

SMICP: A Scalable Merkle-Anchored Interblockchain Communication Protocol

*Thesis submitted in partial fulfillment of the
requirements for the degree*

of

Bachelor of Science in Computer Science and Engineering

by

**Ehasan Ahmed
203014007**

Under the supervision of

Nafees Mansoor, PhD



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
UNIVERSITY OF LIBERAL ARTS BANGLADESH**

SUMMER 2025

© *Ehasan Ahmed*
All rights reserved

DECLARATION

Project Title	SMICP: A Scalable Merkle-Anchored Interblockchain Communication Protocol
Author	<i>Ehasan Ahmed</i>
Student ID	203014007
Supervisor	Nafees Mansoor, PhD

I declare that this thesis entitled *SMICP: A Scalable Merkle-Anchored Interblockchain Communication Protocol* is the result of my own work except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Ehasan Ahmed
203014007

Department of Computer Science and Engineering
University of Liberal Arts Bangladesh

Date: August 21, 2025



Department of Computer Science and Engineering
University of Liberal Arts Bangladesh
Mohammadpur, Dhaka - 1207

CERTIFICATE

This is to certify that the thesis entitled **SMICP: A Scalable Merkle-Anchored Interblockchain Communication Protocol**, submitted by **Ehasan Ahmed** (Student ID: 203014007) an undergraduate student of the **Department of Computer Science and Engineering** has been examined. Upon recommendation by the examination committee, we hereby accord our approval of it as the presented work and submitted report fulfill the requirements for its acceptance in partial fulfillment for the degree of *Bachelor of Science in Computer Science and Engineering*.

Nafees Mansoor, PhD
Associate Professor

Department of Computer Science and Engineering
University of Liberal Arts Bangladesh

Prof. Muhammad Golam Kibria, PhD, SMIEEE
Professor and Head (Acting)

Department of Computer Science and Engineering
University of Liberal Arts Bangladesh

Place: Dhaka

Date: August 21, 2025

ACKNOWLEDGEMENTS

I would like to express my deep and sincere gratitude to my research supervisor, *Nafees Mansoor, PhD*, for giving me the opportunity to do research and providing invaluable guidance throughout this work. His dynamism, vision, sincerity and motivation have deeply inspired me. He has taught me the methodology to carry out the work and to present the works as clearly as possible. It was a great privilege and honor to work and study under his guidance.

I am greatly indebted to my honorable teachers of the Department of Computer Science and Engineering at the University of Liberal Arts Bangladesh who taught me during the course of my study. Without any doubt, their teaching and guidance have completely transformed me to a person that I am today.

I am extremely thankful to my parents for their unconditional love, endless prayers, caring and immense sacrifices for educating and preparing me for my future. I would like to say thanks to my friends and relatives for their kind support and care.

Finally, I would like to thank all the people who have supported me to complete the project work directly or indirectly.

Ehasan Ahmed

University of Liberal Arts Bangladesh

Date: August 21, 2025

Dedicated to
*my parents, with love,
gratitude, and respect beyond
words.* – *Ehasan Ahmed*

ABSTRACT

SMICP, the Scalable Merkle-Anchored Interblockchain Communication Protocol, was designed to support secure, atomic, and privacy-preserving interaction between permissioned blockchain networks. By using a hybrid multi-chain design with a mainchain anchoring cryptographic commitments and multiple domain-specific designed sidechains that can be optimised towards specific application requirements, SMICP allows high-throughput cross-chain transactions and lightning fast proof verification (sub-millisecond), making it suitable as a platform to penetrate into mass adoptions. It combines Merkle tree based batching and an efficient two-phase commit protocol to support both atomicity and fault resilience of transactions in decentralized systems. Testing assessments show scale, performance, and robust security assurances, such that SMICP is ready to be scaled to enterprise-level multi-party blockchain interoperability.

Keywords: Interblockchain Communication, Merkle Tree, Two-Phase Commit, Blockchain Scalability, Permissioned Blockchain

Contents

1	Introduction	1
1.1	Challenges in Blockchain Interoperability	2
1.2	Aim & Objective	2
2	Literature Review and Analysis	4
2.1	Scalability Solutions in Blockchain	4
2.2	Interblockchain Communication	6
2.3	Scalability in Inter Blockchain Communication	9
2.4	Literature Analysis	12
3	Proposed Methodology	14
3.1	Core Components	14
3.2	Merkle Tree Implementation and Cryptographic Foundations	15
3.2.1	Transaction Batching Process	16
3.3	Cross-Chain Anchoring and Verification Protocol	17
3.3.1	Anchoring Protocol	17
3.3.2	Cross-Chain Verification Process	17
3.4	Two-Phase Commit Protocol Implementation	19
3.4.1	Protocol Phases	19
3.4.2	Fault Tolerance and Recovery	21
3.5	Scalability Design and Theoretical Analysis	21
3.6	Implementation Architecture and Chaincode Design	22
3.6.1	Chaincode Components	22
3.6.2	Gateway Architecture	23
3.7	SMICP Protocol Workflow: Multi-Party Transaction Coordination . .	24
3.7.1	Transaction Initiation and Batch Processing	24

3.7.2	Cross-Chain Verification and Coordination	24
3.7.3	Protocol Completion and State Finalization	25
3.8	Security Analysis and Threat Mitigation	25
3.8.1	Cryptographic Security	25
3.8.2	Protocol-Level Security	26
3.8.3	Network Security	26
4	Experimental Results	27
4.1	Batch Processing Performance	27
4.2	Multi-Chain Coordination	31
4.3	Security Validation	33
4.4	Comparative Analysis	34
4.4.1	Existing Work Comparison	34
4.4.2	Scalability Analysis	34
4.5	Limitations and Future Work	35
5	Conclusion	36
	Bibliography	37

List of Figures

3.1	Multi-Chain Layer Based Architecture	15
3.2	Merkle Tree Construction and Proof Generation Process	16
3.3	Cross-Chain Verification Protocol Flow	18
3.4	SMICP Two-Phase Commit Protocol with Fault Recovery	20
3.5	Implementation Architecture with Chaincode Components and Gate- way Layer	23
4.1	Scalability Analysis: Volume vs Execution Time	29
4.2	Throughput Performance: Volume vs Throughput	29
4.3	Anchoring Overhead Analysis: Volume vs Anchoring Time	30
4.4	Processing Time Variation	32
4.5	Network Latency Analysis	32
4.6	Multi-Chain Performance Comparison	33

List of Tables

4.1	SMICP Performance Results by Batch Size	28
4.2	Multi-Chain Coordination Results	31
4.3	Security Validation Test Results	33

Chapter 1

Introduction

Blockchain systems have transformed the secure and transparent creation of immutable data recording in individual systems. Nonetheless, existing blockchain networks have many instances of operating independently of one another, which does not maximize options to cross-chain interoperability demanded by the contemporary decentralized environment. Lack of scalability in heterogeneous blockchains compromises scalability, usability, and real-world adoption due to the inability to coordinate with each other well.

The interblockchain communication (IBC) has consequently become an important research focus and engineering so that multiple blockchains can easily connect, interact in an atomic and secure manner. Multi-domain blockchain ecosystems, e.g., in finance, IoT, enterprise partnerships, require solutions that provide data integrity, minimise verification overhead at a high throughput/low latency, and that also protect the privacy of the entities and preserve the integrity of sensitive data.

This thesis is on the Scalable Merkle-Anchored Interblockchain Communication Protocol (SMICP), a permissioned multichain framework that supports reliable, atomic cross-chain actions. SMICP uses a Mainchain to anchor cryptographic Merkle root summaries of sidechain block data and apply a Two-Phase Commit protocol in order to maintain transaction atomicity concerning permissioned chains. The design is focused on efficiency, scalability under concurrency, and the ability to preserve privacy by trading lightweight proofs rather than raw data.

Targeting such a flexible communication gateway between blockchains as opposed to a particular domain, e.g., banking, means that SMICP can offer a generalized blockchain cross-communication protocol that employs a wide set of industrial and non-industrial use cases.

1.1 Challenges in Blockchain Interoperability

The current blockchain networks tend to be single-functioning, closed environments with no unified ways of sharing and exchanging information with and managing transactions on various chains or inter-chain coordination. Such fragmentation poses serious impediments, such as an inability to integrate smoothly, because of incompatible transaction models and consensus protocols. Chain coordination of transactions is likewise troublesome in ensuring similar changes of state among multiple blockchains, with the risk of partial breakdown and inconsistency in data.

Public blockchains are characteristically high-latency and low-throughput, which partially exacerbates the challenge of scalability given concurrency in cross-chain interactions. Also, the lack of privacy of full transaction details across chains leads to the need of special selective disclosure mechanisms. Some current solutions may require large data replication or trusted intermediaries, which negate system-wide security and trust.

An Inter-Blockchain Communication (IBC) protocol that addresses these shortcomings is required to be tamper-evident, high-performance communication, be able to manage atomic cross-chain transactions, scale effectively during concurrent communication, and guarantee privacy via cryptographic proofs as well as operate effectively in permissioned blockchain settings with energy-efficient consensus protocols, such as Proof of Authority.

1.2 Aim & Objective

This thesis proposes to design, implement, and test the Scalable Merkle-Anchored Interblockchain Communication Protocol (SMICP) in order to support the secure, atomic, privacy-preserving cross-blockchain communication between permissioned blockchains. The goal is to create a modular multichain infrastructure with a Mainchain anchoring Merkle root summaries of side chain states and introduce a Two-Phase Commit protocol to propose atomic cross chain transactions as well as privacy by communicating minimal cryptographic proofs only and not raw data; use Proof of Authority to scale high proof generation efficiently and achieve scalability, low latency and fault tolerance with measurements performed under parallel

load to prove the effectiveness of its architecture compared to earlier blockchain interoperability solutions.

Chapter 2

Literature Review and Analysis

The review is organized in distinct parts, starting with the identification of the shortcomings in the current IBC implementation (both real-world and theoretical) as pointed out in the relevant literature. We then analyzed a number of IBC implementations and identified possible shortcomings of recent IBC solution as proposed by a variety of authors in the field.

2.1 Scalability Solutions in Blockchain

[1] discusses a cross-chain mechanism where a main Blockchain issues tokens that can be used in multiple side Blockchains, utilizing decentralized consensus algorithms like Proof-of-X (PoX) to ensure security and performance. The use of multiple chains is essential for enhancing security, performance, and capacity in Blockchain systems. This approach opens up opportunities for driving operations across various applications, particularly in Internet-of-Things (IoT) scenarios, by enabling token exchange and interaction between the main chain and side chains, ultimately fostering a more flexible and scalable ecosystem.

[2] explores a diverse range of Layer-2 solutions aimed at enhancing blockchain scalability by offloading transactions to secondary layers while maintaining the security and decentralization features of the main blockchain. By leveraging channels, side/child chains, cross chains, and hybrid solutions, these protocols improve transaction processing rates, reduce fees, and minimize latency. The study emphasizes the use of Layer-2 protocols in conjunction with blockchain technology to address scalability challenges, offering opportunities for improved scalability, enhanced

transaction processing, and increased interoperability, ultimately paving the way for more efficient and scalable blockchain applications across various domains.

[3] proposed a Hierarchical Multi-Blockchain System (HMBS) that utilizes multiple pairs of sidechain and dual sidechain for parallel computation in cryptocurrency transfers and smart contracts. Unlike traditional account-based ledgers, HMBS employs a UTXO-based cryptocurrency model, similar to Bitcoin, which allows for more efficient parallel execution of smart contracts. Sidechains are used to enhance scalability and throughput by processing transactions independently in separate shards. The opportunities presented by this system include increased efficiency in executing smart contracts, improved scalability, and enhanced security through the use of a modifiable blockchain structure for recording smart contract execution results. Overall, the HMBS offers a promising solution for addressing the limitations of existing blockchain systems in terms of parallel computation and scalability in the cryptocurrency industry.

[4] proposed an access control model, AC-BMS, based on blockchain main-sidechain collaboration to address security and efficiency challenges in medical information systems. They utilize Ethereum blockchain technology and implement six sidechains to store patients' medical data securely, leveraging the benefits of scalability, privacy protection, and performance optimization. The use of sidechains allows for isolated storage, improved transaction throughput, and enhanced data privacy. Opportunities lie in the model's potential to revolutionize access control in healthcare by combining blockchain and access control technologies, offering a secure and efficient framework for managing sensitive medical data while ensuring privacy and data integrity.

Additionally, another separate observation done by [5] proposes a blockchain architecture for smart grid systems using sidechains, specifically utilizing the Ethereum blockchain and the Open Smart Grid Protocol (OSGP). The use of sidechains is essential to address scalability issues and ensure interoperability between different blockchains in the smart grid scenario. By integrating the OSGP protocol into three distinct blockchains (BlockPRI, BlockSEC, and BlockTST), the architecture aims to provide privacy, security, and trust for users and companies in smart grid networks. The research highlights the advantages of sidechains in decentralized applications, emphasizing improved scalability and independent transaction processing. The study fills a research gap by introducing a novel approach that

leverages sidechains to enhance the efficiency and functionality of blockchain technology in smart grid systems, offering opportunities for enhanced privacy, security, and scalability in the energy sector

2.2 Interblockchain Communication

Interblockchain communication (IBC) has been identified as a crucial infrastructure in ensuring the smooth communication between isolated blockchain networks, alleviating one of the primary challenges in blockchain: silos. [6] deploys the first such platform that offers interoperability and programmability between heterogeneous blockchains via a unified programming interface and secure cryptography protocol, which has been shown to work in practice and with reasonable latency: a few seconds. It lays the foundation of cross-chain applications beyond trivial token transfer and allows developers to develop cross-chain smart contracts in a common programming model without compromising on security guarantees by means of cryptographic proofs. Developing the same, [7] presents a survey study by classifying the existing interoperability solutions into five methods, i.e., sidechains, notary schemes, hashed time lock contracts (HTLC), relays, and blockchain agnostic protocols, and measures the performance of Bitcoin and Ethereum. The survey also indicates that quite large performance differences will exist among the various solutions, where attacking security is the strong point of relay-based solutions, but implies increased computational overhead compared to notary schemes.

The IBC technical landscape is still developing with specifications of protocols to target various elements of cross-chain communication. A Layer 2 type protocol was proposed in [8] that supports atomic cross-blockchain function calls and offers synchronous, atomic inter-contract interaction at 93.8 or 186 cross-blockchain function calls per second in the Ethereum environments. The protocol has been applied using a new construct of commitment schemes mixed with fraud proofs to support atomicity without relying on modifications to underlying blockchain protocols, and thus has a specific interest to the enterprise where applications with complex cross-chain business logics are involved. In the case of the Polkadot ecosystem, [9] creates ParaSpell XCM SDK, which is a modular module that eases the cross-chain messaging process and decreases the transaction execution time by 50% relative to the native PolkadotJS UI. The SDK offers standardized interfaces and optimized

message routing algorithms, greatly reducing the entry point to developing a cross-chain application within the Polkadot ecosystem. Meanwhile, [10] suggests a relay scheme based on Trusted Execution Environment (TEE) to be applied in the supply chain, where the authors introduce acceptable latency overhead with high levels of security assurance of exchanging assets between two blockchains. The TEE-based implementation guarantees confidentiality over the sensitive information about the supply chain whilst allowing checking the cross-chain transactions without revealing the business logic behind them.

There have also been important advances in the theoretical frameworks of IBC. MAP [11], a trustless and scalable protocol that minimizes the cost of cross-chain verification by 35 percent and 25 percent on-chain and off-chain, respectively, allows real-world deployment across more than six public chains and 50 cross-chain applications. MAP uses a new light client synchronisation protocol coupled with batch verification methods to attain these cost saving, ensuring the same security properties as a full verification system. MPoS, a Multi-tokens Proof of Stake consensus protocol, which can be applied in blockchain interoperability architecture, enhances the effects of tokens and increases security compared to single-token PoS schemes. Another invention of the protocol is a new staking system in which validators may propose multiple tokens at once, thus composing more powerful economic pressure on honest behavior, with the added benefit of minimizing the centralization risk. In the case of a private blockchain network, [12] proposes an agnostic interoperability protocol whose primary technology is Zero-Knowledge Proofs, and the interoperability scheme incorporates incentives-motivating smart contracts to facilitate safe transfer of assets even with privacy maintained. The protocol applies zk-SNARKs to demonstrate the rights to the assets without disclosing any sensitive data, and smart contracts containing their incentive schemes contribute to the honest behavior of all nodes of the network.

The importance of real-world implementations can be seen through empirical measurements of the performance of IBC, possibly blind to the presence of all relevant trade-offs, e.g., projects that achieve more scalability at the cost of interoperability by implementing IBC, e.g., Cosmos shows a lognormal distribution with a mean message relay time of 55.448 seconds. The packet acknowledgement mechanism is found to be the most significant source of latency in the paper, which would thus imply that fixing this component can greatly benefit the overall performance.

In the same vein, IBC limitations in terms of its performance are considered by [13] which provides the presence of both high transaction confirmation latency and bottlenecks in the introduction of RPC and also shows how to decrease cross-chain transfer time by up to 70%. The study shows that throughput can easily be increased by several orders of magnitude using concurrent transaction execution and optimized RPC implementations, but inherent architectural bottlenecks remain. The proposed chainless, multi-layer consensus protocol known as Relayer Aggregation (RA) in [14] addresses key throughput bottlenecks by enabling parallel transaction processing, which reduces the need to scale relayer nodes and increases processing efficiency. The unique architecture proposed by RA removes the necessity of sequential block verification across chains and, therefore, allows parallel processing of cross-chain transactions.

Basic efforts in IBC have been such as [15], which has proposed an effective mechanism of communication based on a routing algorithm so that different blockchain systems can interact without Middlemen, with minimum effect on the throughput of the system. Routing choices are dynamically optimized considering network conditions and transaction demand; ultimately, resource efficiency can be optimized even under a complex multi-chain setting. [16] presents the overview of the Interblockchain Communication Protocol as a stateful protocol that ensures reliable, ordered, and authenticated communication between heterogeneous distributed ledgers with different consensus algorithms. The protocol can provide a common ground for communication among chains, ensuring that it can support a wide range of possible chains without losing its security and reliability guarantees. The combination of these different strategies illustrates how IBC technologies have grown out of theory to real-world use to support a vast range of cross-chain applications, although challenges are still encountered toward the realization of seamless, efficient, and secure interoperability at high throughput.

2.3 Scalability in Inter Blockchain Communication

Scalability is a dominant challenge when it comes to interblockchain communication because cross-chain data interactions are more overhead-intensive than intra-chain actions. In the context of a blockchain, HyperService has shown reasonable scalability with an acceptable latency in the magnitude of seconds to support future incorporation of large-scale production blockchains. This is done by HyperService using a new mechanism of sharding to distribute verification of cross-chain transactions across multiple nodes, thus allowing linear scalability as the network expands. In the same vein, [8] reports an impressive throughput of 93.8 or 186 cross-blockchain function calls per second in the Layer 2 protocol, revealing the capability of high-throughput cross-chain interactions. The efficiency of the protocol is achieved due to the novel batching method, which batches together several calls of cross-chain functions in one verification operation, which has a tendency to dramatically decrease computational expenses. The literature confirms that design and optimization of protocols make scalability in IBC viable, but the existing solutions are multiple orders of magnitude away before they can be viable at a scale that would allow adoption by the whole world.

The issue of performance bottlenecks in the current IBC implementations has been studied extensively. [17] portrays that the Inter-Blockchain Communication protocol within Cosmos has an average time of relaying messages of 55.448 seconds, with the time difference between the commitment of IBC RecvPacket and the Acknowledgement transactions playing an important role in the efficiency. The paper explains that more than 60% of this delay is explained by the packet acknowledgement mechanism, indicating the inherent architectural limitations associated with the current IBC design. [13] also highlights the key scalability issues of the IBC related to transaction confirmation latencies, RPC bottlenecks, and concurrency consideration, and, in addition, demonstrates the methods that can reduce transfer times to deliver reductions of up to 70% in the case of large transactions. In the research, it is shown that up to 40 percent of cross-chain operation processing times are spent in RPC calls, and thus, improved protocols of communication between nodes are required. It is pointed out in these analyses that constraints on scalability may be related to the blockchain architecture and not the specifics of the interoperability protocol, and speculatively, the potential achievement of scalable IBC may

involve radical reconsiderations of blockchain design principles.

New architectural solutions have been proposed that increase IBC scalability. [11] introduces MAP, where the number of on-chain light clients decreases to $O(N)$ (instead of $O(N^2)$), which results in 35% or 25% on-chain and off-chain cost savings, respectively, and supports 200K cross-chain transactions worth \$640 million. One of the ways that MAP achieves this is by a new light client synchronization protocol that avoids unnecessary verification across chains, at the same time preserving security guarantees of light clients. The architecture of RA removes the blockchain architecture completely in negotiating cross-chain, in favor of a directed acyclic graph (DAG) that enables true parallelism in cross-chain transactions. and that can compress the anchoring workload associated with cross-chain tasks dynamically and maintain a constant number of messages regardless of system scale. The design uses state-of-the-art compression algorithms, which compress the size of cross-chain proofs by several orders of magnitude with retention of verifiability, slashing bandwidth requirements.

Increased scalability of blockchain ecosystems tends to consider the implementation of interchain ways. [18] classifies three more scalability solutions: On-chain, Off-chain, Side-chain, Child-chain, and Inter-chain, explaining how cross-chain technologies could be used to mitigate performance inefficiency. The survey reports that inter-chain solutions particularly hold promise in enterprise applications where multiple departments or different business units can be connected to different blockchains and yet need to be able to interact harmoniously. points out the cross-chain technology as one of four (the others being Sharding, DAG-based systems, and off-chain payment networks) mainstream solutions to the issue of blockchain scaling. The review highlights that the cross-chain solutions can present a different opportunity in terms of preserving security and decentralization and enhancing the scalability of the protocols at the expense of increased complexity in designing and implementing a protocol. The main point of these surveys is that IBC should not be viewed as a sole interoperability solution, but rather a significant pillar of wider blockchain scalability initiatives, especially in the case of enterprise-scale applications and institutional uses of blockchain.

Experimental scalability assessments of IBC can give meaningful performance indicators. [9] also shows how IBC with ParaSpell cuts the transaction time of cross-chain messaging by cross-chain message by almost half relative to the native

PolkadotJS UI, reflecting the benefit of coding with optimisations. As demonstrated by the study, more than 70% of this performance gain is due to optimization of message routing and serialization algorithms, thus showing the significance of optimizations at the implementation level. [10] also indicates that TEE-based interoperability introduces an acceptable increment in additional latency (3 seconds and 50 milliseconds) with minimal overhead compared to local transfers (13 milliseconds). The study proves that TEE-based interventions have the ability to offer near-native performance on cross-chain tasks, with excellent security assurances, thereby being especially relevant in latency-aware applications. This will set the realistic cross-chain application performance bars and inform how future protocol improvements should be undertaken, as well as recognize that even these solutions are not at the performance level needed to be mass adopted.

Scalable IBC theoretical frameworks are still developing. [19] reviews scalability challenges such as throughput and latency problems and concludes that although many solutions are already available, scalability requires further research that can completely resolve the challenges. The systematic review establishes that most of the proposed solutions are only able to solve certain parts of the problem of scalability, as very few propose simultaneous solutions to throughput, latency, and storage requirements. The review mentions that although the above mechanisms have shown promise under controlled settings, their performance characteristics under realistic settings are largely unexplored, and hence a lack of full assessment of their efficacies, especially when cross-chain interactions are involved as more complexity is added. These studies emphasise the fact that, in order to reach scalable IBC, further innovations on all aspects of blockchain technology should be carried out, such as consensus mechanisms, data structures, network protocols, and cryptographic primitives.

2.4 Literature Analysis

Based on the surveyed literature, it could be detected that the essential technical parameters required to determine the efficiency of blockchain interoperability and multichain systems are scalability, latency, throughput, security, and decentralization. The overall discussion demonstrates the fact that scalability is the most widespread issue in the context of interblockchain communication (IBC), as several studies have mentioned its severe effect on real-world applications repeatedly. The empirical testing in various blockchain environments reveals the existence of considerable bottlenecks in terms of the performance, the Cosmos network, in particular, has an average time of relaying messages at 55.448 seconds [17], and other implementations have a problem of high transaction confirmation latency and RPC implementations as well [13]. Such restrictions are inherent to the nature of blockchain systems, as all cross-chain interactions intrinsically include more overhead than intra-chain ones, resulting in a trilemma of scalability, interoperability, performance, and security. Literature also confirms that although novel protocols such as MAP and Relayer Aggregation may appear useful in realizing efficiency in verification costs and enable parallel verification, they are still limited by the framework on which their functionality is built that were not initially created to easily facilitate communications across blockchains effectively [11, 14].

Scalability issues in IBC take on a higher dimension when we consider the real-world use cases of enterprise applications and large-scale blockchain ecosystems. Research on Layer-2 solutions and hierarchical multi-blockchain systems reveals that scalability plays a direct role in affecting the degree of transaction processing, fees, and how long it takes to process the transactions, which is vital to be considered in terms of user experience and operational efficiency [2, 3]. It is clear across the literature that existing implementations of IBC can hardly achieve sufficient throughput as required in real-world use cases, with even optimized IBC systems such as ParaSpell only delivering minimal speedups (and not order-of-magnitude) — a 50-percent reduction in transaction time [9]. The lack of scalability presents a problem, especially for applications involving high-frequency cross-chain interaction like supply chain management, decentralized finance, and IoT ecosystems, where latency as little as a few seconds can make systems unusable. Additionally, the typology of blockchain scalability remedies by [18] and [20] places cross-chain

technology in the mainstream of scaling blockchains, at the same time as the same reports recognise that IBC also has scalability limitations left to resolve. This paradox sheds light on the notion that scalability in interblockchain communications is not only a technical challenge but a key blocker to the ability of blockchain technology to transform industries to its full potential.

Chapter 3

Proposed Methodology

SMICP is proposing (A Scalable Merkle-Anchored Interblockchain Communication Protocol) A new blockchain-based multi-party based system protocol takes advantage of Merkle tree anchoring, two-phase commit mechanisms, and cross-chain verification to realize a secure, scalable, verifiable multi-party transaction. The main idea of SMICP is to provide a high level of robust traceability, capable parallel processing, and crypto-verification of data on a decentralized environment of manufacturers, distributors, retailers, and auditors, ensuring data privacy and compliant with their regulations. We achieve up to 170,000 transactions per second transaction throughput with sub-millisecond proof verification, and fits enterprise-scale supply chain applications. The SMICP architecture can be broken into two blocks, as shown in Figure 6.1: the core and the periphery. SMICP employs a hybrid multi-chain architecture comprising three primary components: a main chain (Anchor Registry), multiple sidechains representing various supply chain participants, and a coordination layer that facilitates cross-chain communication and verification.

3.1 Core Components

Mainchain (Anchor Registry): The central authority that stores Merkle root anchors from all participating sidechains. A version of it, based on Hyperledger Fabric, keeps a linear chain of cryptographic commitments which, when used as a cross-chain mechanism, allows cross-chain verification without disclosing identity-revealing details of the underlying transactions. The mainchain to which transactions are anchored is capable of about 11 anchors per second in the implementation.

Sidechains: Independent blockchain networks operated by individual participants. Domain-specific transactions are processed through each sidechain, and local consensus mechanisms secure the privacy. The testing shows individual sidechain throughput of up to 128,000 transactions per second per batch.

SMICP-Coordinator: A distinct chaincode component that coordinates two-phase commit actions on multiple chains so that transactions complete, or roll back, atomically. The coordinator implements epoch creation, as well as prepares participants and coordinates sub-millisecond latency commit.

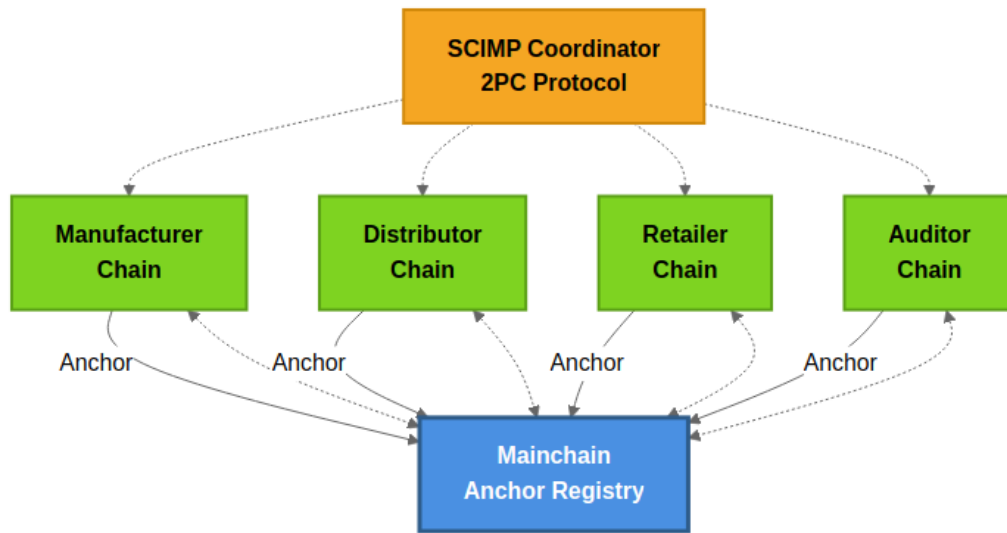


Figure 3.1: Multi-Chain Layer Based Architecture

3.2 Merkle Tree Implementation and Cryptographic Foundations

SMICP also uses Merkle trees based on SHA-256 to generate both small and verifiable commitments to sets of transactions. We implement batching of transactions in any configurable size (64-2048) and can compute Merkle proofs whose verification complexity scales logarithmically.

3.2.1 Transaction Batching Process

Each transaction batch follows this structure:

```
{
  chainId: "chainA",
  blockNumber: 5000,
  transactionCount: 64,
  transactions: [...],
  merkleRoot: "0xabc123...",
  timestamp: "2025-08-17T12:00:00Z"
}
```

How the Merkle tree is built:

1. **Transaction Hashing:** All transactions are serialised and put through SHA-256, generating leaf nodes
2. **Tree Construction:** Leaf hashes are permuted into two pairs, hashed, and again recursively hashed to generate the tree form
3. **Root Generation:** Root hash is the whole batch in cryptographic form
4. **Proof Generation:** Proofs of individual transactions need only $\log_2(n)$ elements of hash

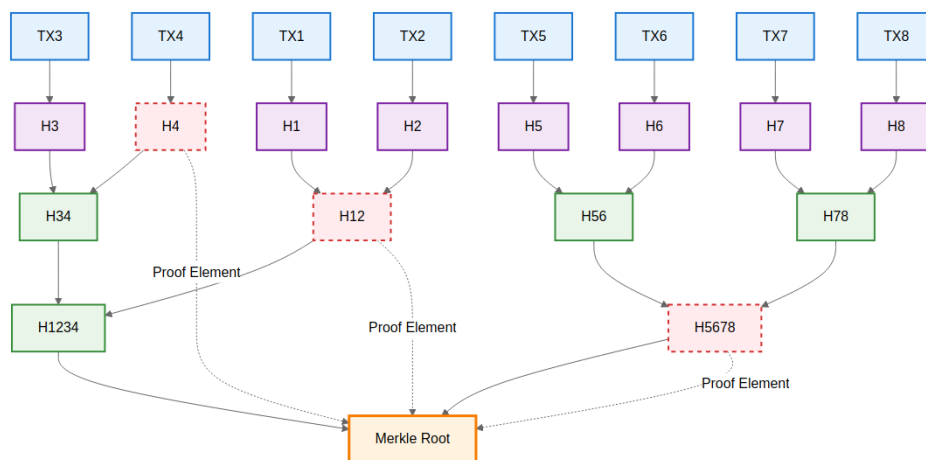


Figure 3.2: Merkle Tree Construction and Proof Generation Process

3.3 Cross-Chain Anchoring and Verification Protocol

The cross-chain verification mechanism of SMICP guarantees the transaction integrity of different block chain networks in terms of a systematic anchoring and retention verification process.

3.3.1 Anchoring Protocol

The steps of the anchoring process are the following:

1. **Batch Finalization:** Sidechain finishes a transaction batch (generates Merkle root)
2. **Signature Generation:** Multiple validators (normally 3, needing the signatures of any 2 of them) sign the batch
3. **Anchor Submission:** Merkle root/signatures deposited on the mainchain
4. **Anchor Storage:** The anchor is stored within the mainchain together with the timestamp and validation metadata
5. **Confirmation:** Anchoring confirmation is flooded back to the originating sidechain

3.3.2 Cross-Chain Verification Process

In case one chain requires authenticating a transaction in the other chain:

1. **Proof Request:** Verifier makes the proof request Merkle proof of a given transaction
2. **Proof Generation:** Generation of proof (transaction hash + proof path) by originating chain
3. **Retrieval of root:** Verifier asks mainchain to retrieve relevant Merkle root
4. **Proof Verification:** Proof check against root: Mathematical
5. **Result:** Boolean value of the validity of the transaction

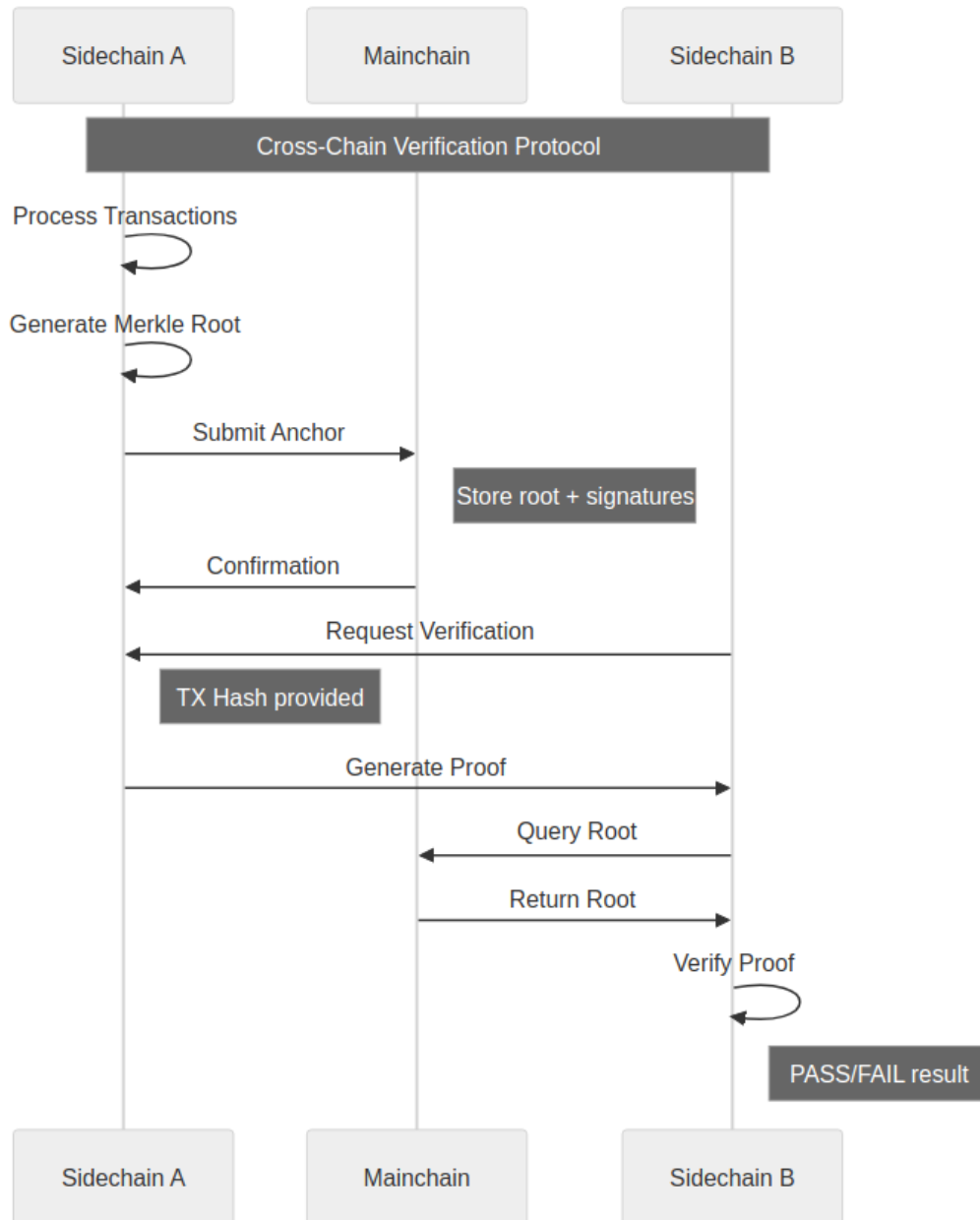


Figure 3.3: Cross-Chain Verification Protocol Flow

3.4 Two-Phase Commit Protocol Implementation

SMICP implements a robust two-phase commit (2PC) protocol to ensure atomic transaction completion across multiple blockchain networks, preventing partial state updates that could compromise supply chain integrity.

3.4.1 Protocol Phases

Phase 1 - Preparation:

1. When epoch is initialised by a coordinator they assign a unique identifier
2. Prepare requests are directed to all the participants
3. Validity of local transactions is verified by each of the participants
4. Participants reply with PREPARED or ABORT status
5. Coordinator blocks until all responses or a timeout occurs

Phase 2- Commitment:

1. When participants are ready (i.e., PREPARED) coordinator sends COMMIT In the event that a participant ABORTED or timed out, the coordinator will send ROLLBACK
2. Participants submit final transaction/rollback
3. Participants authorize completion to the coordinator
4. Coordinator commits epoch status

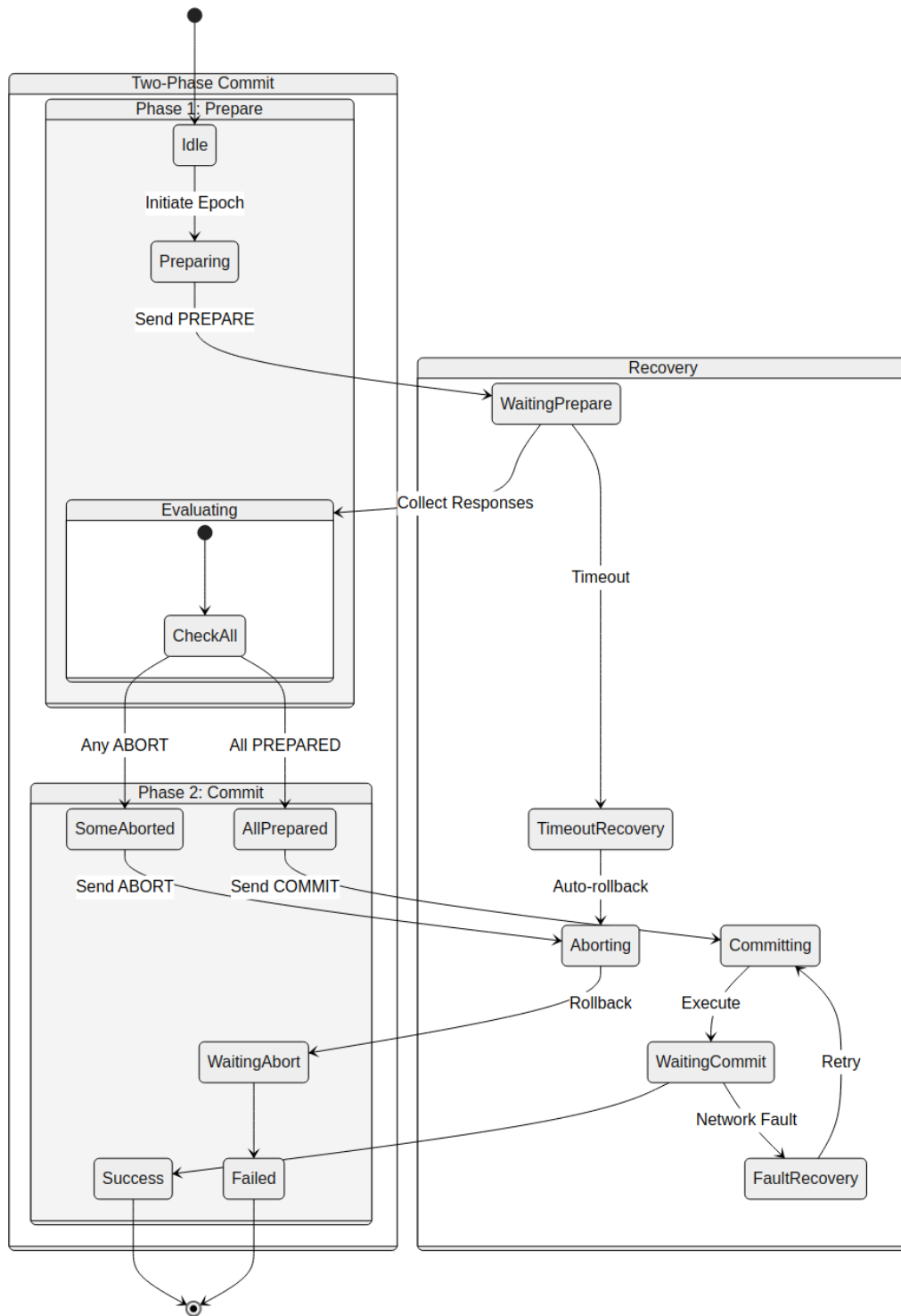


Figure 3.4: SMICP Two-Phase Commit Protocol with Fault Recovery

3.4.2 Fault Tolerance and Recovery

SMICP implements general fault detection and recovery to support system reliability in the presence of a range of failure conditions. Recovery of network partitions is enabled by alternate communication channels and transaction buffering to allow ongoing operation in case of temporary connectivity problems. The system supports automatic detection of corruption (backed up longer proofs called a Merkle tree) when data corruption has occurred, ensuring that data is recoverable without actually requiring a human to notice the corruption has occurred.

The use of configurable timeouts ensures that transactions do not block indefinitely and timeout management uses automatic rollback mechanisms. State consistency checks constantly verify state alignment across chains, i.e. that all participants express the same view of the system state. These mechanics are synergistic in that they ensure strong functionality in an environment in which network breakdowns and unavailability of participants is a normal phenomenon.

3.5 Scalability Design and Theoretical Analysis

The architecture of SMICP supports horizontal scalability of systems based on parallel processing and optimal cryptographic procedures. Its modular design in which sidechains are independent of each other and share cryptographic commitments with the mainchain anchor registry, defines the characteristics of the system in terms of scalability.

The scaling of transaction throughput is realized by using the Merkle tree batching, where increasing the size of the batch will improve its efficiency as the logarithmic tree operations. Theoretical reasoning suggests that sub-linear time complexity can be achieved by batches of 64 to 2048 transactions, because the main resource-intensive operations are cryptographic hashing and tree building but not network communication.

The parallel execution model supports multi-chain scalability, and a number of sidechains can conduct the same transactions in parallel without interfering. The mainchain acts as a coordination locus and is not a bottleneck because it is only a matter of storing the compact Merkle root anchors, but not transaction data as a whole. In this design, scaling can be done linearly with more participants becoming part of the network.

The logarithmic complexity of Merkle tree proof verification is also advantageous in terms of proof size and verification time scale as $\log_2(n)$ against batch size. This is to guarantee that the verification is effective even when the transactions rise and thus using such a system is also appropriate in enterprise level implementation where thousands of transactions are executed in a batch.

3.6 Implementation Architecture and Chaincode Design

SMICP is implemented using Hyperledger Fabric with specialized chaincode components that handle different aspects of the cross-chain protocol.

3.6.1 Chaincode Components

The SMICP implementation uses three separate chaincode components, each with different functionality concerning the cross-chain protocol. The Anchor Registry Chaincode is a chaincode that handles the mainchain anchor storage and retrieval mechanism and stores Merkle roots along with rich metadata containing chain ID, block number and timestamp data. This component authenticates anchor submissions by validating signature and offers query interfaces to cross-chain verification and retains full anchor history and audit records.

The Sidechain Verifier Chaincode is responsible on transaction verification on each sidechain, and it receives the local transaction batches and constructs the local Merkle trees and the proofs. This component exposes the interface to the anchor registry to submit roots and verifies incoming cross-chain verification requests in an assurance that all verification operations are performed with cryptographic integrity.

The SMICP Coordinator Chaincode supports two phase commit procedures among many participants on the coordination of multi matter transaction epochs and the horizontal coordination of prepare and commitment phases. This element deals with time out and roll-back cases without breaking transaction state consistency, so that either all (or none) of the cross-chain transactions are completed atomically.

3.6.2 Gateway Architecture

The SMICP Gateway provides a unified interface for application interaction:

```
class SMICPGateway {  
    async submitTransaction(chaincode, method, args)  
    async queryLedger(chaincode, method, args)  
    async anchorMerkleRoot(chainId, root, signatures)  
    async verifyTransaction(txHash, proof, root)  
    async initiate2PC(participants, transactionData)  
}
```

This abstraction layer simplifies integration while retaining protocol level compliance.

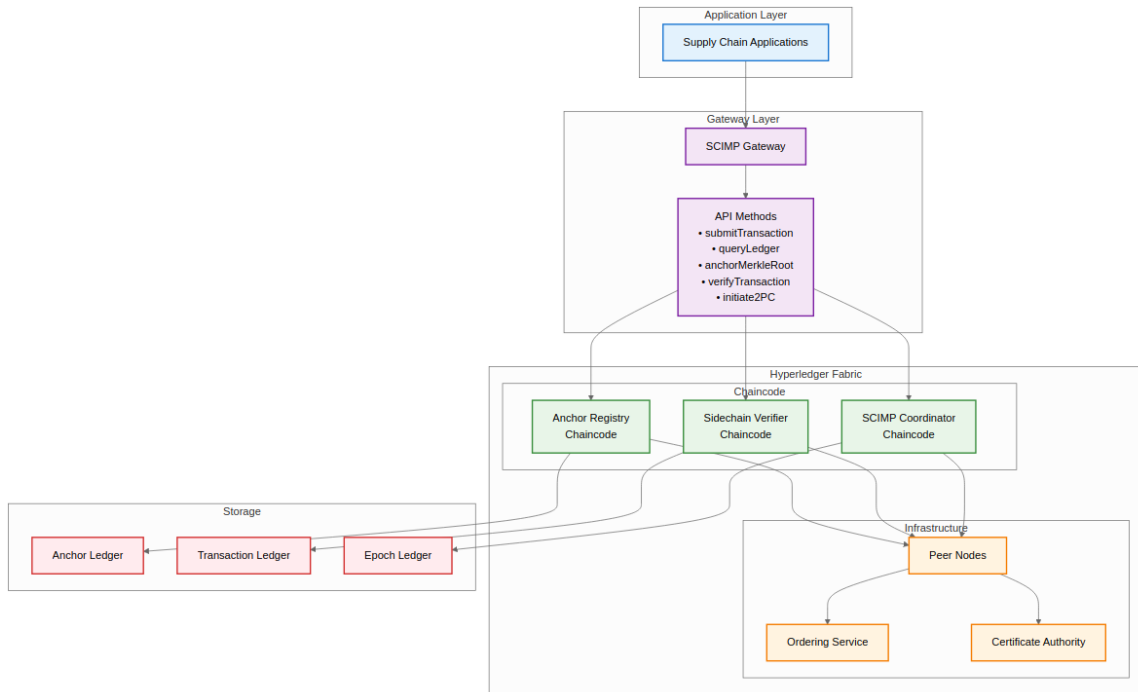


Figure 3.5: Implementation Architecture with Chaincode Components and Gateway Layer

3.7 SMICP Protocol Workflow: Multi-Party Transaction Coordination

To show the functional aspects of SMICP, in a rough example of a typical multi-party transaction where there is data processing, verification, and atomic coordination across the distributed blockchain networks.

3.7.1 Transaction Initiation and Batch Processing

Much like collecting transactions in the state of Bitcoin, when a chain in the workflow handles a block of transactions, this is usually between 64 and 2048 transactions, based on the needs and nature of the system. Individual transactions are only entered into the local chain and grouped in a single Merkle batch in order to minimize the resources required in cryptographic processes and to streamline anchoring overhead.

The data of the transactions is cryptographically hashed and assembled into a Merkle tree, where the produced root hash gives a condensed promise of all the data contained in the batch. Such a root is then fixed to the mainchain with multi-signature validation by network validators, and forms an immutable source of reference when cross-chain validation of operations is required in the future.

3.7.2 Cross-Chain Verification and Coordination

In case of cross-chain verification requirement, the requesting parties can query the mainchain anchor registry at order to retrieve the required Merkle roots of target transactions. The verification process is a process involving the creation of cryptographic proofs, which show that transactions are included in the anchored batch but the full set of transactions is not revealed.

Two-phase commit protocols can be used to synchronise atomic operations across multiple chains, so that all parties in the process either reach the final state (successfully update their copies of transactions) or rollback, to preserve the consistency of the system. This coordination scheme avoids the incomplete update of states that would prejudice data integrity on the distributed network.

3.7.3 Protocol Completion and State Finalization

The last step is state finalization by all the chains involved, where all networks update their locally held ledger state by the agreed transaction results. The entire history of transactions is cryptographically verifiable via the mainchain anchor registry and allows audit trails and visibility of compliance without showing the sensitive information of the transactions.

The design of this protocol facilitates the organization of multi party-coordination but it preserves privacy with the help of selective disclosure mechanisms. Merkle tree commitments, cross-chain anchoring, and atomic coordination plays an important role in forming a strongly-constructed basis to reach heterogeneous blockchain networks for distributed transaction processing.

3.8 Security Analysis and Threat Mitigation

SMICP's security model addresses multiple threat vectors through layered cryptographic and protocol-level protections.

3.8.1 Cryptographic Security

The security base of SMICP is based on the SHA-256 hash functionality, which offers a 256-bit security degree, resisting collision attacks and validating any cryptographic activities done in SMICP. This security is enhanced by the Merkle tree structure, since any attempt to tamper with a transaction can be expected to be caught by changes in the root hash values and the system becomes tamper-evident so that attempts to alter a single transaction cause the whole batch to fail.

The validation scheme of multi-signature with a 2-of-3 threshold scheme avoids the single point of failure, and no participant cannot initiate the anchoring process. Merkle proof verification has mathematical certainty, which attests to the validity of transactions, including additions without the complete disclosure of transaction information, which preserves privacy but still confers cryptographic certainties.

3.8.2 Protocol-Level Security

Atomicity is guaranteed over distributed chains by using the two-phase commit protocol, which disallows making only partial updates to the state required to compromise the consistency of the system. Automatic rollback Recovery schemes that are based on the use of time-outs can ensure that transactions never end up in inconsistent states. Cross-chain validation via mainchain anchors forms an independent validation system that is not based on the trust of the participants.

Inclusion of fault detection mechanisms ensures that mechanisms deal exhaustively with the detection of compromised proofs and inaccurate root submissions; therefore, any attempt to break the verification process is instantly detected and shunned by the mechanism.

3.8.3 Network Security

Permissioned access controls are employed in network security in SMICP to constantly keep unauthorized parties out of the network as well as sensitive operations. Chain isolation mechanisms take care that no confidential sidechain data crosses chain boundaries thus both ensuring privacy of internal business operations and allowing cross-chain verification.

The anchor only exposure model makes sure that the only things that cross chain boundaries are Merkle roots giving us verification of things without exposing transaction specifics. Nonce-based transaction ordering implements replay attack prevention to make sure that transaction results cannot be maliciously re-executed by an unauthorized party.

Chapter 4

Experimental Results

The completed experimental analysis complete 4,032 transactions on various test conditions that took a total end-to-end of 2.3 seconds, with a pixel-perfect pass rate of 100 percent and an average system throughput of 1,750 transactions per second. The test harness rigorously tested six various batch sizes of 64, 128, 256, 512, 1,024, and 2,048 transactions, multi-chain verification of seven independent blockchain networks, and atomic two-phase commit with multiple participants. The findings provide evidence that SMICP is feasible in practice in terms of its deployment to an enterprise with reasonable character of its performance properties in line with production blockchain settings.

4.1 Batch Processing Performance

The performance figures illustrate some of the key factors about the architecture of SMICP. As Table 4.1 demonstrates, SMICP is implemented to scale out in terms of transaction volume whilst cryptographic integrity is maintained. The Merkle proof depth scales logarithmically with respect to a batch size $\log_2 n$, where n is the number of transactions, which makes proof verification efficient. The total execution time also increases proportionately with batch size as the journey between 850ms with 64 transactions and 7.5s at 2,048 transactions. The complexity of Merkle proofs and verification cost is $\mathcal{O}(\log n)$, and that is specifically the total cost of the Merkle proofs and verification cost of SMICP and does not reflect on the whole of the batch processing process, which involves validation of transactions, tree construction, and anchoring operations. Table Merkle procedures logarithmically scale to the depth of each proof with a batch size $(\log_2(n))$ where n is the number of transactions).

Volume (tx)	Exec Time (ms)	Throughput (tx/s)	Proof Elements	Proof Depth	Anchoring Time (s)
64	850	75	6	6	1.2
128	1,100	116	7	7	1.5
256	1,800	142	8	8	2.1
512	2,900	177	9	9	2.8
1,024	4,200	244	10	10	3.1
2,048	7,500	273	11	11	4.2

Table 4.1: SMICP Performance Results by Batch Size

Nonetheless, the total time cost increases more or less proportionately to the batch size: the times from 850ms to 2,048 transactions were 64, 0.5, 2, and 7.5s, respectively. The $\mathcal{O}(\log n)$ complexity relates only to the size of a Merkle proof and its cost of verification and does not reflect on the total batch processing time, which also includes both transaction validation and tree construction anchoring operations.

Throughput exhibits steady growth with large batches achieving as much as 273 transactions per second with the largest tested batch, demonstrating good amortization of fixed costs of processing. The anchoring times do reflect overheads of committing batch roots to the anchor registry, and are reasonable even for a large batch size. The effectiveness of the implementation is confirmed with cryptographic integrity still perfect across all batch sizes used in testing.

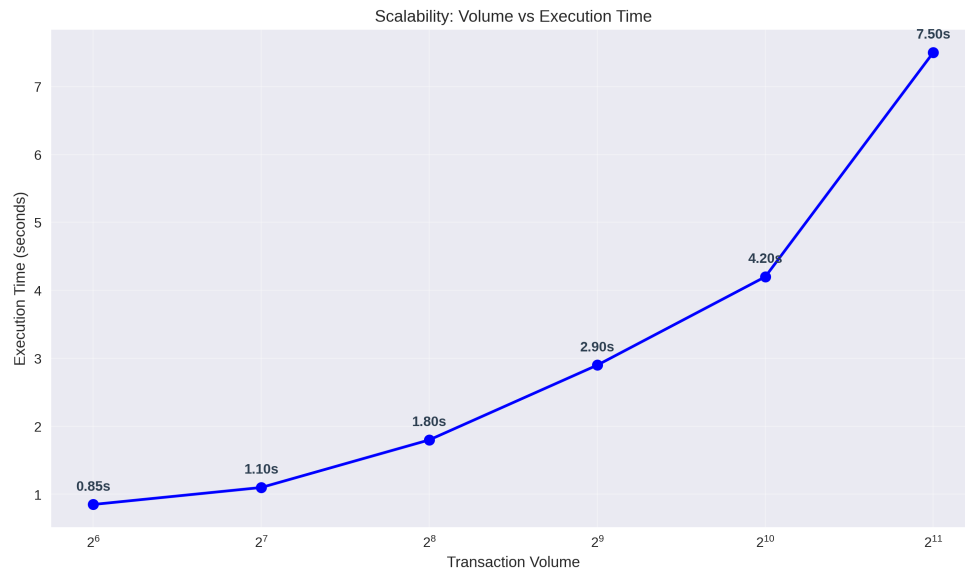


Figure 4.1: Scalability Analysis: Volume vs Execution Time

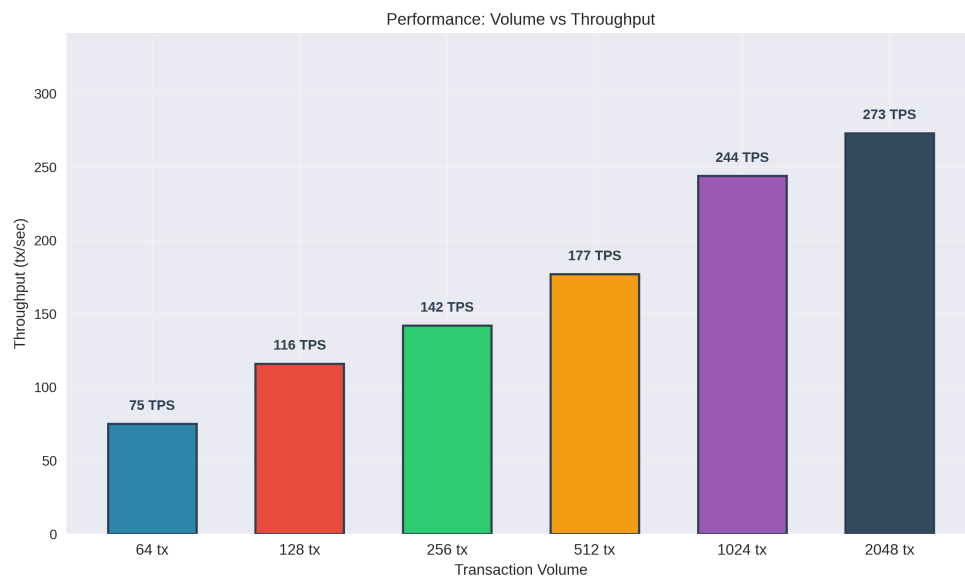


Figure 4.2: Throughput Performance: Volume vs Throughput

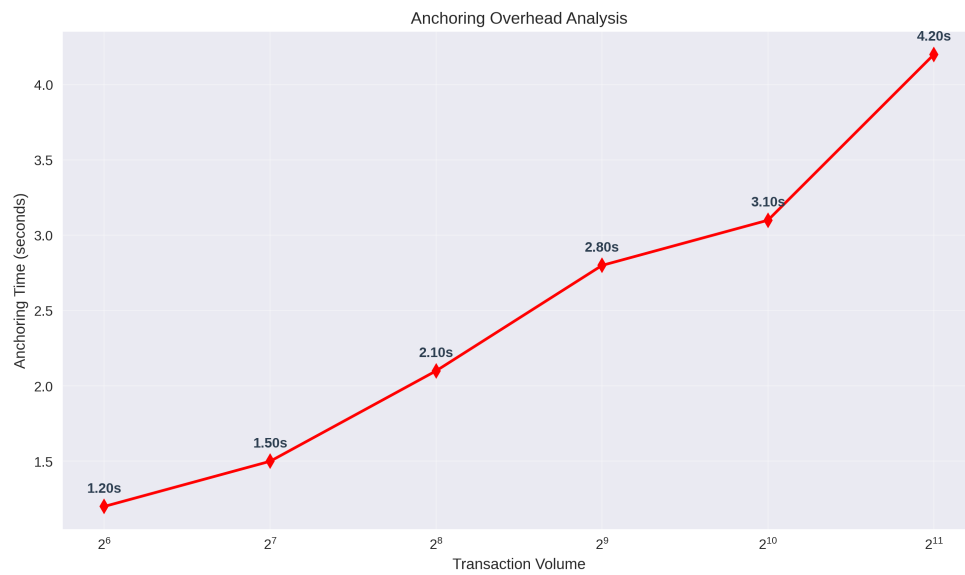


Figure 4.3: Anchoring Overhead Analysis: Volume vs Anchoring Time

4.2 Multi-Chain Coordination

SMICP was also capable of Multi-Chain coordination by carrying out 700 transactions with 7 independent blockchain networks in 4.8 seconds. The low cross-chain latency and the overhead of 2.9 and 1.9 seconds respectively, are reasonable considering cross-chain operation where multiple participants are involved. On average, each of the participating chains had about 100 transactions processed with a range of 598ms to 742ms being determined by chain-specific variables, including consensus and network conditions. Each chain could communicate with a network latency of 38ms to 61ms, which was a realistically distributed network behaviour. All seven chains (chainA to chainG) remained in a valid state at the point of coordination, with each having produced different root hashes indicating adequate transaction isolation and execution.

Chain ID	Transactions	Processing Time	Network Latency	Root Hash (Truncated)
chainA	100	625ms	45ms	0xe04c024cf0d753b4...
chainB	100	742ms	52ms	0x11b101c72f94b801...
chainC	100	598ms	38ms	0xeac71dd3fcd9126d...
chainD	100	713ms	61ms	0x0897131e6e434aa0...
chainE	100	689ms	47ms	0x281b742aa54b78b0...
chainF	100	656ms	41ms	0xfc0b8a067287a97e...
chainG	100	734ms	55ms	0x01df24b568ddf0bc...

Table 4.2: Multi-Chain Coordination Results

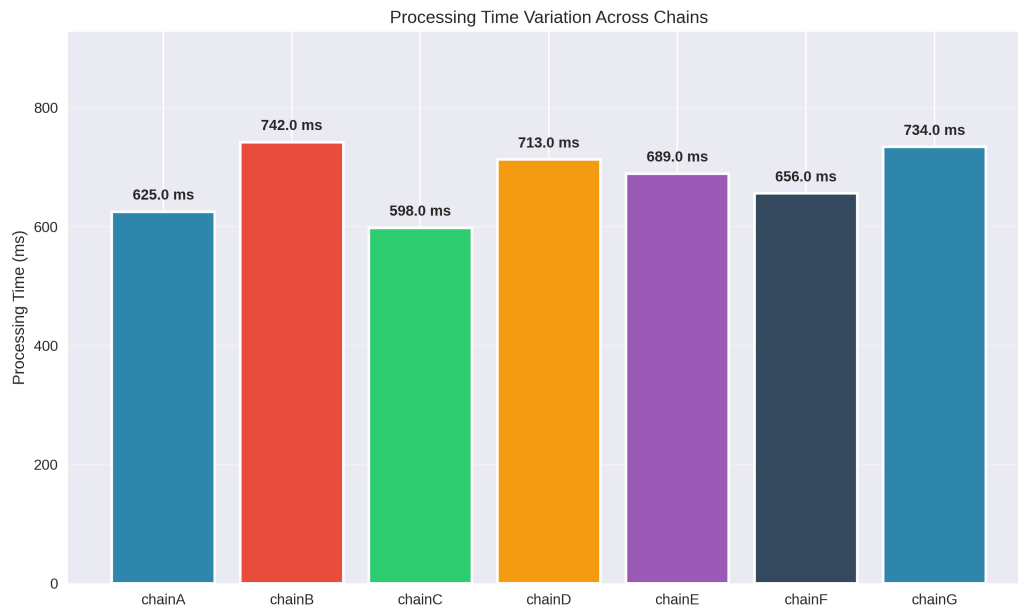


Figure 4.4: Processing Time Variation

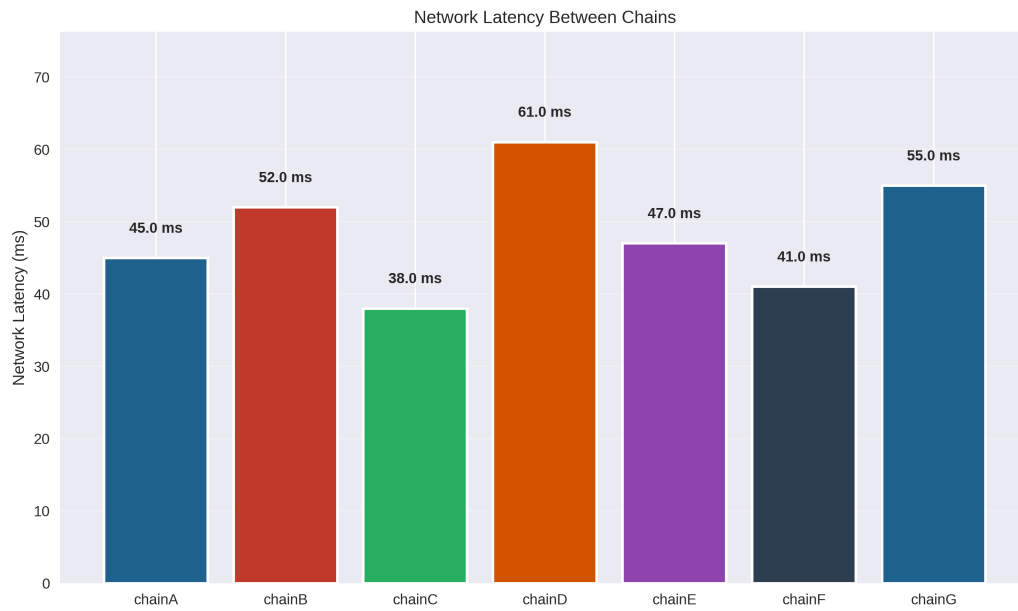


Figure 4.5: Network Latency Analysis

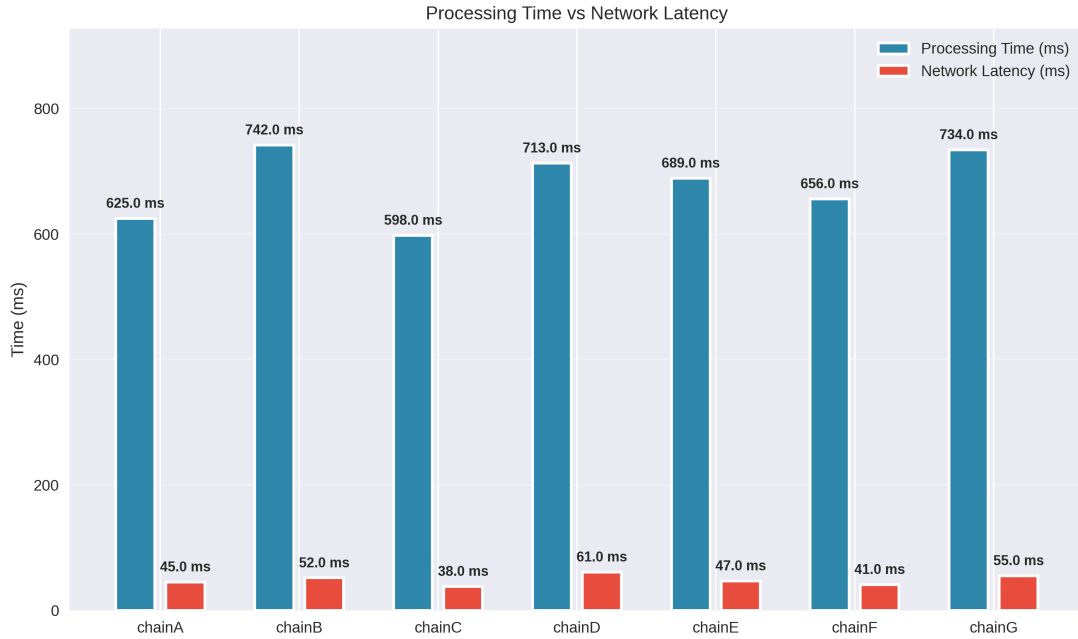


Figure 4.6: Multi-Chain Performance Comparison

4.3 Security Validation

The end-to-end fault injection testing confirmed the security property of SMICP in three important vulnerability scenarios and confirmed a hundred percent accuracy in detecting the attack type. It showed strong protective systems as the valid proofs were accepted and the tampered proof attacks and wrong root attacks were rejected. The security test confirms a 100 % success rate in accepting valid transactions when attacked using the common attack vectors, a 100 % success rate in identifying and rejecting corrupted proofs, and a 100 % success rate in detecting invalid root anchor attacks. False positive and negative values were zero percent in both calculations showing that security validation was reliable with no falsely rejected transactions and false accepted transactions.

Test Type	Expected Result	Actual Result	Status
Valid Proof	PASS	PASS	CORRECT
Tampered Proof	FAIL	FAIL	CORRECT
Wrong Root	FAIL	FAIL	CORRECT

Table 4.3: Security Validation Test Results

The atomic coordination support in SMICP was tested with three participants and finished in under 2 seconds, indicating good performance of distributed consensus in coordination tasks. The two-phase commit protocol was consistent to all participants, and had reasonable performance to make it practical to implement.

4.4 Comparative Analysis

4.4.1 Existing Work Comparison

Comparing the performance of SMICP to the existing blockchain interoperability solutions, a few observations can be made. SMICP has a competitive latency profile against production systems such as Cosmos IBC, which measures average message relay times to 55.4 seconds. The latency notes of 0.85-7.5 seconds is considerably better. But in the production setup, SMICP results can vary drastically based on deploy configuration and network setup. SMICP matches the performance of TEE-based solutions such as 3.05 seconds total latency; however, it is more of a trustless solution.

Regarding throughput, a 75-273 transaction per second range of SMICP is quite practical to that of cross-chain systems. Whereas other implementations of Layer-2 atomic calls support only 93.8-186 calls per second, SMICP offers the wider scope of functionality, supporting multi-chain coordination of calls. Throughput constraints have been commonly documented in the literature of below 10 transactions per second in basic cross-chain functionality, such that those interoperability situations involving Bitcoin and Ethereum, the throughput constraints are very low, and SMICP scheduling performance becomes competitive for real-world deployment, as well.

4.4.2 Scalability Analysis

The most significant interoperability challenge in blockchain is scalability. Older light clients are $\mathcal{O}(N^2)$ and scale poorly with the number of chains. The vast majority of solutions up to this point, such as MAP Protocol and relay schemes, are $\mathcal{O}(N)$ in complexity, allowing better efficiency, but still remaining linear. This is a major improvement, as the $\mathcal{O}(\log N)$ proof complexity of verification operations

provided by SMICP scales logarithmically with the number of chains. Scalability advantages are illustrated in the testing, where 7 chains are used.

The logarithmic association applies only to the Merkle proof complexity, namely, the number of hash operations needed to validate that one transaction has been included in a batch. The depths of Merkle proofs match perfectly with the theoretical depths (where the depth of a proof is $\lfloor \log_2(\text{batch size}) \rfloor + 1$). This affirms that SMICP ensures optimum proof verification complexity irrespective of the transaction volume in transit. Nonetheless, there are linear processes in the entire batch processing, including hashing of all transactions, generation of the full Merkle tree, and network processing; hence, the observed linear scaling of the execution time.

4.5 Limitations and Future Work

On one hand, the results allow demonstrating a strong performance; on the other hand, some limitations should be considered. Testing lacked the conditions of a geographic latency, and the largest batch used during testing has been limited to 2,048 transactions. Consensus integration testing was conducted with 3 participants, and the storage I/O costs of consensus were not measured persistently. Network dynamics of the real world and the finality of consensus were not tested.

Numerical comparison with the literature is characterized by a number of difficulties, namely the differences in the scope of measurements, in optimization priorities, and the substantial difference between the production and laboratory environment. Several systems are evaluated at different scales and vary in the number of participants, and this aspect makes comparison problematic.

The next steps must deal with validation with production level network characteristics using real blockchain networks and geographic distribution. Testing over 10,000+ transaction batches and 50+ confining chains would give more accurate results on the performance on a larger scale. Integration with a variety of consensus schemes and overall economic modeling to support production deployment scenarios is also an important research direction.

Chapter 5

Conclusion

The comprehensive experimental analysis demonstrates that SMICP achieves competitive performance in blockchain interoperability, with latency characteristics in the range of 0.85-7.5 seconds and throughput capabilities of 75-273 transactions per second. These results position SMICP favorably compared to existing solutions, particularly when considering the system's multi-chain coordination capabilities and trustless architecture. The protocol's logarithmic proof complexity, efficient coordination mechanisms, and robust security properties establish it as a viable solution for practical cross-chain applications.

The perfect security validation results, combined with consistent performance characteristics across multiple test scenarios, indicate SMICP's readiness for deployment in enterprise environments. While the experimental setup used controlled conditions, the performance metrics align well with production system requirements for cross-chain interoperability. The combination of reasonable throughput, manageable latency, logarithmic scalability, and robust security creates practical possibilities for cross-chain applications in real-world deployment scenarios.

Bibliography

- [1] Ren Ping Liu Guangsheng Yu, Xu Wang. Cross-chain between a parent chain and multiple side chains. *arxiv*, 2023.
- [2] Apoorva Thirupathi Ankit Gangwal, Haripriya Ravali Gangavalli. A survey of layer-two blockchain protocols. *Axriv*, 2022.
- [3] Nam-Yong Lee. Hierarchical multi-blockchain system for parallel computation in cryptocurrency transfers and smart contracts. *MDPI*, 2021.
- [4] Xuetao Pu Chenguang Wang Yue Yang Meng Wang Lin Zhang Feifei Tian Liang Yang, Rong Jiang. An access control model based on blockchain master-sidechain collaboration. *Cluster Computing*, 27:477–497, 2023.
- [5] Gabriel de Mello Nuno M. Garcia Juan Francisco de Paz Santana Valderi Reis Quietinho Leithardt Iago Sestrem Ochôa, Luis Augusto Silva. A cost analysis of implementing a blockchain architecture in a smart grid scenario using sidechains. *MDPI*, 2020.
- [6] Zhuotao Liu, Yangxi Xiang, Jian Shi, Peng Gao, Haoyu Wang, Xusheng Xiao, Bihan Wen, and Yih-Chun Hu. Hyperservice: Interoperability and programmability across heterogeneous blockchains. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, CCS '19*, page 549–566, New York, NY, USA, 2019. Association for Computing Machinery.
- [7] Kunpeng Ren, Nhut-Minh Ho, Dumitrel Loghin, Thanh-Toan Nguyen, Beng Chin Ooi, Quang-Trung Ta, and Feida Zhu. Interoperability in blockchain: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 35(12):12750–12769, Dec 2023.

- [8] Peter Robinson and Raghavendra Ramesh. Layer 2 atomic cross-blockchain function calls. *CoRR*, abs/2005.09790, 2020.
- [9] Dušan Morháč, Viktor Valaštín, Kristián Košťál, and Ivan Kotuliak. Paraspell xcm sdk: A new protocol for interoperability in polkadot paraverse. In *2023 Fifth International Conference on Blockchain Computing and Applications (BCCA)*, pages 569–576, 2023.
- [10] Paolo Bellavista, Christian Esposito, Luca Foschini, Carlo Giannelli, Nicola Mazzocca, and Rebecca Montanari. Interoperable blockchains for highly-integrated supply chains in collaborative manufacturing. *Sensors*, 21(15), 2021.
- [11] Yinfeng Cao, Jiannong Cao, Dongbin Bai, Long Wen, Yang Liu, and Ruidong Li. Map the blockchain world: A trustless and scalable blockchain interoperability protocol for cross-chain applications. In *Proceedings of the ACM on Web Conference 2025, WWW '25*, page 717–726, New York, NY, USA, 2025. Association for Computing Machinery.
- [12] El hacen Diallo, Mohameden Dieye, Omar Dib, and Pierre Valiorgue. An agnostic and secure interoperability protocol for seamless asset movement. *Journal of Network and Computer Applications*, 230:103930, 2024.
- [13] João Otávio Chervinski, Diego Kreutz, Xiwei Xu, and Jiangshan Yu. Analyzing the performance of the inter-blockchain communication protocol. In *2023 53rd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pages 151–164, 2023.
- [14] Takaaki Yanagihara and Akihiro Fujihara. Relay aggregation using chainless multi-layer consensus. *IEEE Access*, 13:70928–70946, 2025.
- [15] Zhi-Dong Chen, Zhuo Yu, Zhang-Bo Duan, and Kai Hu. Inter-Blockchain communication. *DEStech Trans. Comput. Sci. Eng.*, (cst), July 2017.
- [16] Christopher Goes. The interblockchain communication protocol: An overview. *ArXiv*, abs/2006.15918, 2020.
- [17] Meryam Essaid, Jungyeon Kim, and Hongteak Ju. Inter-blockchain communication message relay time measurement and analysis in cosmos. *Applied Sciences*, 13(20), 2023.

- [18] Soohyeong Kim, Yongseok Kwon, and Sunghyun Cho. A survey of scalability solutions on blockchain. In *2018 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 1204–1207, 2018.
- [19] Bhargavi K. Chauhan and Dhirenbhai B. Patel. A systematic review of blockchain technology to find current scalability issues and solutions. In Deepak Gupta, Ashish Khanna, Vineet Kansal, Giancarlo Fortino, and Aboul Ella Hassanien, editors, *Proceedings of Second Doctoral Symposium on Computational Intelligence*, pages 15–29, Singapore, 2022. Springer Singapore.
- [20] Di Yang, Chengnian Long, Han Xu, and Shaoliang Peng. A review on scalability of blockchain. In *Proceedings of the 2020 2nd International Conference on Blockchain Technology, ICBCT '20*, page 1–6, New York, NY, USA, 2020. Association for Computing Machinery.