Cover Letter — Novelty and Scope Statement

Dear Editors and Reviewers,

This submission formalizes **genesis-anchored lineage verification** as a distinct verification primitive for light clients operating in offline, intermittently connected, or asynchronous blockchain systems.

Positioning Relative to Existing Work

Prior work on light clients (SPV [1], Flyclient [2], Mina [3]) addresses *continuity*—verifying that one state follows another—and *execution*—verifying that state transitions follow protocol rules. This paper addresses an underexplored third dimension: *lineage*—verifying that a light client has reconnected to the same network after downtime, without socially trusting genesis.

The key distinction:

- **Existing work** optimizes *how* to verify blockchain properties (efficiency, succinctness, proof size)
- **This work** addresses *which* blockchain is being verified (network identity and origin authentication)

These are orthogonal verification goals, not competing approaches. Genesis anchoring is designed to compose with existing techniques by providing an authenticated genesis layer that current systems assume as a trusted constant.

Why This Problem Has Been Underexplored

Genesis has historically been treated as a social constant rather than a cryptographic surface because:

1. Genesis occurs once at network inception when social coordination costs are minimal
2. Established networks achieve strong social consensus on canonical genesis through years of operation
3. The distinction between network identity and state continuity was not previously isolated as requiring separate cryptographic treatment

However, as blockchain adoption expands to mobile and web platforms with intermittent connectivity, the inability to cryptographically verify network identity becomes a significant limitation. Browser-based clients, cross-chain applications, and offline-resilient verifiers benefit from cryptographic origin authentication rather than relying solely on social consensus.

Core Novelty Claims (Modest Framing)

To our knowledge, existing literature has not:

- **Isolated network identity verification as a distinct primitive** separate from state continuity and execution correctness
- **Formalized the distinction between lineage and continuity** for light-client verification (Definition 4.4)

- **Treated genesis as a cryptographically time-bounded object** through external proof-of-work anchoring, rather than a socially assumed constant
- **Analyzed how external temporal anchoring enables identity-preserving reentry** for light clients after network outages or restarts, particularly in asynchronous DAG systems

We do not claim this problem is entirely unrecognized. SPV, Flyclient, and Mina implicitly assume genesis trust. Our contribution is to *make this assumption explicit*, formalize an alternative approach, and demonstrate its applicability to offline-resilient and DAG-based systems.

Scope and Limitations

This work addresses network identity preservation, not complete light client security. We explicitly do not claim to solve:

- Execution correctness (validating state transitions)
- Data availability (ensuring block data is accessible)
- Fraud detection (identifying invalid blocks)
- Fork choice (determining canonical branches)

Genesis-anchored verification is a foundational trust primitive intended to compose with execution proofs, fraud proofs, and data availability systems. Section 9.5 and Section 11 explicitly position this work as orthogonal to existing techniques.

Why Reviewers Should Care

This paper:

1. Identifies an implicit trust assumption in existing light client architectures
2. Provides a formal framework for removing that assumption through external anchoring
3. Demonstrates applicability to emerging use cases (browser verification, intermittent connectivity, DAG systems)
4. Explicitly characterizes what is and is not verified, preventing overstatement of security properties

The contribution is architectural and verification-focused, filling a gap in the light client security model rather than competing with existing verification techniques.

We believe this framing is appropriately scoped, orthogonal to existing work, and relevant to modern verification contexts where social genesis trust is insufficient.

Sincerely,
[Authors]

References

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

[2] B. Bünz et al., "Flyclient: Super-light clients for cryptocurrencies," IEEE S&P 2020.

[3] I. Meckler and E. Shapiro, "Coda: Decentralized cryptocurrency at scale," O(1) Labs, 2020.