PAPER 4 — MATHEMATICS & CRYPTOGRAPHY

The Formal Cryptographic Model of NOVAK: Deterministic Execution, HVET, RGAC, and Provable Action Boundaries

Abstract

This paper presents the mathematical foundations of the NOVAK Protocol, focusing on deterministic proof-before-action execution, hash verification primitives, and the Recursive Global Audit Chain. We formally define the system's correctness properties, failure boundaries, and cryptographic assumptions.

---

1. Introduction

The NOVAK Protocol enforces deterministic computational integrity through:

- **Execution Identity Receipts (EIR)**
- **Hash-Verified Execution Traces (HVET)**
- **Recursive Global Audit Chains (RGAC)**
- **Deterministic Safety Gates**

This aligns NOVAK with research traditions in:

- formal verification
- cryptographic commitments
- model checking
- attested execution
- verifiable computing

---

2. Mathematical Preliminaries

Let:

- **R** = rule function
- **D** = input data
- **O = R(D)** = correct output
- **HR = H(R)**
- **HD = H(D)**
- **HO = H(O)**

NOVAK constructs:

HVET = H(HR ∥ HD ∥ HO ∥ timestamp)

Where **H** is a cryptographic hash satisfying:

- preimage resistance
- second-preimage resistance
- collision resistance

---

3. Deterministic Safety Gate

The Safety Gate enforces:

```
If HVET_proposed ≠ HVET_expected → Reject Execution
```

No probabilistic interpretations.
No heuristics.
No learning-based inference.

It is a **hard, algebraic execution rule**.

---

4. Recursive Global Audit Chain (RGAC)

Each action appends:

```
RGAC(n+1) = H( RGAC(n) // HVET(n) )
```

This produces:

- immutability
- global consistency
- independent verifiability
- temporal ordering guarantees

Aligned with blockchain **only in audit**,
NOT in execution, consensus, or decentralization.

---

5. Correctness Theorems

Theorem 1: Undetectable Tampering is Impossible

If H is secure and HVET includes HR, HD, HO, then:

Any attempt to modify:

- the rule
- the input
- the output
- intermediate values

produces a mismatch.

Theorem 2: Unauthorized Execution is Undecidable Under NOVAK

Given deterministic R and validated HVET:

```
R'(D') cannot execute if H(R'), H(D'), H(R'(D')) differ
```

Theorem 3: Forward Integrity

No function of future state can retroactively alter past RGAC entries without breaking hash hardness assumptions.

---

6. Computational Hardness Model

NOVAK relies solely on:

- SHA-256 class hash functions
- Concatenation commitments
- No need for asymmetric cryptography
- No need for zero-knowledge proofs
- No need for consensus protocols

This increases:

- speed
- scalability
- implementability
- verifiability

7. Conclusion

NOVAK is a deterministic integrity enforcement system whose mathematical properties ensure unforgeable execution. It represents a new category of cryptographically enforced computation.