Helios: An Interchain Paradigm of Reputation-Based Cross-Chain Consensus

Frederick MARINHO
frederick@helios-foundation.org
10/06/2024

Disclaimer

This whitepaper is intended to provide an overview of the Helios Chain project's potential capabilities, goals, and innovations. It is not an offer to sell or a solicitation of an offer to buy any assets or securities, nor does it constitute financial, investment, or legal advice.

The information contained herein represents the current vision for Helios Chain but is subject to change as the project evolves. No guarantees or commitments are made regarding the features, functionality, or timeline described in this document. Helios Chain, its developers, and contributors assume no responsibility for any actions taken based on the information presented herein.

Readers are encouraged to conduct independent research and consult with professional advisors before making any financial decisions or investments related to blockchain technologies.

Abstract

This paper introduces and highlights a new consensus paradigm using a hybrid variation of the Proof of Stake (PoS) labeled **Interchain Proof of Stake and Reputation (I-PoSR)**. By leveraging assets stored and staked on multiple external blockchain networks to secure the Helios network, **I-PoSR** thereby creates a decentralized, cross-chain consensus model that is flexible and robust. Through **I-PoSR**, we integrate a reputation-based mechanism called **Hyperion** that acts as a light client to verify and enhance external chain data, enabling the network to certify the information received and sent over multiple blockchains. The paper explores the technical structure of the **I-PoSR** consensus model, illustrating how it provides an interoperability framework for a scalable, resilient network capable of benefiting from the industry's most significant players and innovative networks.

1. Introduction

Current blockchain systems usually rely on a single consensus mechanism to maintain and assure their network's integrity, making an ecosystem non-time resilient and hardly impacted by technological upgrades and improvement, a single point of failure from a single cryptographic solution point of view. We have all heard concerns about the evolution of the ecosystem, such as, “What about quantic? What about AI? What will happen next, then?”

The lack of long-term technical vision forces us to constantly upgrade and fork the existing solutions, which are, in fact, inherent to evolution and are in the essence of the human condition. The only possibility is to accompany and fix what was true once and became false. **I-PoSR** presented here is designed to fill that gap by leveraging the current and future best consensus and cryptographic solutions that can be collateralized into a single chain that evolves in real time.

I-PoSR leverages the best of current and future cryptographic and consensus solutions, allowing them to be collateralized into a single, adaptive chain. Much like stablecoins such as DAI (USDS), backed by diversified assets to ensure stability, **I-PoSR** aggregates security from multiple blockchains, creating a flexible, resilient consensus that draws on the collective strengths of various significant networks. In doing so, **I-PoSR** enables Helios Chain to adapt to new cryptographic standards and consensus innovations as they emerge, making it future-proof and technologically resilient.

2. Chain Overview

The main points of the **Helios Chain** vision are flexibility and robustness. It is designed to address the limitations of traditional single-consensus systems through **Interchain Proof of Stake and Reputation (I-PoSR)** to build a secure, adaptable, and future-proof blockchain ecosystem.

Helios delivers a resilient and made-for multi-chain interoperability network by leveraging cross-chain staked assets and enabling a decentralized reputation layer. The vision behind this is to become the ecosystem for a diversified and interconnected blockchain ecosystem, where users can safely manage and transact diverse assets across chains without compromising on security or decentralization (ex, central bridges or CEXs)

The main components of the ecosystem are the **Helios Core** and the **Hyperion Module.**

1. Helios Core

The **Helios Core** serves as the foundational layer of the ecosystem, which serves as the primary chain and the central component of the network’s consensus and security architecture. It is responsible for the network's core functions, including the creation (minting) of **Helios** tokens, governance, and collaboration with external networks through **Hyperion Modules.**

When assets from external chains (such as ETH from Ethereum) are deposited through **Hyperion Modules**, the **Helios Core** wraps these assets to create a corresponding Helios-native token. This process is based on feedback from Hyperion, which confirms the deposit events on the original blockchain. This wrapped token is accessible on Helios and can be used within the ecosystem.

Once users have received wrapped assets on Helios, they can either delegate these funds to a node they control or delegate to other nodes. **Helios** provides an interface for users without nodes to facilitate easy delegation to trusted entities, allowing them to participate in the network's consensus without running a node.

Only nodes with a sufficient delegation-weighted asset stake power score can participate in the consensus process. The **Helios Core** distributes consensus responsibilities based on the node score derived from the amount and diversity of funds delegated to each node. The score influences a node's ability to participate in validation, enhancing security and incentivizing high-performing, diversified nodes.

2. Hyperion Modules

Hyperion Modules are attached to the central core and are non-mandatory for operating a **Helios Core** and participating in the main consensus. It is optional but essential for the whole network workflow. Running a node with Hyperion enabled will grant extra Helios rewards from the main chain comparable to the MEV boost enabled on an Ethereum node.

The **Hyperion modules** are a decentralized network of sub-modules that connect the **Helios Core** to external blockchain ecosystems, providing critical cross-chain verification and asset transfer capabilities like a relayer or a bridge. Each Hyperion can be started with different sub-modules that act as **light clients** for a specific external blockchain (e.g., Ethereum, Solana), allowing it to validate transactions and manage deposits and withdrawals between **Helios Core** and external chains.

The **Sub-Modules** verify deposits and withdrawals on external chains, ensuring that assets like ETH on Ethereum are securely deposited before being wrapped on Helios. This process involves multi-signature attestation, where multiple **Hyperion modules**, each associated with specific trusted nodes, confirm and authenticate cross-chain transactions. This verification ensures that all cross-chain assets are properly authenticated and meet Helios's security standards before being utilized within the ecosystem.

When users initiate transactions on **Helios Core** that involve external chains, such as withdrawing a token back to Ethereum, the **Hyperion Modules** manage the relay of these transactions. They gather multi-signature confirmations from reputable nodes sent to the subchain contracts and provide event feedback to Helios, allowing the Core to update balances and mint/burn tokens accordingly.

In the **Helios ecosystem**, any asset from a supported blockchain can be wrapped and utilized as long as it is transferred to the Helios network through an authorized sub-module. **Hyperion Modules** provide a flexible, open-source structure that allows new sub-modules and light clients to be added, enabling integration with additional blockchain networks.

To integrate a new blockchain, developers can contribute a new sub-module by creating a light client for the desired chain and implementing it within the existing **Hyperion framework**. This open-source approach allows Helios to expand its supported chains quickly, ensuring continuous adaptability as new blockchain networks emerge.

Once a new sub-module and light client are developed, the next step is validation through Helios's **DAO** vote governance system. The **DAO** conducts a vote to determine if the new sub-module is valid and can participate in the consensus. Upon approval, assets from this chain become potentially eligible to contribute to Helios's consensus, and their value is recognized within the network. This decentralized approval process ensures that only trusted chains and assets are included, supporting a secure and diverse ecosystem while allowing **Helios Chain** to remain adaptable and future-proof.

Node participation is regulated by a scoring system based on delegated assets. The consensus process prioritizes nodes with higher scores derived from diverse and substantial delegations. This incentivizes nodes to maintain a high level of performance and security, thereby reinforcing **Helios Chain's** overall resilience by leveraging multiple network consensus.

2. Interchain Proof of Stake and Reputation (I-PoSR)

Helios Chain is entirely based and depends on the Interchain **Proof of Stake and Reputation (I-PoSR)** model, a hybrid consensus designed to address the limitations of single-consensus mechanisms. Unlike standalone **Proof Of Stake (POS)** that only operates on a single network and relays its entire security to a single asset value, **I-PoSR** integrates cross-chain staked assets from multiple networks (e.g. Ethereum, Solana, Avalanche) and uses a reputation-based scoring to ensure consensus external chain data reliability.

By staking assets from external diversified chains, Helios achieves security through diversified, slashable assets across multiple networks, increasing the decentralization of his stakes. **Hyperion-enabled** nodes incorporate a reputation mechanism that rewards consistent and trustworthy nodes, reducing the reliance on a single node and promoting a more decentralized data relayer pool.

In the **I-PosR** model, users can delegate assets (such as ETH, SOL, and AVAX) to validator nodes on Helios. Each asset contributes to the node total score, derived from the quantity, diversity, and risk profile of the tokens held. The score is dynamically calculated with constant **DAO** updates, creating a market of weight value for consensus assets based on their evolution over time.

Each type of token is assigned points proportional to its value and risk level based on a vote within the **DAO governance** structure, implying the majority of Helios holders, to incentivize diversification tokens with lower representation within the network, receive higher multipliers increasing the potential reward for nodes holding those consensus assets.

Nodes with **higher scores** based on their diversified assets and scores gain a larger share of the consensus rewards and participate more actively in the consensus. This process encourages nodes to balance their holdings for maximum security and efficiency of the multi-chain total stakes. If everyone stakes only in ETH, the decentralized beneficial aspect will drastically be reduced as we need to leverage each network asset properly.

To calculate the rewards for each node, **I-PoSR** applies a formula that combines the token points and network risk factors as follows:

$$Reward_i = \frac{\sum_j \left(P_{i,j} \times \left(1 + \frac{100 - \left(\frac{T_j}{T_{total}} \times 100 \right)}{100} \right) \right)}{\sum_k \sum_j \left(P_{k,j} \times \left(1 + \frac{100 - \left(\frac{T_j}{T_{total}} \times 100 \right)}{100} \right) \right)} \times R$$

- P*i*, *j*: Points for token type *j* held by Node *i* (e.g, 50 SOL X 170 points)
- T*j*: Total network points for token type *j*.
- *T total*: Total points across the network (sum of all T*j* for all token types).
- R: Total reward to distribute (e.g, 100 Helios)
- Reward *i*: Final total reward for the Node

This formula incentivizes nodes across the network to hold a diversified set of tokens from consensus-integrated chains and applies a risk multiplier based on the whole distribution. Nodes that support less common assets receive a proportionally higher reward.

In addition to its core staking mechanism under Helios Core supervision with assets delegation and governance importance vote, I-PoSR incorporates a reputation-based system to manage and validate cross-chain transactions securely through **Hyperion modules**.

Once developed and open-source, each subchain (e.g., Ethereum) submodule has to be accepted and integrated as an available submodule for Hyperion. Any nodes acting as **Hyperion-enabled** nodes can embed the new submodules into their execution if they want to benefit from the new integrated network rewards (as native asset fee or bridged token fee).

The network will then decide if the assets from that new chain hold any value as long as users are willing to use the subjacent asset from that new sub-module. This is the first part of integrating a new blockchain into **I-PoSR**. Once it has been accepted and seems utilizable with enough hyperion nodes securing the network, the community can settle a governance vote to select some potential assets from that chain sub-modules confirmed as valid to secure the network.

Each subchain smart contract initially recognizes known reputed nodes at deployment for genesis provided by the **Helios Foundation** or, later, from the DAO governance. Then, nodes can build a reputation over time through participation and consistent multi-signature contributions. The more a **Hyperion-enabled** node confirms external chain information with reputed nodes certifying the same data, the more reputation it wins.

A node reputation R_{node} is calculated based on initial reputation multi-signature reputation, and decay over time, assuring the Hyperion keeps being present on some transfers to subchains as follows:

$$R_{\text{node}} = R_{\text{initial}} + \sum_{s \in \text{signatures}} f(s) + b_{\text{random}} - d(t)$$

- $R_{initial}$: Initial reputation for the node.
- $f(s)$: Reputation gained for each valid multi-signature s contributed by a trusted Hyperion node.
- b_{random}: Raise applied to one node per transaction, temporarily increasing its reputation and selection luck, derived from an unpredictable source
- $d(t)$: Decay function over time t, gradually reducing the reputation for inactive nodes.

Nodes experience a natural decay in reputation to ensure they stay active and cannot take control of the multi-sig process. If all nodes fall below the reputation threshold due to inactivity, it will dynamically adjust based on the maximum average reputation of the most trustworthy nodes, ensuring the network does not get stuck while staying safe.

We introduce the b_{random} variable to prevent Sybil attacks by randomly selecting one **Hyperion-enabled** node per transfer. This node receives a reputation raise without necessarily earning it to a reputed node. This node gains priority for future transactions, promoting network diversity.

We also require a **Hyperion-enabled** node to delegate into the **Helios Chain** with a minimum score for the node. It is mandatory to ensure we can also slash malicious actors trying to fake Hyperion's external chain data. The minimum viable score has to be determined at the genesis by comparing the node score with asset weight from the starting chains, therefore giving a minimal node score required to enable **Hyperion** on a delegating **Helios** node.

Once minimal default has been set, **a DAO governance vote** is possible with the majority of the circulating supply to change the rules and potential minimal score if the network feels like there are incoherence or intrinsic security risks.

When a transfer withdrawal request from Helios is emitted, the funds are burned and saved into the next block. Then, the **Helios Core** will scan the network for available and compatible **Hyperion-enabled** nodes connected to the correct sub-modules with the minimal reputation threshold and accept the fee models. To know the threshold of accepted reputation, an average score from the sub-chain smart-contract known nodes with reputation is calculated as follows:

$$\text{Reputation}_{\text{min}} = \frac{\sum_{i=1}^{5} R_i}{5}$$

- R_i: Reputation score of each of the last 5 Hyperion nodes that processed a transfer on the subchain
- 5: Represents the five most recent nodes, giving a real-time average of the reputation needed for eligibility.

By leveraging the reputation scores of the last nodes that handled transfers, Helios maintains an up-to-date, dynamic threshold for each subchain without needing continuous smart-contract interactions. **Helios Core** can check on his previous blocks to see which nodes emitted transfers and their reputation, knowing the minimum required for the subchain without interconnection.

For the fee model of each sub-module, the **Hyperion-enabled** nodes can choose between two models: accepting only native assets that are included in Helios consensus means that we can easily calculate the minimal fees required by one Hyperion for that specific asset as Helios Core knows its stake score or accept to be rewarded in the external chain token directly, meaning that the node owner receives either Helios Token or the asked asset to be withdrawn (e.g., Ethereum).

Hyperion can also ask an external off-chain source for an asset score to help him decide his minimal requirement to ensure he is not losing from signing and executing the cross-chain transfers. It is getting more tricky for any wrapped assets that hold zero intrinsic value to accept fees to be paid. That's where the off-chain source can help to decide which asset the **Hyperion-enabled** node considers valid to be used and the minimal accepted fee.

Let's give a more concrete example:

The **Hyperion Ethereum sub-module** contains the native assets Ethereum and WETH, both used and valid for the consensus, weighted at a balance of 1000 decided by constant **DAO validation**. One can determine that based on the ETH market price (let's use 5000$ as an example), one total Helios Wrapped ETH equals a balance of 1000, so a cost of approximately 5000$: it is now more accessible for the Hyperion to decide the minimum of fees to ask to perform the transfer.

Now, the user wants to burn and transfer the MAGA token to ETH, which is entirely unweighted and unknown from the consensus. Hyperion requests a fixed Helios token price based on the relative subchain native asset value or asks his trusted off-chain third party if he knows the MAGA token contract and its intrinsic value, giving the flexibility to win rewards in MAGA token if wanted.

When a **Hyperion-enabled** node meets all the requirements to participate in the Helios network, specifically the minimum delegation weight score and reputation needed to act as an official signer, it can register itself as active for a specific sub-chain. During registration, which is refreshed periodically, the node specifies its minimum acceptable fee for processing transfers on that sub-chain and chooses its fee model. The node can accept fees in Helios tokens only, setting a minimum fee for each sub-chain or directly receiving payments in the **transferred asset** (e.g., the token being burned). Then, the network pings it to confirm its acceptance of that specific asset as payment.

Once the node is online and these conditions are met, the network recognizes the node as ready and eligible for transaction handling on the designated sub-chain. When a user initiates a transfer through the Helios front-end, **Helios Core** calculates an average minimum fee in Helios tokens or the user's selected token based on the registered fees of all eligible **Hyperion-enabled nodes** for that sub-chain.

In this process, Helios Core randomly selects several **Hyperion nodes** as potential primary nodes to handle the transaction. If none of the selected nodes pick up the transaction within a designated timeframe, they are penalized on their **delegated stake**, as they have declared their availability to process transactions for this sub-chain. If the transaction remains unhandled, Helios Core mints the user's tokens back on Helios in the next block, restoring the original balance after the grace period.