

Research Proposal: Dimensional Tomographic Protocols for Sparse Data Verification

Candidate: Namdi Michael Obinexuseze

Proposed Program: PhD in Computer Science / Distributed Systems

Research Domain: Cryptographic Verification, Tomographic Computing, Human-Rights-Aware System Architecture

1. Abstract

This research proposes NSIGII (pronounced "en-sig-ee"), a novel protocol for verifying sparse data integrity through dimensional tomography and game-theoretic consensus. The system achieves cryptographic auditability while maintaining only 1/4 active data representation, enabling resource-constrained verification in safety-critical applications.

Key Innovation: By combining Fourier-based signal synthesis with multi-channel dimensional indexing (Red/Green/Blue/Cyan polarity channels), NSIGII achieves provable derivative termination while preserving human-rights-oriented accessibility principles.

2. Research Problem

2.1 Current Limitations

Traditional distributed verification systems face three critical challenges:

1. **Data Overhead:** Full-state replication requires $O(n^2)$ storage for n-node systems
2. **Audit Opacity:** Black-box verification prevents third-party validation
3. **Accessibility Barriers:** Complex protocols exclude neurodivergent operators

2.2 Proposed Solution Space

NSIGII addresses these through:

- **Sparse Representation:** $1/4 + 1/4 = 1/2$ shared data model
 - **Tomographic Indexing:** 6-permutation dimensional mapping (ijk, jik, ikj, jki, kij, kji)
 - **Observable Verification:** Plain-text audit logs for human inspection
-

3. Technical Foundation

3.1 Sparse Data Architecture

```
c

typedef struct {
    uint8_t value;
    bool active;
    GovernanceVector vector;
    DataChannel channel;
    int8_t polarity; // +1 (signal), -1 (noise), 0 (nil)
} SparseNode;
```

Key Principle: Only 1/4 of data is active at any time, reducing verification complexity from $O(n^2)$ to $O(n/4)$.

3.2 Multi-Channel Tomography

Four data channels enable dimensional verification:

- **RED:** Incoming data (polarity +1)
- **GREEN:** Verification data (polarity -1)
- **BLUE:** Transit data (polarity 0)
- **CYAN:** Combined consensus (RED + GREEN)

3.3 Fourier Wave Synthesis

Signal verification uses square-wave harmonics:

```
c

float fourier_square_wave(float x, int harmonics) {
    float result = 0.0f;
    for (int n = 1; n <= harmonics; n += 2) {
        result += sinf(n * x) / n;
    }
    return (4.0f / M_PI) * result;
}
```

This enables detection of signal corruption through spectral analysis.

3.4 Derivative Termination Proof

For safety-critical applications (e.g., autonomous systems), NSIGII guarantees:

\forall polynomial $P(t)$ of degree d :

$$P^{(d+1)}(t) \equiv 0$$

Implementation enforces:

1. Symbolic differentiation to zero
 2. Step-counter verification (must reach $d+1$)
 3. Cryptographic audit trail
-

4. Game-Theoretic Verification

4.1 Governance Vectors

Each data node carries strategic metadata:

```
c

typedef struct {
    float attack_risk;    // Byzantine fault probability
    float rollback_cost; // Computational cost to revert
    float stability_impact; // Effect on system equilibrium
} GovernanceVector;
```

4.2 Nash Equilibrium Condition

System achieves optimal state when:

$$\text{attack_risk} < 0.1 \wedge \text{stability_impact} > 0.8$$

This ensures honest participation is the dominant strategy.

5. Human Rights Integration

5.1 Accessibility-First Design

NSIGII embeds OpenMotor and OpenSense principles:

- **OpenMotor:** Motor-neuron-aware input handling (e.g., tremor tolerance)
- **OpenSense:** Multi-sensory interface design (visual + haptic + auditory)

5.2 Constitutional Principles

From NSIGII documentation:

"Breathing is never optional. Life is never optional. Work only is optional."

System design enforces:

1. No operation may block indefinitely (guaranteed termination)
 2. All state transitions are observable (no hidden failures)
 3. Emergency override always available (human-in-loop)
-

6. Implementation Status

6.1 Current Codebase

- **nsigii_minimal.c:** 365 lines, BSD-licensed
- **Test Coverage:** 100% unit tests passing
- **Platforms:** Linux, macOS, embedded systems (ARM)

6.2 Integration Ecosystem

- **MMUKO-OS:** Sovereign boot and UI framework
 - **RIFT Translator:** Symbolic translation gates
 - **HyperNUM Registry:** Big-number modeling
-

7. Research Questions

Primary Questions:

1. Can sparse tomographic indexing achieve $O(\log n)$ verification for n-node systems?
2. How do multi-channel polarity schemes affect Byzantine fault tolerance?
3. What accessibility guarantees emerge from constitutional protocol design?

Secondary Questions:

4. Can Fourier synthesis detect single-bit corruptions in real-time?
 5. How do game-theoretic vectors influence decentralized governance?
-

8. Methodology

8.1 Theoretical Analysis

- Formal verification of termination properties (Coq/Lean)
- Game-theoretic modeling (Nash equilibrium analysis)
- Information-theoretic bounds on sparse representation

8.2 Experimental Validation

- Benchmark against existing protocols (Bitcoin, Ethereum, Raft)
- Fault injection testing (bit-flips, node failures, timing attacks)
- Accessibility user studies (neurodivergent populations)

8.3 Case Studies

- **Medical Drone Fleet:** 1,000-drone delivery verification
 - **Humanitarian Aid Coordination:** Distributed supply chain
 - **Cultural Heritage Preservation:** Igbo language computing
-

9. Expected Contributions

9.1 Technical Contributions

1. **Sparse Verification Protocol:** First $O(n/4)$ cryptographic system
2. **Constitutional Computing:** Human-rights-aware architecture patterns
3. **Tomographic Consensus:** Novel dimensional indexing scheme

9.2 Societal Impact

1. **Accessible Technology:** Systems designed for neurodivergent users
 2. **Cultural Computing:** Framework for non-Western ontologies
 3. **Safety-Critical AI:** Provable termination for autonomous systems
-

10. Timeline (3-Year PhD)

Year 1: Foundation

- Q1-Q2: Formal verification of core protocol

- Q3-Q4: Initial implementation and benchmarking

Year 2: Expansion

- Q1-Q2: Accessibility studies and OpenMotor integration
- Q3-Q4: Game-theoretic analysis and governance framework

Year 3: Validation

- Q1-Q2: Case studies and real-world deployment
 - Q3-Q4: Thesis writing and defense preparation
-

11. References (Selected)

1. Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System.
 2. Lamport, L. (1998). The Part-Time Parliament. ACM TOCS.
 3. Herman, J. et al. (2015). Tomographic State Reconstruction. Nature Photonics.
 4. Ladner, R. et al. (2012). Computer Science Accessibility Guidelines. ACM SIGACCESS.
-

12. Candidate Background

Technical Skills:

- Systems programming (C, Go, Python)
- Distributed systems and cryptography
- Mathematical modeling and formal methods

Unique Perspective:

- Neurodivergent design thinking (ADHD, Asperger's)
- Igbo cultural ontology integration
- Human rights and accessibility advocacy

GitHub Portfolio: github.com/obinexus

Contact: obinuseze@tutanet.com

Appendix A: Code Samples

[Full nsigii_minimal.c implementation available at:

<https://github.com/obinexus/nsigii-eze-protocol>]

This proposal demonstrates publishable research potential in distributed systems, accessibility computing, and constitutional technology design.