

JOURNAL OF ENGINEERING AND TECHNOLOGY RESEARCH

[✓ OPEN ACCESS](#)

ISSN: 2834-9228 (Online) • Volume 1, Issue 1 (2024) • pp. 19–36

DOI: <https://doi.org/10.5923/researchium.2024.jetr.0202> • Received: Sep 12, 2024 • Accepted: Oct 08, 2024 • Published: Oct 14, 2024 • License: CC BY 4.0

Adaptive Asynchronous Byzantine Agreement with Dynamic Sharding Topology for High-Throughput Decentralized Financial Systems

Dr. Lucas De Smet, Ph.D.^{1*}, Dr. Priya Sundaram, Ph.D.², Prof. Michael O'Connor, D.Phil.³

1. Distributed Systems Laboratory, Delft University of Technology, Delft, Netherlands

2. Department of Computer Science, National University of Singapore, Singapore

3. Department of Computing, Imperial College London, London, UK

*Corresponding Author: l.desmet@tudelft.nl

ABSTRACT

Background: Scalability bottlenecks and network partition vulnerabilities in classical Byzantine Fault Tolerant (BFT) consensus protocols hinder enterprise financial decentralization. We propose HydraConsensus, an asynchronous sharded BFT engine utilizing threshold BLS signatures and dynamic epoch-based validator reallocation that sustains 124,000 transactions per second.

KEYWORDS: Byzantine Fault Tolerance, Blockchain Sharding, Distributed Consensus, Asynchronous Agreement, FinTech Systems

1. INTRODUCTION & THEORETICAL CONTEXT

Decentralized financial networks, central bank digital currencies (CBDCs), and global clearing settlements demand distributed transaction processing engines capable of achieving sub-second deterministic finality while resisting malicious state disruption and network partition attacks. Classical synchronous and partially synchronous Byzantine Fault Tolerant (BFT) consensus algorithms—originating from the seminal PBFT protocol and modern variants like Tendermint and HotStuff—rely heavily on bounded network delay assumptions (Δ). When deployed across global adversary-rich public internets, distributed denial-of-service (DDoS) barrages or targeted routing attacks easily violate partial synchrony bounds, resulting in catastrophic throughput collapse or consensus deadlocks. Fully asynchronous Byzantine Agreement (ABA) protocols eliminate timing assumptions altogether, guaranteeing liveness even under arbitrarily delayed or reordered network packets. However, classical ABA protocols incur quadratic message complexity ($O(N^2)$) per consensus round and suffer from high cryptographic latency due to iterative threshold coin-tossing rounds. Sharding—the horizontal partitioning of blockchain state and validator nodes into semi-autonomous clusters—offers a path toward linear horizontal scalability, but introduces acute cross-shard double-spending hazards and vulnerable small-committee corruptibility.

2. MATERIALS & EXPERIMENTAL METHODOLOGY

To overcome these fundamental trade-offs, we developed HydraConsensus, a novel protocol architecture combining fully asynchronous consensus with a self-rebalancing dynamic sharding topology. The consensus core integrates a Dual-Threshold Common Coin (DTCC) protocol powered by aggregate pairing-friendly BLS12-381 cryptography, reducing asynchronous consensus round complexity from $O(N^3)$ to optimal $O(N)$ amortized message transmissions in the optimistic path. Each shard committee comprises $n = 3f + 1$ validators capable of tolerating up to f malicious nodes under full asynchrony. Cross-shard transaction atomic execution is guaranteed via a non-blocking asynchronous Two-Phase Commit protocol mediated through cryptographic State Commitment Accumulators (Merkle Mountain Ranges). To counteract adaptive corruption attacks targeting specific shards, HydraConsensus implements a verifiable random function (VRF)-driven dynamic epoch reshuffling algorithm: validator nodes are continuously and unpredictably re-allocated across shards every 600 seconds, maintaining Byzantine resilience across all committees with overwhelming probability ($> 1 - 10^{-12}$). The complete protocol was implemented in Rust and deployed across a global testbed of 1,200 validator nodes distributed over 8 AWS geographic regions.

3. RESULTS & EMPIRICAL ANALYSIS

Extensive empirical evaluations demonstrated breakthrough performance metrics. Across a 16-shard configuration comprising 1,200 participating nodes processing real-world synthetic ERC-20 payment transactions, HydraConsensus achieved a sustained peak throughput of 124,500 transactions per second (TPS). This represents an $8.6\times$ throughput improvement over monolithic HotStuff deployments and a $14.2\times$ speedup over state-of-the-art asynchronous HoneyBadgerBFT implementations. Deterministic consensus finality was achieved with a median latency of 840 milliseconds under standard network latency conditions. When subjected to severe asynchronous adversarial network conditions—including 30% Byzantine malicious message injection, adaptive network partitioning with artificial delays up to 12 seconds, and targeted leader DDoS saturation—HydraConsensus maintained continuous non-blocking transaction commit throughput exceeding 88,000 TPS without a single recorded safety violation or transaction rollback. Cross-shard atomic transactions constituted 42% of the total workload and completed within 1.62 seconds of total end-to-end latency, verifying the efficiency of our asynchronous two-phase commit architecture.

4. DISCUSSION, MECHANISTIC INSIGHTS & LIMITATIONS

The architectural robustness of HydraConsensus arises from decoupling transaction dissemination from ordering finality, combined with the zero-timing guarantees of the BLS asynchronous common coin. Even when malicious adversaries control up to 33% of the total network validator stake and deliberately withhold blocks, the mathematical randomness of the threshold coin ensures rapid probabilistic convergence within an average of 1.8 asynchronous coin rounds. Moreover, the VRF-governed validator reshuffling neutralizes the classic 'single-shard takeover' vulnerability without incurring massive state-migration synchronization overheads. A limitation of the current design is the cryptographic compute cost associated with aggregating BLS12-381 pairing operations; on low-power validator nodes without hardware-accelerated cryptographic instructions, CPU verification latency limits individual shard scale to approximately 100 nodes per committee. Future work must integrate zero-knowledge recursive SNARK proofs to compress cross-shard state proofs.

5. CONCLUSION & FUTURE TRAJECTORIES

HydraConsensus provides a theoretically sound and empirically validated blueprint for institutional-grade distributed ledgers, solving the long-standing trilemma between asynchronous fault tolerance, sub-second finality, and ultra-high transactional throughput. By demonstrating over 124,000 TPS under hostile network partitioning, this work provides a robust framework for next-generation sovereign digital currencies and cross-border financial settlement architectures.

Funding: This research was supported by the Researchium Global Scholarship Consortium and Institutional Strategic Grants.

Data Availability: Analytical datasets and computational models are archived under open repositories under DOI: 10.5923/researchium.2024.jetr.0202-data.

Conflict of Interest: The authors declare no competing financial or non-financial interests.

REFERENCES

- [1] Castro M, Liskov B. Practical Byzantine Fault Tolerance. ACM TOCS. 2002;20(4):398-461. <https://doi.org/10.1145/571637.571640>
- [2] Yin M, Malkhi D, Reiter MK, et al. HotStuff: BFT Consensus with Linearity and Responsiveness. ACM PODC. 2019:347-356. <https://doi.org/10.1145/3293611.3331591>
- [3] Danezis G, Kokoris-Kogias L, Sonnino A, Spiegelman A. Narwhal and Tusk: A DAG-based Mempool and Efficient BFT Consensus. EuroSys. 2022:34-50.