \arg\min_{i \in W}(h_i).$$

This model neutralizes latency advantages and anchors fairness purely on computational quality.

8 Conclusion

Astatine Chain represents more than a distributed PoW prototype, it is an allegory of balance between chance and structure. Three nodes, three perspectives of the same equation, searching for order within mining powers. Halving teaches restraint; retargeting teaches adaptation; and the mining race reminds us that perfection, not haste, defines victory.

In this system, miners do not race against time but against entropy itself. Victory belongs to the one who finds beauty in the smallest number.

Keywords: Proof-of-Work, Injective, Mining Fairness, Halving, Retargeting, Cryptography.