0.1 Introduction

The research and development of the Multi-chain Stablecoin Protocol have culminated in a series of architectural and algorithmic innovations designed to address the fundamental limitations of current Decentralized Finance (DeFi) infrastructure. Throughout the implementation of the Collateralized Debt Position (CDP) mechanism across heterogeneous blockchain networks, several critical challenges emerged, ranging from liquidity fragmentation and cryptographic incompatibility to asynchronous state inconsistency. This chapter details the specific solutions devised to overcome these hurdles. These contributions constitute the core intellectual value of the thesis, demonstrating a novel approach to cross-chain interoperability that prioritizes security, capital efficiency, and user experience over traditional bridging methods.

0.2 The State Orchestration Architecture via Hub-and-Spoke Model

0.2.1 Problem Identification

The current decentralized finance (DeFi) landscape is plagued by structural inefficiencies that hinder mass adoption and capital optimization. Through the analysis of existing multi-chain protocols, three critical problems have been identified:

First, there is a severe fragmentation of state data across isolated blockchain networks. In the current paradigm, a user's financial position on Ethereum is completely invisible to applications on Arbitrum or Solana. This isolation forces users to manage disjointed portfolios, making it impossible to leverage assets on one chain to support a position on another without physically moving the tokens. This lack of interoperability results in a rigid and inflexible user experience where managing solvency across multiple chains becomes a manual and error-prone process.

Second, the integration of traditional bridging solutions into DApps introduces significant friction. Standard bridges often suffer from high latency and exorbitant gas fees, as they require complex on-chain verification steps for every transfer. For a user who simply wants to use their ETH collateral to borrow stablecoins on a faster chain, the cost and time delay of bridging can negate the economic benefits of the transaction.

Third, and perhaps most critically, the traditional lock-and-mint bridge model results in a massive inefficiency of capital. In these systems, liquidity is often statically locked in bridge contracts on both the source and destination chains to maintain the peg of wrapped assets. This idle liquidity represents a "dead weight" in the ecosystem; billions of dollars are trapped in smart contracts without generating yield or contributing to the broader market depth, leading to a suboptimal utilization of the total value locked (TVL).

0.2.2 Proposed Solution

To address these challenges, this thesis implements a State Orchestration Architecture utilizing a Hub-and-Spoke topology. This model fundamentally shifts the focus from bridging assets to synchronizing state.

The primary advantage of this solution is its scalability. By designating Solana as the central Hub and EVM chains as Spokes, the system allows for the seamless addition of new blockchain networks without significant resource expenditure. Adding support for a new chain (e.g., Base or BNB) does not require deploying a complex mesh of bridges; it simply involves deploying a lightweight Controller Contract that reports to the central Hub. This makes the protocol highly adaptable to the rapidly evolving blockchain landscape.

Regarding capital efficiency, the architecture unlocks the potential of the otherwise idle collateral. Since the assets are locked in the protocol's native vaults on the source chains (Spokes) rather than being wrapped and fragmented, the protocol can implement strategies to rehypothecate this liquidity. The locked assets can be securely deployed into low-risk farming protocols (e.g., Aave or Compound) on their native chains to generate yield. This revenue stream can then be distributed back to the users or used to subsidize the borrowing rates, transforming the passive "lock" into an active, yield-bearing position.

Finally, the architecture provides a unified interface for position management. The Hub maintains a "Universal Wallet" that aggregates the user's collateral and debt from all connected Spokes into a single global state. This allows users to view and control their entire financial health from one dashboard. A user can deposit ETH on Ethereum and immediately see their borrowing power increase on Solana, enabling a fluid and cohesive cross-chain experience that was previously unattainable.

0.3 Cross-chain Identity Verification and Request Integrity

0.3.1 Problem Identification

Establishing a secure and unified user identity across heterogeneous blockchain networks presents a multi-faceted challenge that encompasses both logical association and technical verification. During the implementation phase, three distinct hurdles emerged regarding identity management and data integrity.

First, there is an inherent difficulty in linking identities across isolated ledgers. In a decentralized environment without a centralized Know-Your-Customer (KYC) database, an Ethereum address and a Solana address are cryptographically distinct entities with no intrinsic mathematical relationship. Determining whether two wallets on different chains belong to the same physical user is impossible through simple observation. Without a robust binding mechanism, the system cannot safely aggregate collateral from one chain to back debt on another, as it cannot verify the common ownership of assets.

Second, the reliance on an off-chain infrastructure (the Guardian) introduces a "Man-in-the-Middle" risk regarding data integrity. Since the Guardian is responsible for relaying the user's intent from the EVM chain to the Solana Hub, there is a theoretical risk that a compromised or malicious Guardian could tamper with the request payload.For example, changing the destination address of a minting request before submitting it to the Solana contract. The system must ensure that the message processed on the destination chain is exactly identical to the message authorized by the user on the source chain, regardless of the relayer's honesty.

Third, this verification process is technically complicated by cryptographic incompatibility. The Ethereum ecosystem relies on the Secp256k1 elliptic curve for digital signatures, whereas Solana natively utilizes the Ed25519 curve. This mismatch means that a standard Solana smart contract cannot natively verify an Ethereum signature to authenticate a user's request. Developing a custom verification algorithm in user-space code (e.g., in Rust/Anchor) is computationally prohibitive and would likely exceed the transaction compute budget, rendering the solution unscalable.

0.3.2 Proposed Solution

To overcome these obstacles, this thesis proposes a Cryptographically Bound Identity Protocol that leverages a hybrid verification model to ensure both identity linkage and request integrity.

To address the identity linking problem, the solution implements a "Handshake Protocol." The user is required to explicitly authorize the linkage by signing a binding request with their destination wallet (Solana) that references their source wallet (Ethereum). This creates a bidirectional cryptographic proof of ownership. Once verified, the Solana Hub stores this relationship in the Universal Wallet state. This effectively treats the Ethereum address as an authorized "signer" or "controller" for the Solana-based position, allowing the system to logically group disparate addresses under a single user profile.

Regarding data integrity and the Guardian's trust assumption, the solution adopts a "Verify-on-Chain" strategy. The Guardian is treated strictly as an untrusted courier.

When a user initiates a request, they sign the hash of the payload (including amount, nonce, and action type) using their private key. The Guardian transmits both the raw payload and the user's signature to Solana. The Solana contract reconstructs the hash from the raw payload and verifies it against the signature. If the Guardian attempts to tamper with even a single byte of the payload, the hash will change, and the signature verification will fail. This ensures that the Guardian cannot forge or modify user requests, guaranteeing end-to-end integrity.

Finally, to resolve the cryptographic incompatibility, the system utilizes Solana's Native Secp256k1 Program. Instead of performing the heavy elliptic curve calculations within the main business logic, the Gateway contract invokes this precompiled native program via a Cross-Program Invocation (CPI). This allows the system to recover the public key from an Ethereum signature efficiently and securely. By comparing the recovered address with the linked identity stored in the Universal Wallet, the system achieves a trustless authentication mechanism, enabling an Ethereum user to drive state transitions on Solana using their native credentials.

0.4 Asynchronous State Consistency via Mutex and Saga Pattern

0.4.1 Problem Identification

The distributed nature of cross-chain communication introduces fundamental challenges regarding data consistency that do not exist in traditional monolithic architectures. Three critical issues were identified during the system design:

First, unlike centralized database systems which guarantee Atomicity, Consistency, Isolation, and Durability (ACID) within a single transaction scope, cross-chain operations are inherently non-atomic. A transaction executing on Ethereum cannot natively roll back its state based on the outcome of a transaction on Solana. Once a transaction is mined on the source chain, it is finalized, regardless of whether the subsequent steps on the destination chain succeed or fail. This lack of global atomicity creates a dangerous window where the system state is essentially "in transit" and undefined.

Second, independent blockchains operate in complete isolation; they are essentially "blind" to each other's state. The EVM Controller has no direct way of knowing whether the Solana Hub has successfully processed a minting request or if it failed due to a logic error (e.g., slippage or insufficient collateral). Without a robust synchronization mechanism, this information gap can lead to permanent state divergence, where assets are locked on one chain without a corresponding value issuance on the other.

Third, the asynchronous gap between request and finalization exposes the system to synchronization risks, specifically Replay Attacks and Desynchronization. If a user can initiate multiple requests rapidly before the first one is settled, they might exploit the latency to double-spend their collateral or confuse the state machine. Furthermore, if a failure occurs on the destination chain and is not properly propagated back to the source, user funds could be permanently frozen in the smart contract, leading to a catastrophic loss of trust and financial value.

0.4.2 Proposed Solution

To restore a form of distributed atomicity and ensure eventual consistency, this thesis implements a dual-layer synchronization protocol combining Pessimistic Locking (Mutex) and the Saga Pattern.

To address the issue of state isolation and replay attacks, the EVM Controller implements a strict Mutex (Mutual Exclusion) Lock. Every user is assigned a unique, monotonically increasing nonce. When a user initiates a cross-chain request, the contract checks the lock status. If unlocked, it immediately sets a ‘isLocked‘ flag to true and increments the nonce. This action effectively freezes the user’s ability to interact with the system until the current lifecycle is resolved. This mechanism serializes the asynchronous events, converting a complex parallel state problem into a manageable sequential workflow, thereby neutralizing race conditions and replay attempts at the source.

To solve the problem of non-atomic failures and potential fund loss, the Guardian middleware adopts the Saga Pattern for distributed transactions. The cross-chain operation is modeled not as a single transaction but as a saga, it is a sequence of local transactions. The Guardian monitors the execution result on the Solana Hub. If the transaction succeeds, the saga proceeds forward, and the Guardian triggers a "Finalize" callback to the EVM chain to mint tokens and release the lock. However, if the Solana transaction fails (e.g., due to a sudden drop in collateral price causing a health factor check failure), the Guardian executes a Compensating Transaction. It submits a "Revert" proof to the EVM Controller, which triggers the release of the locked assets and the Mutex. This ensures that the system always converges to a consistent state either fully successful or fully rolled back eliminating the risk of funds being stuck in limbo.

0.5 Hub-Centric Liquidation Strategy with Liquidity Rebalancing

0.5.1 Problem Identification

In any Collateralized Debt Position (CDP) protocol, the liquidation mechanism is the ultimate line of defense against insolvency. However, in a cross-chain environment where collateral is held on a source chain (e.g., Ethereum) while debt is issued on a destination chain (e.g., Solana), the traditional liquidation model faces a critical latency bottleneck.

The primary problem is the reaction time to market volatility. Cross-chain communication typically involves a delay ranging from minutes to hours depending on network congestion and finality thresholds. In a scenario of extreme market volatility (a "Black Swan" event), the price of collateral can plunge significantly within seconds. If the protocol relies on a cross-chain message to trigger liquidation or to move assets from Ethereum to Solana to pay the liquidator, the delay may cause the position to become under-collateralized (Bad Debt) before the liquidation transaction is finalized. The inability to react instantaneously to price feeds renders standard cross-chain liquidation models unsafe and capital inefficient.

0.5.2 Proposed Solution

To guarantee system solvency and incentivize liquidators, this thesis introduces a Hub-Centric Liquidation Strategy supported by an active Liquidity Rebalancing Mechanism.

The core solution determines that the liquidation event must be executed exclusively on the Solana Hub. Since Solana holds the global state (the Universal Wallet) and has sub-second block times, it is the only environment capable of processing real-time solvency checks. When a user’s Health Factor drops below the threshold, a liquidator interacts directly with the Solana Main Contract to repay the debt. This interaction is atomic and immediate, preventing the accumulation of bad debt regardless of the congestion status on the EVM chains.

To ensure that liquidators are compensated immediately without waiting for cross-chain bridging, the system maintains a "Liquidity Reserve" on Solana. The Guardian infrastructure implements a proactive Rebalancing Algorithm. It constantly monitors the ratio of collateral held in the EVM Vaults versus the Solana Reserve. When the reserve on Solana falls below a safety margin, the Guardian automatically bridges a portion of the idle collateral from the EVM spokes to the Solana Hub. This ensures that there is always sufficient physical liquidity on Solana to pay out liquidators instantly.