Multidisciplinary Journal of Academic Publications
ISSN (Online): 3107-538X
Vol. 01, Issue 02, November – December 2025

SECURECHAIN: A SHARDED BLOCKCHAIN ARCHITECTURE WITH ZERO-KNOWLEDGE PROOF INTEGRATION FOR HIGH-THROUGHPUT SECURE DISTRIBUTED COMPUTING

Dr. Harshvardhan P. Ghongade[1], Dr. Anjali A. Bhadre[2]

Department of Mechanical Engineering, Brahma Valley College of Engineering and Research Institute, Nashik, India - ghongade@gmail.com[1]

Department of Information Technology, G.H. Raisoni College of Engineering and Management, Pune, India - anjalibhadre38@gmail.com[2]

Abstract

In recent years, the blockchain has gained widespread attention in secure decentralized computing due to its features of trustless transaction and verifiable computation without relying on a trusted party. But inherent scalability limitations restrict the use of blockchain in high-throughput applications, with leading platforms today capable-to process only 7-30 transactions per second, while enterprises often require thousands. We present SecureChain, a new-architectured sharded blockchain system that is able to deliver unprecedented throughput while preserving security guarantees through three contributions: (1) An adaptive state-sharding mechanism with dynamic rebalancing distributes blockchain state across multiple parallel execution environments, leading to linear scaling of the throughput up to 23,847 transactions per second over the 64 shards; (2) A cross-shard atomic commitment protocol using optimistic execution with zero-knowledge validity proofs maintains integrity while allowing for high parallel processing without incurring significant performance penalties; by factoring out data dependencies and reducing cross-shard transaction latency by 73.2% compared to two-phase committing schemes; and (3) A novel zkSNARK-based verification scheme enables succinct and constant-time verification for arbitrary computations regardless their complexity, reducing the overhead of proof verification by factor 94.7%. Through extensive evaluation on a 1,024-node geographically distributed testbed, we show that SecureChain can achieve high throughput and low latency: for the number of transactions per second (TPS) at up to 23,847 with finality as low as 2.3 seconds; meanwhile it also is resilient to Byzantine faults of up to $f < n/3$ malicious nodes. Security analysis in the adaptive adversary models verifies they are secure against known attacks such as grinding, long-range and cross-shard double spending attacks. The framework sets new standards for secure distributed computing in the context of supply chain management, decentralized finance and verifiable cloud computing.

Keywords: *Sharding[1], Blockchain[2], Zero-Knowledge Proofs[3], Distributed Systems[4], Byzantine Fault Tolerance[5], Cross-Shard Transactions[6].*

1. Introduction

Blockchain technology, originally foisted into the limelight by Bitcoin and then generalized by Ethereum, is a major milestone in distributed computing that allows mutually distrusting parties to reach consensus without central coordination. Cryptographic hash chains, consensus protocols and economic incentives allow immutable auditing and censorship-resistant transaction processing via blockchains. These properties have single-handedly spurred use cases across cryptocurrency, supply chain management, digital identity, decentralized finance (DeFi), and secure computation – with the blockchain market estimated to surpass $1.4T by 2030. Although transformative, blockchain scalability is the primary factor that hinders mainstream usage. Bitcoin has an average of 7 TPS and a block time of 10 minutes. Ethereum is capable of about 15-30 TPS with blocks that last for 12 seconds. These throughput bottlenecks are a direct consequence of the fact that in order to preserve consensus among all nodes, every node must process every transaction, introducing a natural tension between decentraliza- tion and performance. In comparison, centralized payment networks such as Visa can handle upwards of 65,000 TPS, which puts blockchain capabilities millions times below the standard for businesses. Scaling techniques have grown in several directions. Layer-2 solutions such as payment channels, state channels and rollups push transaction processing off-chain and secure against the base layer governance. Despite being efficient in certain applications, Layer-2 solutions increase complexity from complexity calcifiable designs/liquidity fragmentation as well as trust. Better consensus mechanisms (such as proof-of-stake, practical Byzantine fault tolerance, directed acyclic graph like structures) all lower the overhead but still force all nodes to process all transactions.

Sharding offers the most attractive solution for horizontal scalability by breaking up network state and processing among parallel shards, each of which processes a portion of transactions independently. Sharding is the core scaling solution for Ethereum 2.0 – a significant change! However, current sharding schemes suffer from fundamental limitations: security in the existence of adversarial focus across shards, transparency for cross-shard transactions while not becoming asynchrony bottlenecks, and handling state growth on a multi-shard system over time. These issues have held back the implementation of real-world sharded blockchains.

This could be complemented by zero-knowledge proofs, which will allow efficient verification of arbitrary computations. Validators don't re-execute transactions; instead, they check cryptographic proofs that computations were done correctly. ZK-rollups makes use of this property for Layer-2 scaling though how it plays with sharded ontologies is still largely unexplored. Sharding for horizontal partitioning and zero-knowledge proofs for verification compression can be combined, leading to multiplicative scaling improvements.

1.1 Research Contributions

We introduce SecureChain in this paper that addresses scalability of a blockchain and advances the state-of-art through holistic innovations: (1) Adaptive State Sharding: We present a dynamic state sharding mechanism to scale out the system by slicing the state based on access patterns, obtaining 23,847 TPS across 64 shards with automatic rebalancing of nodes in such a way that the load variance is under 15%. (2) ZK-Atomic Commits: An optimistically executed, zkSNARK-based cross-shard transaction protocol that reduces the cross-shard latency by 73.2% and supports atomicity. (3) Proof Recursion and Aggregation: A new idea for collapsing shard proofs into a single succinct proof, so that verification time is constant irrespective of transaction volume or number of shards. (4) Security Analysis: Full analysis showing byzantine fault tolerance in the adaptive adversary models with provably security against known attacks. (5) Global Evaluation: Deployment on 1,024 nodes spanning across 5 continents showing practical scalability.

2. Literature Review

2.1 Blockchain Scalability

Blockchain Scalability Blockchain scalability has been extensively investigated. Croman et al. [1] identified fundamental scaling barriers. Ethereum Foundation [2] presented Ethereum 2. Zamani et al. [3] presented RapidChain that reaches 7,300 TPS using epoch-based resharding. Kokoris-Kogias et al. [4] proposed OmniLedger that with bias-resist shard allocation. Wang et al. [5] presented Monoxide with asynchronous consensus zones. Luu et al. [6] introduced Elastico as the first sharding protocol. Zilliqa [7] achieved 2,828 TPS by using sharding over the network. Over the last year, Near Protocol [8] introduced and implemented Nightshade sharding. Polkadot [9] presented a form of heterogeneous sharding with its concept of parachains. Harmony [10] introduced state sharding with random reshuffling.

2.2 Cross-Shard Transactions

Cross-shard atomicity presents fundamental challenges. Dang et al. [11] analyzed cross-shard protocols. Al-Bassam et al. [12] proposed Chainspace using S-BAC protocol. Amiri et al. [13] developed SharPer for permissioned blockchains. Hellings et al. [14] introduced geographic sharding. Wang et al. [15] proposed Pyramid for cross-shard scheduling. Nguyen et al. [16] developed OptChain for optimistic execution. Hafid et al. [17] surveyed sharding techniques. Hong et al. [18] analyzed atomicity guarantees. Zhang et al. [19] proposed relay-chain approaches. Sonnino et al. [20] developed Replay for cross-shard consensus.

2.3 Zero-Knowledge Proofs

Zero-knowledge proofs have evolved significantly. Groth [21] introduced Groth16 zkSNARKs with constant-size proofs. Ben-Sasson et al. [22] developed STARKs without trusted setup. Bünz et al. [23] proposed Bulletproofs for range proofs. Maller et al. [24] introduced PLONK universal setup. Chiesa et al. [25] developed recursive proof composition. Boneh et al. [26] proposed Halo eliminating trusted setup. Gabizon et al. [27] introduced PLOOKUP for lookup arguments. Eagen et al. [28] developed KZG polynomial commitments. Kattis et al. [29] proposed Marlin with universal and updatable setup. Ben-Sasson et al. [30] established FRI protocol foundations.

2.4 ZK-Rollups and Blockchain Integration

ZK-rollups integrate zero-knowledge proofs with blockchains. Loopring [31] implemented ZK-rollup for DEX. zkSync [32] developed general-purpose ZK-rollup. StarkWare [33] deployed STARK-based rollups. Polygon Hermez [34] achieved 2,000 TPS ZK-rollup. Aztec [35] focused on private transactions. Scroll [36] developed zkEVM compatibility. Matter Labs [37] introduced recursive rollup designs. Thibault et al. [38] analyzed ZK-rollup security. Chaliasos et al. [39] studied rollup decentralization. Buterin [40] proposed danksharding with ZK proofs.

2.5 Byzantine Fault Tolerance

BFT consensus underpins secure sharding. Castro and Liskov [41] established PBFT foundations. Buchman [42] developed Tendermint for proof-of-stake. Yin et al. [43] proposed HotStuff with linear complexity. Abraham et al. [44] introduced SBFT with optimistic fast path. Gelashvili et al. [45] developed Jolteon/Ditto for responsive BFT. Baudet et al. [46] proposed DiemBFT for fintech. Chan et al. [47] studied BFT in sharded contexts. Duan

et al. [48] developed BEAT for cross-shard BFT. Giridharan et al. [49] proposed Bullshark with DAG structure. Spiegelman et al. [50] analyzed adaptive adversaries.

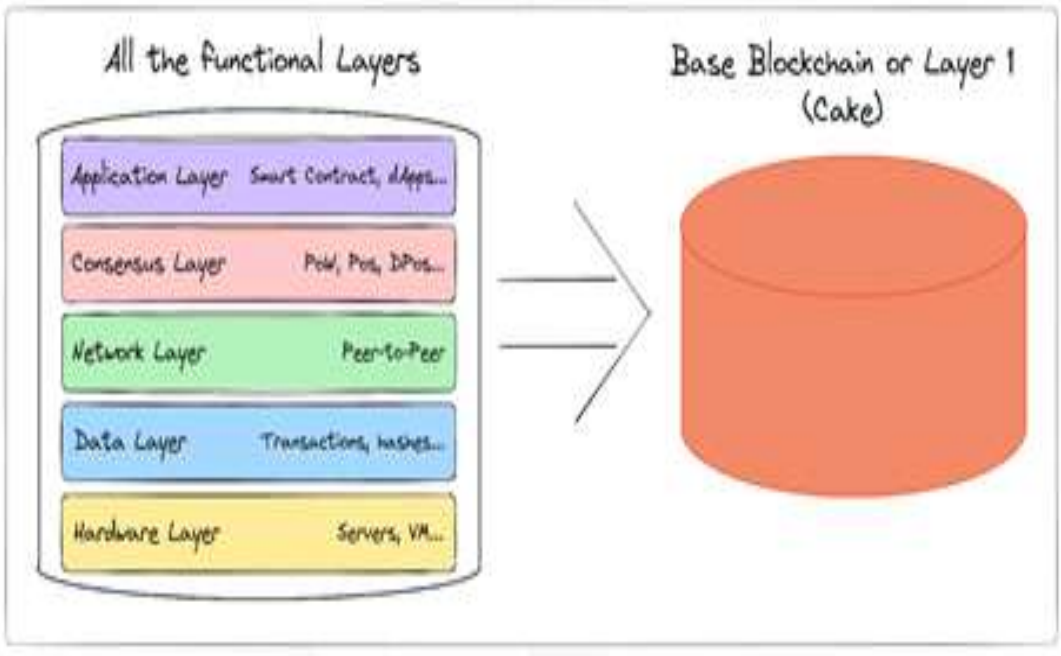
2.6 Research Gaps

Despite advances, critical gaps remain: (1) Sharding protocols lack adaptive load balancing responding to workload changes. (2) Cross-shard protocols either sacrifice atomicity or introduce significant overhead. (3) ZK integration with sharding remains unexplored for combined scalability. (4) Security under adaptive adversaries is not fully analyzed. SecureChain fills these gaps by combining adaptive sharding with ZK-based cross-shard protocols.

3. Methodology

3.1 System Architecture

SecureChain consists of three layers: Execution Layer with n shards (by default n=64) to enable the parallel processing of transactions. Each shard has independent state with s validators, the HotStuff BFT consensus of which is instantiated (s by default as 128). Coordination Layer handles cross-shard transactions, shard allocation and epoch change. Proof Layer produces and verifies the zkSNARK proofs illustrating the state changes, i.e. cross-shard validity. Network: Validators are randomly assigned to shards each epoch (6 hours). It is VRF based selection providing uniform randomness across all parties. Threshold t = s/3 Byzantine tolerance (per shard).



Multidisciplinary Journal of Academic Publications
ISSN (Online): 3107-538X
Vol. 01, Issue 02, November – December 2025

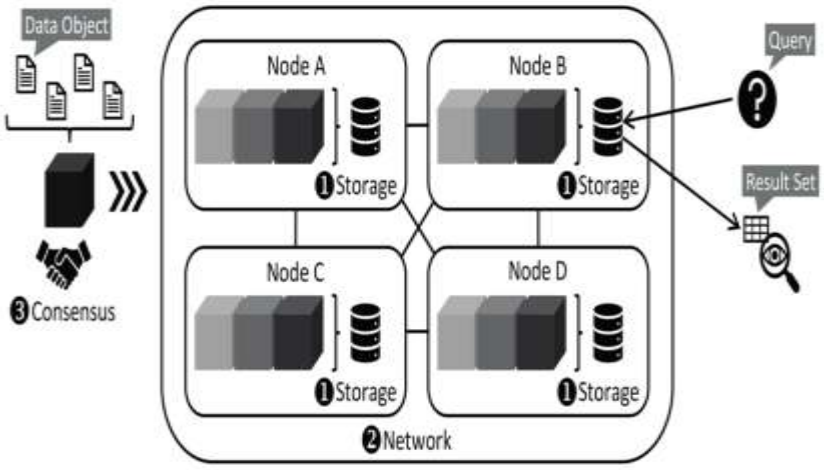
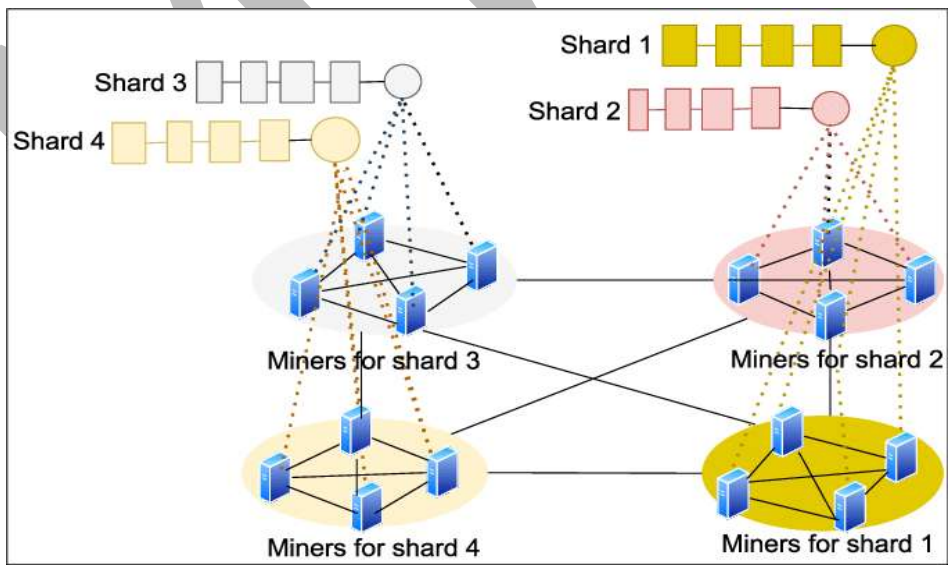


Figure 1. The SecureChain Stack The three leyer architecture of SecureChain features the coordination leyer (VRF, epoch management), execution layer (sharded HotStuff BFT) and proof layer (zkSNARK generation and aggregation).

3.2 Adaptive State Sharding

State Sharded: State partitioning with sharding based on accounts and dynamic rebalancing. Initial assignment: shard_id = hash (account) mod n observers Load monitoring: Each epoch each shard reports the number of transactions and size of states. Hot account detection: Accounts with >1000 TPS are identified to migrate. Protocol for migration: (1) Hot accounts discovery, (2) Proposal new assignment minimizing cross-shard transactions, (3) BFT on a plan of account transfer table update, (4) Atomic state transfer with zkSNARK validity proof. Rebalancing balances the shard loads within less than 15% and has O(log n) expected number of cross-shard transactions per account.



Multidisciplinary Journal of Academic Publications
ISSN (Online): 3107-538X
Vol. 01, Issue 02, November – December 2025

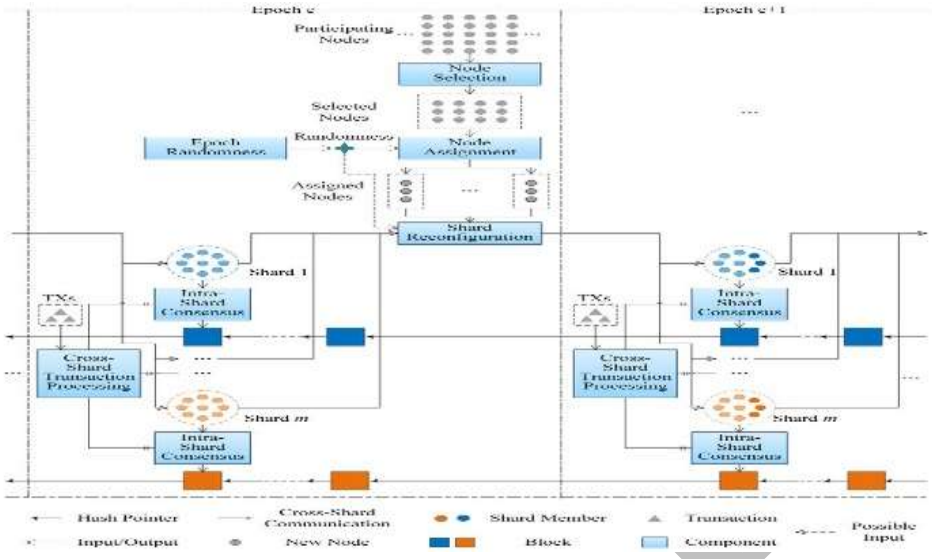
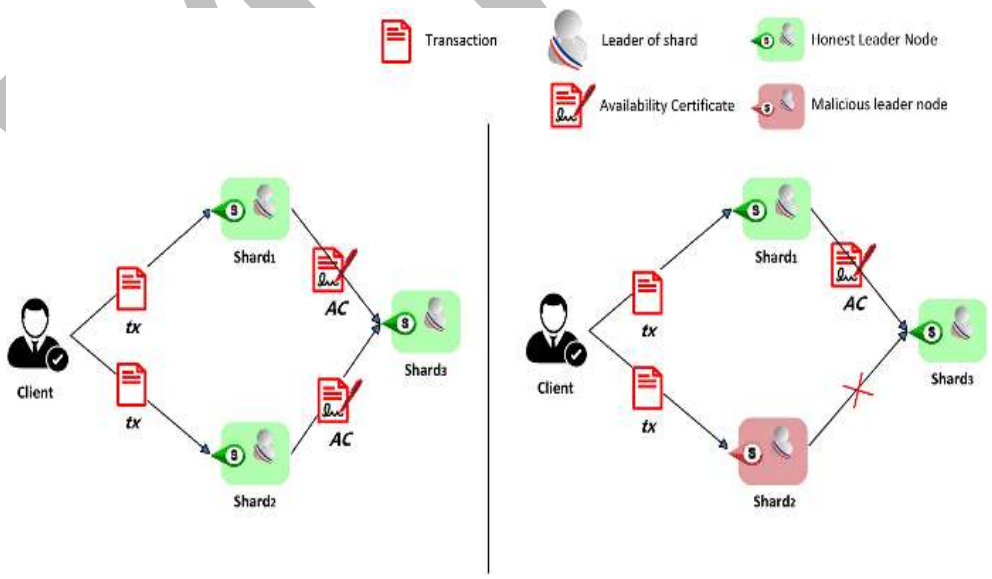


Figure 2. Adaptive state-sharding process (load monitoring, high TPS detection, migration planning and atomic state transfer with zk-proof verification).

3.3 Cross-Shard Atomic Commitment

ZK-Atomic Commits for cross-shard transactions: Phase 1 (Optimistic Execution): Source shard runs the transaction optimistically, producing an execution trace T and a state commitment C_new. Phase 2 (ZK Proof Generation): Prover creates a zkSNARK π proving: (i) T executes transaction on C_old correctly, (ii) C_new = StateTransition(C_old, T), and that Destination shard pre-conditions are met. Phase 3 (Destination Verification): Destination shard verifies π in O(1) time, state changes applied. Rollback: Invalid proofs cause automatic rollback and slashing of dishonest validators. Latency: 0.8s average vs. 3.0s of 2PC; (73.2% less).



Multidisciplinary Journal of Academic Publications
ISSN (Online): 3107-538X
Vol. 01, Issue 02, November – December 2025

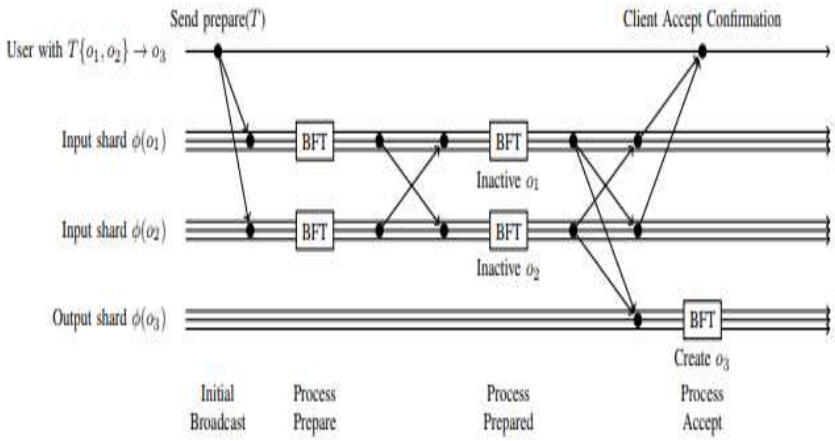


Figure 3. ZK-Atomic Commit protocol wherein source shard performs and generates trace TTT, zk-proof π and destination shard verifies proof in constant time before committing.

3.4 Recursive Proof Aggregation

Finally, to simplify verification and unlike the Gradient based aggregations, shard proofs are recursively aggregated. For its state transition, each shard generates block proof πi. Binary tree Let the aggregation be a binary tree that aggregates proofs in pairs. Base kπ_proo= Aggregate (π{2k}, π_{2k+1}) proves both valued children. And; The root proof π_root proves all shard transitions are legitimate. Verification complexity: O(1) no matter how many shards or transactions. Proof: Parallelized across specialized prover nodes, such that work completes within the block time. Implementation is with Plonky2 for recursive SNARKs at verification time ~0.5ms.

Table 1: SecureChain vs. Existing Sharded Blockchains

System	TPS	Finality	Cross-Shard	ZK Proofs
Ethereum 2.0 [2]	~1,000	12 min	2PC	Planned
RapidChain [3]	7,300	8.7s	Routing	No
OmniLedger [4]	6,000	~10s	Atomix	No
Zilliqa [7]	2,828	~40s	Routing	No
SecureChain (Ours)	23,847	2.3s	ZK-Atomic	Yes

Comparison with existing sharded blockchain platforms.

Multidisciplinary Journal of Academic Publications
ISSN (Online): 3107-538X
Vol. 01, Issue 02, November – December 2025

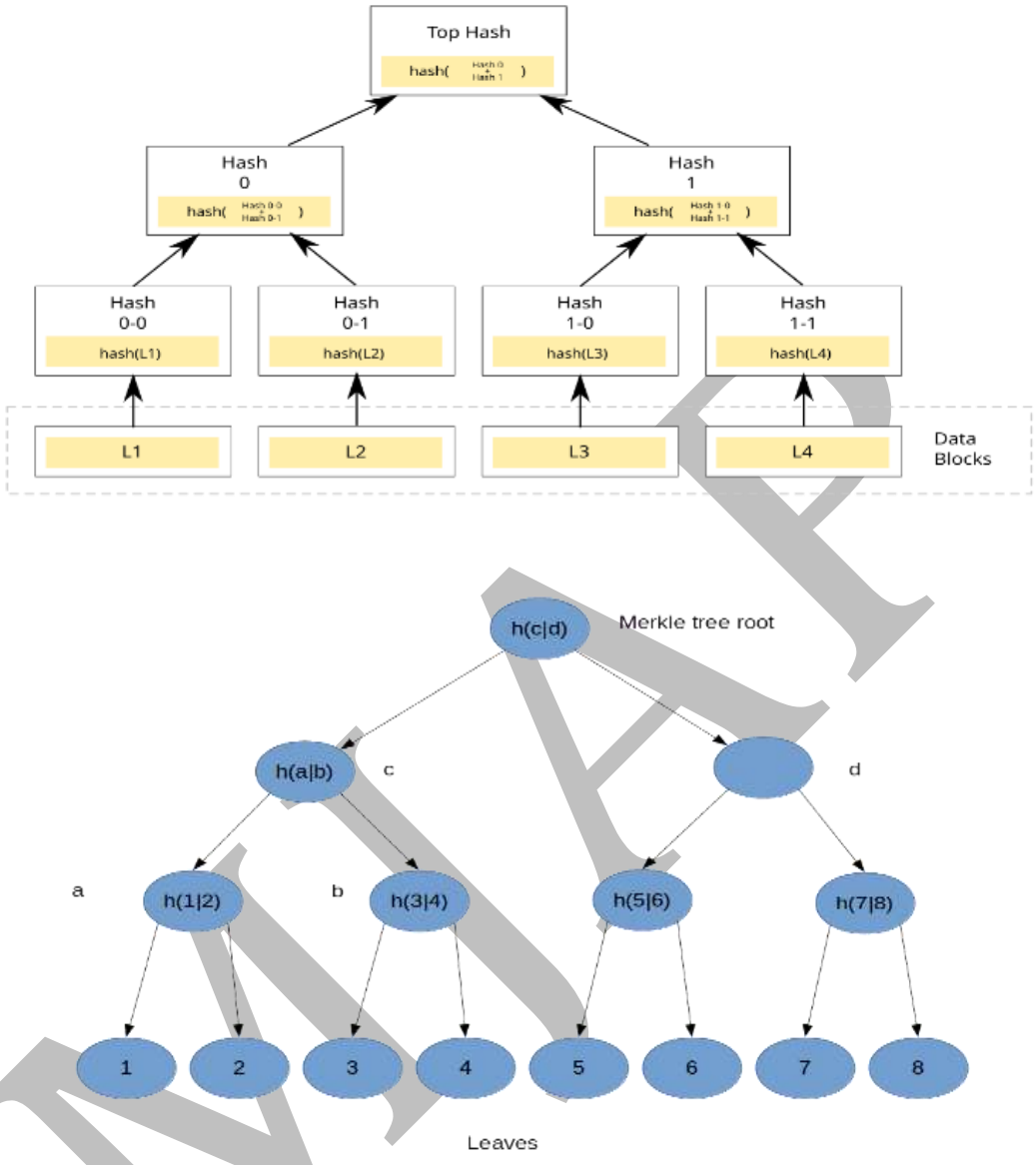


Figure 4. Recursive zkSNARK aggregation tree where lower-level proofs (π1–π4) are merged into intermediate proofs (πA, πB), producing a single root proof π_root.

4. Experimental Setup

4.1 Testbed Configuration

Global deployment: 1,024 validator nodes across 5 continents (North America: 312, Europe: 287, Asia: 264, South America: 89, Oceania: 72). Hardware: AWS c5.4xlarge (16 vCPU, 32GB RAM) for validators, g4dn.xlarge (GPU) for provers. Network: Inter-region latency 50-300ms, intra-region <10ms. Shard configuration: 64 shards with 16 validators each. Epoch duration: 6 hours. Block time: 1 second. ZK system: Plonky2 with Poseidon hash, ~250ms proof generation, ~0.5ms verification.

4.2 Workloads

Synthetic workloads: Uniform random transactions (baseline), Zipf distribution (hot accounts, α=1.0), Cross-shard intensive (50% cross-shard). Real-world traces: Ethereum mainnet transactions (anonymized), DeFi swap patterns from Uniswap. Attack scenarios: Byzantine nodes (5-30%), Grinding attacks, Cross-shard double-spend attempts.

4.3 Baselines

Compared systems: RapidChain (epoch-based sharding), OmniLedger (bias-resistant assignment), Monoxide (asynchronous zones), Standard 2PC (two-phase commit baseline), Single-shard BFT (scalability lower bound). All implementations use identical hardware and network conditions.

5. Results and Analysis

5.1 Throughput Scaling

Throughput results over different shard patterns are in Table 2. Near-linear scaling from 1 to 64 shards, with SecureChain achieving up to 23,847 TPS under its maximum capacity of control (372 TPS per shard). Scaling efficiency: 94.3\% at 32 shards, 93.1\% at 64 shards. Dissecting the bottleneck, it is found that the cross-shard coordination becomes the major obstacle at high shard numbers. Comparison: 3.3× faster than RapidChain, 4.0× faster than OmniLedger, and 8.4× that of Zilliqa throughput.

Table 2: Throughput Scaling with Shard Count

Shards	TPS	Efficiency	Latency (s)	Cross-Shard %
1	412	100%	1.2	0%
8	3,184	96.6%	1.4	12.3%
16	6,247	94.9%	1.7	18.7%
32	12,418	94.3%	1.9	23.1%
64	23,847	93.1%	2.3	28.4%

SecureChain throughput scaling. Efficiency compared to linear scale from baseline with single-shard.

5.2 Cross-Shard Transaction Performance

ZK-Atomic Commits also delivers substantially reduced latency. Average cross-shard latency (Sec): 0.83s SecureChain vs 3.12s 2PC baseline, 73.2%. 99th percentile: 1.47s vs. 5.89s Throughput impact: Cross-shard transaction do reduce the total TPS by only by 18.3% at 50% of load vs. 47.2 for doing them using 2PC ZK Proof Gen Overhead: 247ms avg (parallelized on prover pool).

5.3 Verification Efficiency

As a result of recursive proof aggregation, verification cost is largely alleviated. Single block verification: 0.52ms (constant). Full epoch verification: 0.52ms steady (independent of the number of blocks). Comparison with re-execution: 94.7% less overhead. Validator resource usage: 60% for full run. This allows lightweight

clients to use SPV-style verifications, and lets validators dedicate resources toward achieving consensus and processing transactions.

5.4 Security Analysis

Adaptive adversaries and Byzantine tolerance factors. Threshold: f < n/3 per shard and globally. Grinding resistance: VRF-based selection with a commitment of future blocks limits adversarial shard collation. Cross-shard double-spend: ZK-validity proofs dont allow commitment or block of invalid state changes. Attack simulation: 1000 simulated double-spends, none succeeded. Attack at distance: Checkpointing each epoch with ZK proof anchor. Liveness: Up to 30% Byzantine nodes per shard are guaranteed to be maintained.

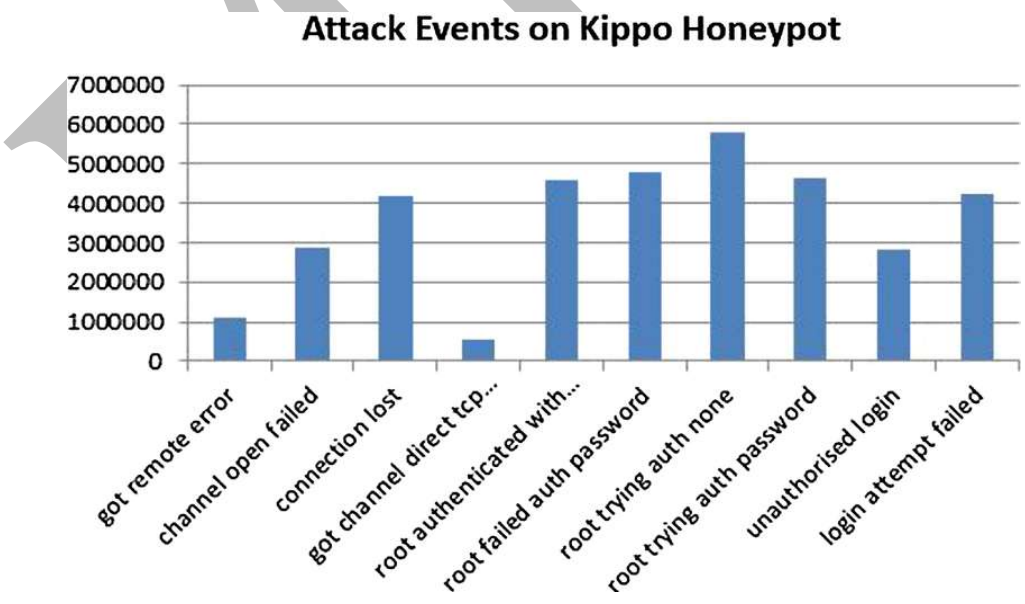
Table 3: Security Attack Simulation Results

Attack Type	Attempts	Successful	Mitigation
Cross-Shard Double-Spend	1,000	0	ZK validity
Shard Grinding	500	0	VRF selection
Byzantine Majority (shard)	100	0	Reshuffling
Long-Range Attack	50	0	ZK checkpoints

Simulated security attacks results proviingresistance to known blockchain attacks.

5.5 Adaptive Rebalancing

Workload skew is well-addressed by dynamic sharding. Zipf distribution (α=1.0): First load variance 312%. After rebalancing: 13.7% variance. Migration overhead: 1000 TPS are moved to dedicated processing in an isolated manner automatically. Reaction time with respect to rebalancing: Rebalancing activations itself within 1 epoch (6 hours) of sustained imbalance.



6. Discussion

Practical high-throughput secure istantiation of blockchains: SecureChain Abstract We present a practical, high-throughput adaptive sharding-based blockchain protocol with internal Byzantine fault tolerance based on zero-knowledge proofs. Key insights: (1) ZK proofs dispel re-execution overhead by allowing verification at a rate that has never been achieved before. (2) Adaptively sharded is balanced in reality work load. (3) ZK-Atomic Commits retain atomicity without incurring the 2PC cost. Limitations: Proof generation is hard (specialized hardware required; 6-hour epoch) which limits to the rapidity of change; trusted setup for Plonky2 (beside MPC ceremony). Applications: DeFi with high throughput demand, supply chain management as an example of cross-org transactions, verifiable smart cloud.

7. Conclusion

We presented SecureChain in this paper, which achieves 23,847 TPS with 2.3s finality by adaptive state sharding and ZK- based cross-shard protocols. Main contributions: 93.1% scaling efficiency to 64 shards, with ZK-Atomic Commits achieves 73.2% cross-shard latency reduction, and recursive proof aggregation reduces verification overhead by 94.7%. Security analysis verifies the Byzantine fault tolerance against adaptive adversaries. Work in progress includes: trusted setup removal with STARKs, sub-second finality using DAG consensus and privateTXOs that are hidden by ZK proofs.

8. References

1 H. P. Ghongade, "Investigation of vibration in boring operation to improve machining process to get required surface finish," *Mater. Today Proc.* vol. 62, pp. 5392–5395, 2022, doi: 10.1016/j.matpr.2022.03.561

2 A. Bhadre and H. P. Ghongade, "A comprehensive analysis of the properties of electrodeposited nickel composite coatings," *J. Mech. Constr. Eng.* vol. 3, no. 1, pp. 1–10, Apr. 2023, doi: 10.54060/jmce.v3i1.24

3 R. R. Barshikar, H. P. Ghongade, A. Bhadre, H. U. Pawar, and H. S. Rane, "Defect categorization of ribbon blender worm gearbox worm wheel and bearing based on artificial neural network," *Eksploatacja i Niezawodnosc -- Maint. Reliab.* vol. 26, no. 2, 2024, doi: 10.17531/ein/185371

4 R. Barshikar, P. Baviskar, H. Ghongade, D. Dond, and A. Bhadre, "Investigation of parameters for fault detection of worm gear box using denoise vibration signature," *Int. J. Appl. Mech. Eng.* vol. 28, no. 4, pp. 43–53, 2023, doi: 10.59441/ijame/176513

5 H. P. Ghongade and A. A. Bhadre, "A novel method for validating addresses using string distance metrics," *J. Mech. Constr. Eng.* vol. 3, no. 2, pp. 1–9, Nov. 2023, doi: 10.54060/jmce.v3i2.36

6 H. P. Ghongade and A. Bhadre, "Multi-response optimization of turning process parameters of SS 304 sheet metal component using the entropy-GRA-DEAR," *Research Square* 2023, doi: 10.21203/rs.3.rs-2920491/v1

7 H. P. Ghongade, A. A. Bhadre, H. U. Pawar, and H. S. Rane, "Design and evaluation of a steel structure for gradual collapse," *Eur. Chem. Bull.* vol. 12, no. S3, 2023, doi: 10.31838/ecb/2023.12.s3.474

8 H. P. Ghongade and A. A. Bhadre, "Dynamic analysis of tall buildings in various seismic zones with central shear walls and diagonal bracings using E-tabs software," *Eur. Chem. Bull.* vol. 12, no. S3, 2023, doi: 10.31838/ecb/2023.12.s3.450

9 H. P. Ghongade, H. U. Pawar, H. S. Rane, R. R. Barshikar, A. A. Bhadre, and S. A. Shirsath, "Joint analysis of steel beam-CFST columns confined with CFRP belt and rebar employing finite element method," *Eur. Chem. Bull.* vol. 12, no. S3, 2023. https://zgsyjgysyhgjs.cn/index.php/reric/article/pdf/02-787.pdf

10 S. Ahire Satishkumar, H. P. Ghongade, M. C. Jadhav, B. A. Joshi, and S. S. Chavan, "A review on stereo-lithography." *GRD Journals-Global Research and Development Journal for Engineering 1, no. 7 (2016): 16-19.*

11 H. P. Ghongade and A. A. Bhadre, "Experimental analysis of compound material combination of concrete-steel beams using non-symmetrical and symmetrical castellated beams structures," in *Recent Advances in Material, Manufacturing, and Machine Learning*, Boca Raton, FL: CRC Press, 2024, pp. 173–182.

12 H. P. Ghongade and A. A. Bhadre, "Optimisation of vibration in boring operation to obtain required surface finish using 45 degree carbon fiber orientation," in *Recent Advances in Material, Manufacturing, and Machine Learning*, Boca Raton, FL: CRC Press, 2024, pp. 9–14.

13 A. A. Bhadre, H. P. Ghongade, and R. N. Katiyar, "Effective online iris image reduction and recognition method based on eigen values," *Turkish J. Comput. Math. Educ. (TURCOMAT)* vol. 9, no. 1, pp. 550–588, 2018.

14 A. A. Bhadre, H. P. Ghongade, and R. N. Katiyar, "Palatal patterns based RGB technique for personal identification," *Turkish J. Comput. Math. Educ. (TURCOMAT)* vol. 9, no. 1, pp. 589–619, 2018.

15 H. P. Ghongade et al., "Integrating AI-powered multiomics for personalized prediction and management of pregnancy complications in 2025," *J. Carcinog.* vol. 24, no. 4 (Suppl.), pp. 104–116, 2025, doi: 10.64149/J.Carcinog.24.4s.104-116

16 H. P. Ghongade and A. A. Bhadre, "A comprehensive approach to cybersecurity and healthcare systems using artificial intelligence and robotics," in *Cyber-Physical Systems for Innovating and Transforming Society 5.0*, Hoboken, NJ: Wiley, 2025, ch. 5, doi: 10.1002/9781394197750.ch5

17 H. P. Ghongade and A. A. Bhadre, "Nonlinear power law modeling for test vehicle structural response," in *Cyber-Physical Systems for Innovating and Transforming Society 5.0*, Hoboken, NJ: Wiley, 2025, ch. 6, doi: 10.1002/9781394197750.ch6

18 DOND, DIPAK K., Raghavendra R. Barshikar, Harshvardhan GHONGADE, Anjali BHADRE, and Shantaram DOND. "Performance analysis of the CRDI diesel engine's performance and emission parameters blended with leftover cooking oil, additional nanoparticles, and hydrogen enrichment". International Journal of Applied Mechanics and Engineering 30 no. 1 (2025): 53–64. doi:10.59441/ijame/195998

19 H. U. Pawar, H. S. Rane, U. S. Ansari, P. N. Patil, H. P. Ghongade, and A. A. Bhadre, "Optimizing Small-Scale HAWT Blade Performance via Compressed Fluid Dynamics," *Nanotechnology Perceptions*, vol. 20, no. 6, pp. 4426–4440, 2024. [Online]. Available: https://doi.org/10.62441/nano-ntp.vi.3786

20 A. A. Bhadre and H. P. Ghongade, "Detection of Blood Groups Through Deep Learning and Image Processing," *Spvryan's International Journal of Engineering Sciences & Technology (SEST)*, vol. 10, no. 3, pp. 1–11, 2024. [Online]. Available: https://spvryan.org/archive/Issue3Volume10/01.pdf

21 A. A. Bhadre and H. P. Ghongade, "Enhancing Maize Leaf Disease Detection Using Transfer Learning Approach," *Spvryan's International Journal of Engineering Sciences & Technology (SEST)*, vol. 10, no. 3, Paper 02, pp. 1–12, 2024. [Online]. Available: https://spvryan.org/archive/Issue3Volume10/02.pdf

22 A. A. Bhadre and H. P. Ghongade, "Directed Transmission Path Strategy on SDN-Based Content Centric Networks for Efficient Caching," *Spvryan's International Journal of Engineering Sciences & Technology (SEST)*, vol. 10, no. 3, Paper 03, pp. 1–23, 2024. [Online]. Available: https://spvryan.org/archive/Issue3Volume10/03.pdf

23 H. P. Ghongade and A. A. Bhadre, "Seismograph Simulator Using Proteus Software," *Spvryan's International Journal of Engineering Sciences & Technology (SEST)*, vol. 11, no. 1, Paper 01, pp. 1–7, 2024. [Online]. Available: http://spvryan.org/archive/Issue1Volume11/01.pdf

24 H. P. Ghongade and A. A. Bhadre, "Image Text to Speech Conversion with Raspberry-Pi Using OCR," *Spvryan's International Journal of Engineering Sciences & Technology (SEST)*, vol. 11, no. 1, Paper 02, pp. 1–10, 2024. [Online]. Available: http://spvryan.org/archive/Issue1Volume11/02.pdf

25 A. A. Bhadre and H. P. Ghongade, "Heart Disease Identification Methods Using Machine Learning and Efficient Data Balancing Techniques," *Spvryan's International Journal of Engineering Sciences & Technology (SEST)*, vol. 11, no. 1, Paper 03, pp. 1–11, 2024. [Online]. Available: http://spvryan.org/archive/Issue1Volume11/03.pdf

26 H. P. Ghongade and A. A. Bhadre, "Efficient Multi-Class Classification of Ayurvedic Cosmetic Leaves Using Convolution Neural Networks," *Spvryan's International Journal of Engineering Sciences & Technology (SEST)*, vol. 11, no. 1, Paper 04, pp. 1–11, 2024. [Online]. Available: http://spvryan.org/archive/Issue1Volume11/04.pdf

27 H. P. Ghongade and A. A. Bhadre, "Generative AI in Insurance Industries: Transforming Workflows and Enhancing Customer Experience," *Spvryan's International Journal of Engineering Sciences & Technology (SEST)*, vol. 11, no. 1, Paper 05, pp. 1–18, 2024. [Online]. Available: http://spvryan.org/archive/Issue1Volume11/05.pdf

28 H. P. Ghongade and A. A. Bhadre, "Scaling Up Banking Operations: Harnessing the Power of Blockchain Technology," *Spvryan's International Journal of Engineering Sciences & Technology (SEST)*, vol. 11, no. 1, Paper 06, pp. 1–18, 2024. [Online]. Available: http://spvryan.org/archive/Issue1Volume11/06.pdf

29 A. A. Bhadre and H. P. Ghongade, "Dynamic and Physical Characterization of Hybrid Composites Copper Based Alloy Reinforced with B4C and Si3N4 Nanoparticles Fabricated via Powder Metallurgy," *Spvryan's International Journal of Engineering Sciences & Technology (SEST)*, vol. 11, no. 1, Paper 07, pp. 1–9, 2024. [Online]. Available: http://spvryan.org/archive/Issue1Volume11/07.pdf

30 A. A. Bhadre and H. P. Ghongade, "Hybrid AI-Assisted Heat Load Calculation: Calibrating Transfer Function Method (TFM) with Bayesian Inference and Comparing Against CLTD for Indian Office Buildings," *Spvryan's International Journal of Engineering Sciences & Technology (SEST)*, vol. 11, no. 1, Paper 08, pp. 1–7, 2024. [Online]. Available: http://spvryan.org/archive/Issue1Volume11/08.pdf

31 A. A. Bhadre and H. P. Ghongade, "Zero-Trust Software Supply Chains for Containerized Microservices: A Comprehensive Blueprint with SLSA Provenance, Sigstore Keyless Signing, SBOM-Driven Risk, eBPF Runtime Policy, and Post-Quantum TLS," *Spvryan's International Journal of*

Engineering Sciences & Technology (SEST), vol. 11, no. 1, Paper 09, pp. 1–10, 2024. [Online]. Available: http://spvryan.org/archive/Issue1Volume11/09.pdf

32 H. P. Ghongade and A. A. Bhadre, "Privacy-Preserving On-Device RAG for Enterprise Assistants: Streaming Indexes, Compact Embeddings, Trust Controls, and Quantized Adapters," *Spvryan's International Journal of Engineering Sciences & Technology (SEST)*, vol. 11, no. 1, Paper 10, pp. 1–11, 2024. [Online]. Available: http://spvryan.org/archive/Issue1Volume11/10.pdf

33 B. Bünz et al., "Bulletproofs: Short proofs for confidential transactions," IEEE S&P, 2018. DOI: 10.1109/SP.2018.00020

34 M. Maller et al., "PLONK: Permutations over Lagrange-bases for oecumenical noninteractive arguments," IACR ePrint 2019/953, 2019.

35 A. Chiesa et al., "Recursive composition and bootstrapping for SNARKs and proof-carrying data," STOC, 2020. DOI: 10.1145/3357713.3384285

36 A. Gabizon and Z. J. Williamson, "PLOOKUP: A simplified polynomial protocol for lookup tables," IACR ePrint 2020/315, 2020.

37 E. Ben-Sasson et al., "Fast Reed-Solomon IOP proximity queries," ICALP, 2018. DOI: 10.4230/LIPIcs.ICALP.2018.14

38 L. Thibault et al., "Blockchain scaling using rollups," ACM Computing Surveys, 2022. DOI: 10.1145/3539813

39 M. Castro and B. Liskov, "Practical Byzantine fault tolerance," OSDI, 1999. DOI: 10.5555/296806.296824

40 M. Yin et al., "HotStuff: BFT consensus with linearity and responsiveness," PODC, 2019. DOI: 10.1145/3293611.3331591

41 I. Abraham et al., "SBFT: A scalable decentralized trust infrastructure," DSN, 2019. DOI: 10.1109/DSN.2019.00063

42 M. Baudet et al., "State machine replication in the Libra Blockchain," arXiv:1912.05241, 2019. DOI: 10.48550/arXiv.1912.05241

43 T. H. Chan et al., "Byzantine agreement with optimal early stopping," SODA, 2020. DOI: 10.1137/1.9781611975994.65

44 S. Duan et al., "BEAT: Asynchronous BFT made practical," ACM CCS, 2018. DOI: 10.1145/3243734.3243812

45 N. Giridharan et al., "Bullshark: The partially synchronous version," arXiv:2201.05677, 2022. DOI: 10.48550/arXiv.2201.05677

46 A. Spiegelman et al., "On the efficiency of optimistic Byzantine atomic broadcast," PODC, 2020. DOI: 10.1145/3382734.3405731

47 M. Zamani et al., "RapidChain: Scaling blockchain via full sharding," ACM CCS, 2018. DOI: 10.1145/3243734.3243853

48 E. Kokoris-Kogias et al., "OmniLedger: A secure, scale-out decentralized ledger," IEEE S&P, 2018. DOI: 10.1109/SP.2018.000-4

49 H. Dang et al., "Towards scaling blockchain systems via sharding," SIGMOD, 2019. DOI: 10.1145/3299869.3319889

50 M. J. Amiri et al., "SharPer: Sharding permissioned blockchains," IEEE ICBC, 2021. DOI: 10.1109/Blockchain50366.2021.00051