Lanca Bridging Framework: Enabling Cross-chain Liquidity Through A Hybrid Liquidity Management System

Andy Bohutsky[1], Oleg Kron[1], Nikita Gruzdev[1], Phuc Le[1]

[1]Concero Labs

23 June 2025 **(v1.0)**

Abstract

Cross-chain solutions commonly face significant challenges, such as the need for substantial liquidity to be locked across multiple blockchains, resulting in inefficiencies, and a course to compromise security in favour of achieving faster transaction speeds. This problem encourages us to develop a better solution to address all of these disadvantages without trading off any aspect of the cross-chain bridge trilemma - the Lanca Bridging Framework.

Lanca Bridging Framework is a fully decentralised cross-chain bridging framework enabling seamless value transfer across different blockchain networks. Its technology is based on a novel Parent-Child Pool architecture and dynamic IOU (I-owe-you) system. It addresses these aforementioned challenges by introducing a Parent Pool on a Master chain that manages Child Pools across different blockchains, ensuring efficient liquidity distribution and rebalancing. The framework implements a novel IOU-based rebalancing mechanism where participants called Rebalancers earn fees by maintaining liquidity equilibrium across Child Pools on multiple blockchains. Lanca's technical design ensures high capital efficiency, reduces idle liquidity, and scales effortlessly to any chain in several hours thanks to the underlying Concero's messaging infrastructure. Additionally, security is the top priority in Lanca's architecture, so cross-chain messaging is powered by Concero V2 Messaging, which is employed by both cryptographic and economic security. The Parent-Child Pool models, combined with dynamic rebalancing, create a unified liquidity layer that strengthens cross-chain integration while ensuring security through a thorough validation system.

This whitepaper presents Lanca Bridging Framework's architecture, technical specifications, economic models, and security considerations. It demonstrates how Lanca Bridging Framework can achieve efficient cross-chain liquidity management without compromising decentralisation or security.

Contents

1 Introduction

The blockchain landscape has evolved significantly since Bitcoin's inception and Ethereum's smart contract revolution. These technologies lead to a diverse ecosystem of blockchain networks, each optimised for specific use cases. While this diversity promotes innovation, it has created a fragmented ecosystem where assets and liquidity are isolated across different chains.

As of this writing, the blockchain landscape encompasses 277 Layer-1 [1] and 105 Layer-2 [2] public blockchains, according to Coingecko. However, these figures represent only a fraction of the broader landscape as they represent only public blockchains. The number of private and bank chains developed by financial institutions and banks is growing exponentially, as we can see in the adoption wave through the involvement of capital market players [3]. So, this trend further magnifies fragmentation across the blockchain ecosystem.

Smart contract technology has revolutionised financial markets through decentralised finance (DeFi), which creates a borderless ecosystem for global value transfer and trading across blockchain networks. This transformation has captured significant institutional interest, with The Economist's Q2 research report for OKX projecting digital asset market values to exceed $10 trillion by 2030 [4]. As the DeFi ecosystem expands, the need for efficient cross-chain liquidity solutions becomes increasingly critical for sustaining this growth.

Current blockchain interoperability solutions, particularly trusted bridges and atomic swaps, face significant limitations despite their basic functionality. The primary challenge lies in their requirement for substantial locked capital across multiple blockchains, resulting in inefficient use of resources. Consequently, these solutions struggle to provide seamless cross-chain liquidity while maintaining optimal capital efficiency. The severity of these security risks became evident during 2021-2022, when bridge exploits led to losses exceeding $2 billion.

Existing cross-chain solutions like Stargate and Across Protocol have introduced innovative approaches to blockchain interoperability, each with distinct advantages and limitations. Stargate's unified liquidity pools architecture, powered by its V2 AI Planning Module (AIPM), facilitates seamless cross-chain transfers without the need for wrapped tokens while optimising costs through efficient transaction batching. While its integration with LayerZero provides message verification, Stargate's limited asset support and centralisation risks through its foundation-controlled structure present notable drawbacks.

Similarly, Across Protocol's intent-based architecture, with its competitive relayers and optimistic verification, delivers remarkable transaction speeds through immediate execution and settlement facilitated by UMA's oracle. However, its dependence on a strong relayer marketplace and the capital efficiency challenges stemming from the oracle's challenge period underscore the persistent complexities in developing reliable cross-chain solutions. These limitations in current protocols encourage the development of more comprehensive solutions that can better address the challenges of cross-chain bridging while maintaining security and efficiency.

We developed Lanca after extensive research and experimentation with different cross-chain minimum viable products (MVPs). Through in-depth analysis, we identified that users primarily require some important attributes for cross-chain transactions: security, ease of use and the number of supported blockchains. Existing infrastructure fails to meet these requirements simultaneously, which laid the foundation for the Lanca Bridging Framework (LBF).

LBF introduces a novel Parent-Child Pool architecture, where a Parent Pool on a Master chain manages Child Pools across different networks. This design is complemented by a dynamic IOU system that allows Rebalancers to maintain liquidity equilibrium across pools, significantly improving capital efficiency. Unlike traditional approaches that require equal liquidity across all chains, LBF's model enables more efficient liquidity utilisation.

Our vision is to build a unified liquidity layer that seamlessly connects all blockchain networks, enabling efficient capital flow while maintaining rigorous security. LBF is designed to scale horizontally across any number of chains and vertically through different asset types, providing a foundation for the next generation of cross-chain applications.

2 Technical Overview

To start our vision of a unified cross-chain ecosystem, we developed the LBF by incorporating Concero V2 Messaging. This integration represents a pivotal architectural progression from the initial Concero Messaging V1, enabling LBF to be deployed on any blockchain network with considerable speed and minimal technical configuration.

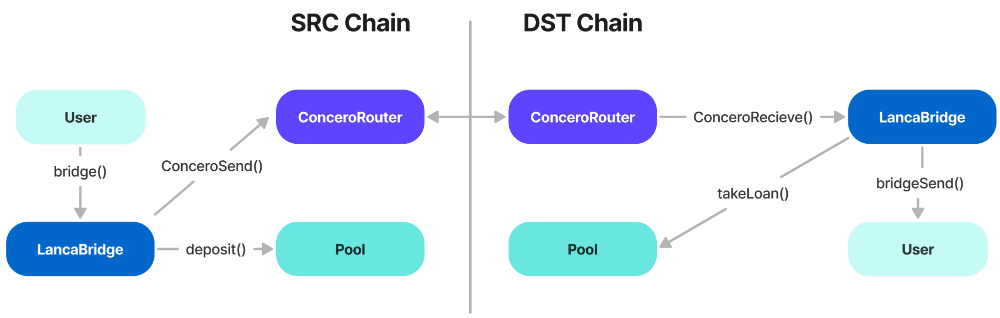

Figure 1: Lanca Bridging Framework Overview

The framework's design leverages the capabilities of Concero V2 Messaging to create an adaptable and efficient communication layer for passing messages and value between networks. This method ensures Lanca can securely connect disperate blockchains, prioritising both performance and security in value cross-chain transfer (Figure 1).

2.1 LBF's Technical Architecture

The LBF's technical architecture is designed around a Parent-Child Pool model, facilitated by Concero V2 Messaging. This structure separates the main liquidity hub (Parent Pool) from operational pools on other networks (Child Pools), enabling efficient and secure cross-chain transactions. The system integrates several automated components to manage liquidity and ensure price accuracy, as depicted in Figure 2.

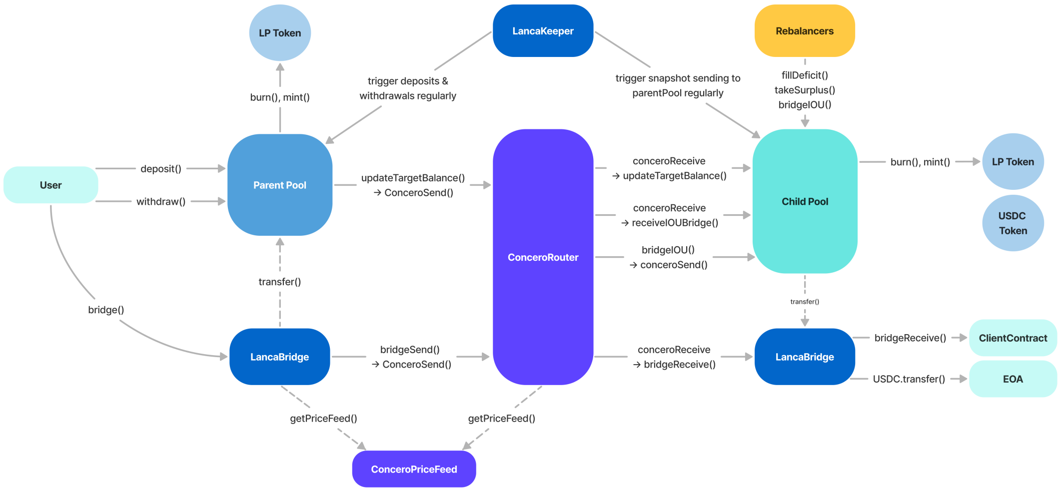

Figure 2: Cross-chain Liquidity Provision and Transaction Flow: Lanca Bridging Framework Architecture

The architecture consists of several key components:

- **Parent Pool**: The central liquidity hub on the Master Chain. It receives deposits from and processes batched withdrawals for liquidity providers.
- **Child Pool**: A pool on a destination chain that holds liquidity to facilitate immediate settlement for cross-chain transactions. Its state is regularly reported back to the Parent Pool.
- **LancaBridge**: Smart contracts on both source and destination chains that manage the logic for sending and receiving assets.
- **ConceroRouter**: The core messaging component that securely relays data and instructions between the Parent and Child Pools on different chains via Concero V2 Messaging.
- **ConceroPriceFeed**: An oracle that provides real-time asset prices to ensure fair value exchange during transactions.
- **LancaKeeper**: An automated operator that triggers time-sensitive contract calls to orchestrate the system. Its primary jobs are to: (1) prompt Child Pools to send daily liquidity snapshots to the Parent Pool; (2) initiate batched deposit and withdrawal processes on the Parent Pool; and (3) finalise withdrawal completions once the required liquidity is consolidated.

- **Rebalancers**: Independent third parties incentivised to maintain liquidity equilibrium in Parent Pool and Child Pools by filling deficits or taking surpluses.

Liquidity Provision Users can act as liquidity providers (LPs) by interacting directly with the Parent Pool.

- To deposit, a user calls deposit function on the Parent Pool. In return for their assets, the pool mints and sends them LP tokens, representing their proportional share of the total liquidity.
- To withdraw, a user calls withdraw function, which burns their LP tokens. The withdrawal is handled in a batch process orchestrated by the LancaKeeper. The detailed withdrawal process is described in the next section.

Systematic Transaction Flow When a user initiates a cross-chain transaction, the process follows a systematic flow:

- **Transaction Initiation**. A user starts a cross-chain transfer by calling the bridge function on the LancaBridge contract on the source chain. The bridge consults the ConceroPriceFeed to determine the asset's value before instructing the ConceroRouter to send the transaction details to the destination chain.
- **Cross-Chain Messaging and Execution**. The ConceroRouter securely transmits the message. On the destination chain, the corresponding LancaBridge contract receives the payload, verifies the transaction, and executes the final transfer. This can be a direct asset transfer to the user's wallet (EOA) or an interaction with a ClientContract for more complex operations.
- **Hybrid Liquidity Management**. To ensure continuous operation, the system employs the hybrid liquidity management approach. The LancaKeeper ensures daily state reporting from Child Pools to Parent Pool, to make sure Child Pools have sufficient liquidity for operations. Aditionally, Rebalancers are incentivised to execute fillDeficit or takeSurplus functions on the Child Pool, ensuring the pool has sufficient funds for outgoing transactions without being over-capitalised. This rebalancing is settled using an IOU mechanism between the pools.

2.2 Hybrid Liquidity Management System: Parent Pool and Rebalancers

The LBF implements a hybrid liquidity management system to ensure optimal pool health and operation. The first layer comprises the Parent Pool's continuous monitoring of Child Pools, dynamically adjusting their targetBalance value based on its network activity. The second layer introduces a novel risk-minimised rebalancing mechanism through an IOU-based system, where autonomous Rebalancers earn fees by maintaining equilibrium across liquidity pools.

2.2.1 Parent-Child Pool Model

The Parent-Child Pool model is central to LBF's capital efficiency. It features a Parent Pool on a designated Master Chain that serves as the primary liquidity hub, connected to multiple Child Pools across different networks. The Parent Pool acts as the main entry and exit point for liquidity, which is then distributed to Child Pools via Concero V2 Messaging.

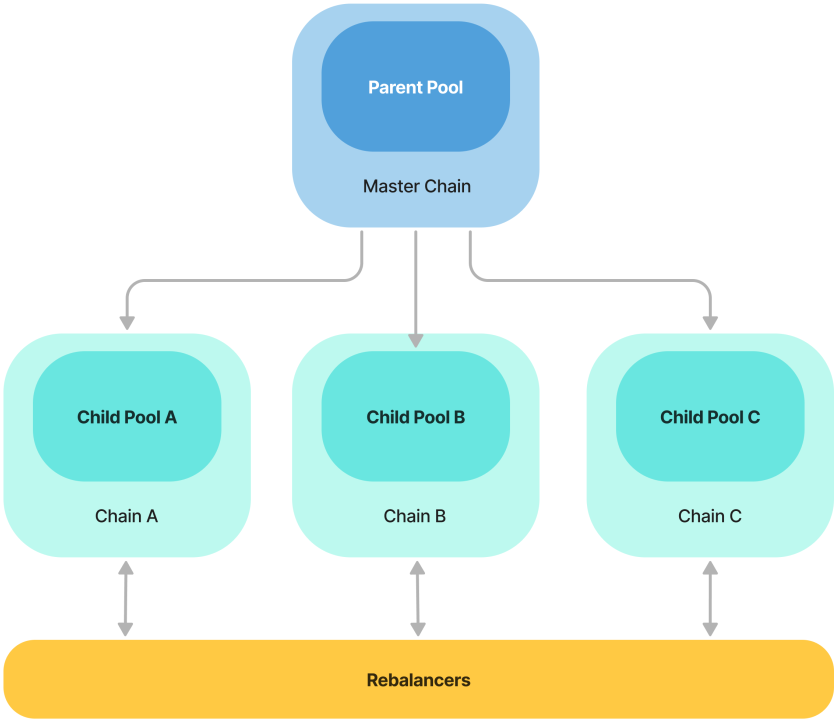

Figure 3: LBF's Parent-Child Pool Model

To manage liquidity, the Parent Pool establishes a dynamic targetBalance for each Child Pool, which is continuously adjusted based on network demand to maximise capital efficiency (Figure 3). This monitoring and communication is facilitated by the LancaKeeper. When a pool's balance deviates from its target, autonomous Rebalancers are incentivised to restore equilibrium. The detailed mechanics of this rebalancing system are presented in the subsequent section.

This design offers several key advantages. It enhances capital efficiency by reusing liquidity, which reduces overall capital requirements. Furthermore, it allows LPs to earn fees generated across all connected chains from a single deposit, while simplifying the provision process through a centralised management model with decentralised execution.

Pool structure The financial state of each pool in the LBF is determined by its ‘activeLiquidity‘ relative to its ‘targetBalance‘. As illustrated in Figure 4, ‘activeLiquidity‘ represents a pool’s operational capital and is calculated by subtracting any accumulated ‘rebalancingFees‘ from the pool’s total ‘balance‘.

$$\text{activeLiquidity} = \text{balance} - \text{rebalancingFees} \tag{1}$$

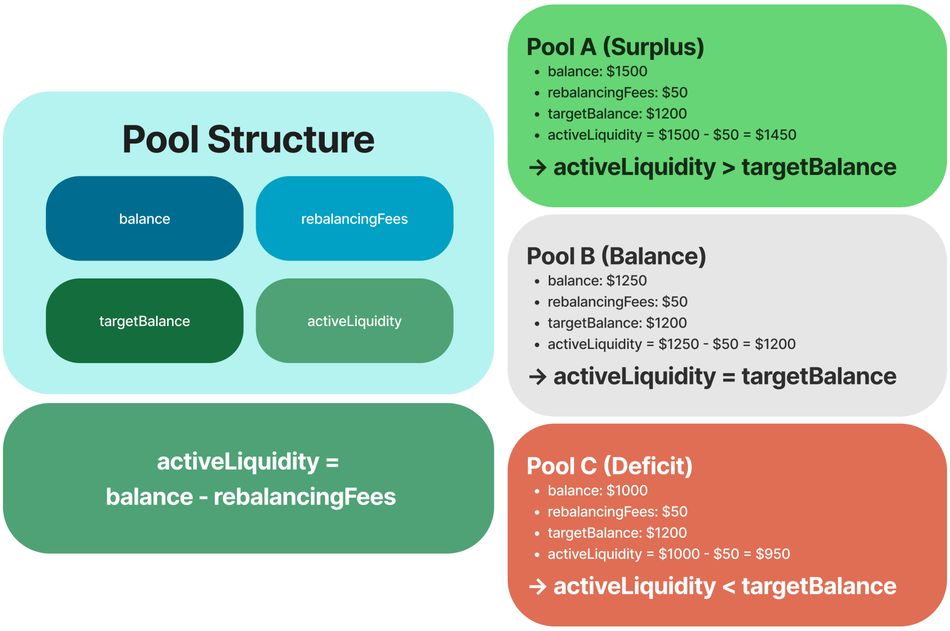

Figure 4: Pool States Determined by Comparing Active Liquidity to Target Balance

The comparison of these two values determines the pool’s status, which dictates the actions available to Rebalancers:

- **Surplus**: A pool is in surplus when its ’activeLiquidity’ is greater than its ’targetBalance’.
- **Balanced**: A pool is balanced when its ’activeLiquidity’ is equal to its ’targetBalance’.
- **Deficit**: A pool is in deficit when its ’activeLiquidity’ is less than its ’targetBalance’.

This clear distinction, exemplified by Pools A, B, and C in Figure 4, allows Rebalancers to efficiently identify and resolve liquidity imbalances across the network.

The introduction of these concepts forms the foundation of the LBF rebalancing mechanism. These variables provide a straightforward way to track each pool’s liquidity status relative to the ideal targetBalance level so that it enables more efficient and responsive rebalancing of liquidity across the network. This, in turn, helps to ensure that each pool has the liquidity it needs to function optimally without tying up excess capital unnecessarily.

2.2.2 Rebalancers

While the Parent Pool's regular rebalancing helps maintain overall system equilibrium, short-term imbalances can still occur between these scheduled top-up cycles. To address these temporary imbalances, Rebalancers act as rapid-response participants in the system. These Rebalancers complement the Parent Pool's core function by providing quick liquidity adjustments, ensuring the system maintains optimal performance even between the Parent Pool's scheduled rebalancing operations.

These complementary layers work in cooperation to maintain sufficient liquidity across the network, particularly during periods of high activity. When a child pool experiences a liquidity deficit, Rebalancers can deposit assets and receive IOUs, which can later be redeemed from pools with surplus liquidity. The faster the rebalancing occurs, the smoother and more seamless the user experience can be.

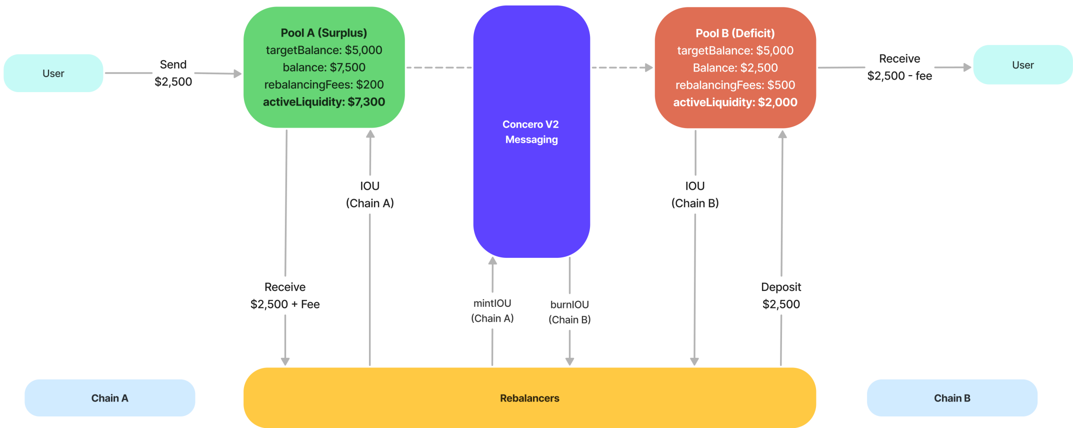

Figure 5: Rebalancers' IOU System Flow

The process works as follows (Figure 5): First, the Parent Pool establishes a targetBalance liquidity level for each pool in the system. This targetBalance value represents the ideal amount of liquidity that a pool should hold for the system to be considered balanced. Based on chains activities and volume, it is distributed sufficiently among all the pools, setting their respective targetBalance levels.

Rebalancers continuously monitor the pools, looking for those that are in a deficit state. When Rebalancers identify a pool in deficit, they can choose to deposit their own capital into that pool. In return for this deposit, the Rebalancers receive an equivalent amount of IOU tokens from the system. These IOUs represent a claim on the liquidity that the Rebalancers have injected into the deficit pool.

The existence of a deficit pool implies that there must be another pool in the system that has a surplus of liquidity. The Rebalancers can leverage this by moving their IOUs to the chain where the surplus pool exists. This movement of IOUs is facilitated by Concero V2 Messaging, which allows for the burning of IOUs on one chain and their minting on another.

Once on the chain with the surplus pool, the Rebalancers can redeem their IOUs. The redemption process involves presenting the IOUs to the surplus pool and receiving an equivalent amount of liquidity in return. This completes the rebalancing cycle, as the liquidity that was initially deposited into the deficit pool is recovered from the surplus pool on a different chain.

For their service in rebalancing liquidity across the network, the Rebalancers earn a fee. This fee serves as an incentive for Rebalancers to monitor and participate in the liquidity rebalancing process actively.

It is important to note that the role of Rebalancers is distinct from that of typical LPs. While LPs add liquidity to the system and earn fees from transactions, Rebalancers are specifically focused on identifying and correcting imbalances in liquidity distribution. Their actions help to ensure that all pools maintain an optimal level of liquidity, which is critical for the smooth operation of the LBF.

Also, IOUs can only be redeemed when a pool has a balance exceeding its targetBalance value. This is a critical design constraint that ensures the protocol's core operational liquidity remains protected. When a Rebalancer seeks to redeem IOUs, they must first verify that their target pool maintains a surplus position. If no surplus exists in their preferred chain's pool, the redemption cannot be processed, requiring the Rebalancer to either wait for a surplus to develop or seek redemption from other pools in the network that do have surplus positions.

2.2.3 Deposit and withdrawal logic

It is important to clarify the logic behind the deposit and withdrawal process in perspective of LPs. This section details the specific flow from LPs deposit into the Parent Pools, how a Parent Pool communicates with Child Pools, and how these Rebalancers deposit assets into pools with a liquidity deficit, receive IOUs representing their deposit, and later redeem those IOUs in pools with excess liquidity to retrieve their original assets plus an additional fee.

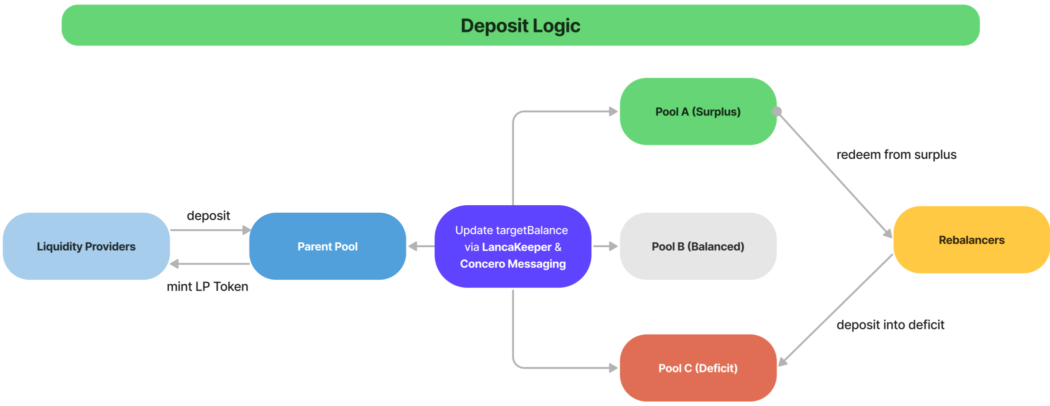

Figure 6: Deposit logic

Deposit Logic The deposit process begins with the LPs sending their assets to the Parent Pool on the Master chain. Upon receiving the deposit from LPs, the Parent Pool mints a corresponding amount of LP tokens to the LPs. These LPs tokens represent the provider's share of the total liquidity in the LBF system. The minting of tokens follows a formula that considers the size of the deposit relative to the existing liquidity in the pools (Figure 6).

$$\text{LP Share} = \frac{\text{User's Deposited Amount}}{\text{Total Pool Value}} \times 100\% \tag{2}$$

$$\text{Share Tokens} = \frac{\text{Deposit Amount} \times \text{Current Total Share Tokens}}{\text{Current Pool Value}} \tag{3}$$

Next, the Parent Pool will monitor and update the targetBalance value for each of the connected Child Pools, including itself. As aforementioned, the targetBalance value represents the target liquidity level that each pool should maintain to ensure sufficient distribution. The update and communication between the Parent Pool and Child Pools are facilitated via Concero V2 Messaging and LancaKeeper.

Once there are depositing transactions happening on chains, it leads to the state that the Child Pools are in a deficit state, they now require additional liquidity to meet their new targetBalance. This is where the Rebalancers comes into play. The Rebalancers identify pools with a deficit and deposits their own assets into those pools to help rebalance the liquidity. In return for their deposit, the Rebalancers received an IOU from the pool. These IOUs serve as proof that the Rebalancers can later redeem their deposited assets plus a fee from the Parent Pool with excess liquidity.

Withdrawal Logic The withdrawal process in LBF is designed to be both secure and efficient, with a dynamic rebalancing-driven mechanism. When a LP initiates a withdrawal, they burn their LP tokens at the Parent Pool, triggering a coordinated rebalancing across the network. This action immediately signals a liquidity need by temporarily increasing the Parent Pool's 'targetBalance' by the requested withdrawal amount, placing it into a deficit state.

This engineered deficit serves as a crucial signal to Rebalancers, who are economically incentivised to restore equilibrium. They identify the Parent Pool's need and deposit liquidity into the Parent Pool, its balance begins to rise toward the new 'targetBalance'.

Once the pool's balance is replenished and meets the adjusted target, the LancaKeeper automatically executes the final withdrawal, transferring the assets to the LP. This entire process (Figure 7) prevent the compromisations of the protocol's operational liquidity. The withdrawal time is not bound by an arbitrary delay but is instead dependent on the speed and efficiency of network-wide rebalancing, creating a more responsive and capital-efficient system.

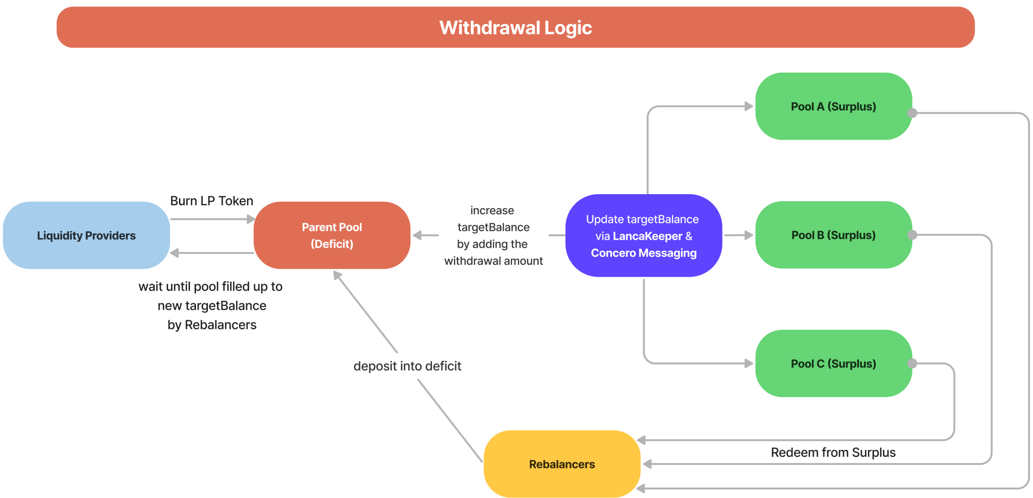

Figure 7: The LBF Withdrawal Process

2.3 Dynamic Liquidity Rebalancing Algorithm

This section outlines LBF's liquidity rebalancing algorithm for dynamically adjusting the target liquidity (referred to as the targetBalance value) of child pools based on their liquidity health. The goal is to reallocate liquidity from pools that are overperforming to those under stress while keeping the overall system liquidity constant.

In LBF, liquidity is distributed across multiple child pools. Over time, some pools may not use it liquidity as efficiently as others. To ensure effective liquidity management, our algorithm now calculates a *Liquidity Health Score (LHS)* comprised of two components:

- A *dynamic usage component* based on the *Liquidity Utilisation Ratio (LUR)*, which captures how actively a pool's liquidity is being used.
- A *dynamic flow component* based on the *Net Drain Rate (NDR)*, which tracks the

net outflow (relative to targetBalance).

Based on the LHS, the Parent Pool will adjust each pool's target liquidity (targetBalance) so that child pools experiencing stress receive additional liquidity, while those operating above target release a portion of their liquidity. The overall total liquidity remains constant.

2.3.1 Key Metrics in the Rebalancing Algorithm

As a dynamic usage component, *Liquidity Utilisation Ratio (LUR)* is a crucial metric that measures the ratio of the combined inflow and outflow of a pool relative to its targetBalance value. This measurement helps capture market or usage intensity. The formula for LUR is calculated as follows:

$$LUR = \frac{\text{Total Volume}}{\text{targetBalance Value}} = \frac{\text{Inflow} + \text{Outflow}}{\text{targetBalance Value}} \tag{4}$$

Because the combined flow is expected to be much higher than the targetBalance value, we normalise LUR using a tunable parameter (K):

$$\text{LUR\_Score} = 1 - \frac{\text{LUR}}{K + \text{LUR}} \tag{5}$$

- **High LUR_Score (close to 1)**: it means the total volume (Inflow + Outflow) is low relative to the targetBalance value. The pool is experiencing low activity and is not under significant stress. Little or no adjustment to the targetBalance target is needed.

- **Low LUR_Score (close to 0)**: it means the total volume is very high compared to the targetBalance value. The pool is heavily used and is likely under stress. This signals that the pool may require an increased targetBalance target to handle the high transaction volume.

The constant (K) in the LUR normalisation formula controls how fast the score declines as activity increases. The (K) can be set based on historical data or simulation and iteration.

Especially, (K) can be based on the sensitivity of the pool. In terms of Sensitivity:

- A *lower (K)* makes the score drop faster (more sensitive), so even moderate activity yields a low score.

- A *higher (K)* reduces the effect of high volume, requiring extreme activity for a similar score drop.

Based on these factors, it ensures that (K) is well-calibrated to reflect true pool stress and drive appropriate liquidity adjustments.

As a dynamic flow component, *Net Drain Rate (NDR)* is a crucial metric that measures the balance between outflows and inflows in a liquidity pool, normalised by its targetBalance Value. This measurement helps monitor the pool's liquidity stability and potential stress points. The formula for NDR is calculated as follows:

$$NDR = \frac{\text{Outflow} - \text{Inflow}}{\text{targetBalance Value}} \tag{6}$$

The NDR quantifies the net effect of liquidity movements:

- NDR ≤ 0: Indicates no net drain or a net inflow, which is ideal.
- NDR > 0: Reflects that withdrawals are exceeding deposits, signaling liquidity stress.

To make this metric more meaningful and comparable, we transform the NDR into a standardised score. NDR is transformed into a score as follows:

$$NDR\_Score = \begin{cases} 1, & \text{if } NDR \leq 0 \\ \max(0, 1 - NDR), & \text{if } NDR > 0 \end{cases} \tag{7}$$

- When the NDR is less than or equal to zero (NDR ≤ 0), the pool receives a perfect score of 1, reflecting optimal liquidity conditions.
- For any positive NDR values, the score follows a linear decrease using the formula max(0, 1 - NDR). This means that as the drain rate increases, the score proportionally decreases, but never falls below 0. The maximum function ensures that even in extreme cases of high drain rates, the score maintains a minimum value of 0.

2.3.2 Liquidity Health Score (LHS)

The two sub-scores are combined into a single *Liquidity Health Score (LHS)* using weight-based averaging. The formula for LHS is as follows:

$$LHS = W_1 \times LUR_Score + W_2 \times NDR_Score \tag{8}$$

At the first iteration, we set the weights as 0.7 for LUR and 0.3 for NDR, supposing that the dynamic usage component is more important than the dynamic flow component. This weight will be adjusted in future iterations based on the performance of the pool during testnet. The formula for LHS is as follows:

$$LHS = 0.7 \times LUR_Score + 0.3 \times NDR_Score \tag{9}$$

- A higher LHS (closer to 1) indicates a well-balanced pool.
- A lower LHS signals liquidity stress.

2.3.3 Dynamic Rebalancing Mechanism

Based on the LHS, we adjust each pool's target liquidity (targetBalance) to reallocate funds cumulatively while ensuring that the overall liquidity remains constant. The steps are as follows:

Step 1: Define a Weighting Function We use a simple function to calculate a weighting factor $f(LHS)$, which increases a pool's weight if its LHS is below 1:

$$f(LHS) = 1 + \alpha \cdot (1 - LHS) \tag{10}$$

where:

- α is a sensitivity parameter, e.g., $\alpha = 1$.
- A lower LHS results in a higher $f(LHS)$, meaning that a pool under stress "deserves" additional liquidity.

Note: The alpha (α) parameter is a crucial control mechanism in our dynamic liquidity rebalancing system that determines how aggressively the system responds to liquidity imbalances.

When a pool's Liquidity Health Score (LHS) deviates from the ideal value of 1:

- A higher α causes more aggressive rebalancing responses to liquidity stress
- A lower α results in more gentle, conservative adjustments

Step 2: Calculate the Weight for Each Pool For each pool (i), the weight (W_i) is given by:

$$W_i = \text{Original targetBalance}_i \times f(LHS_i) \tag{11}$$

Let the total weight for all (N) pools be:

$$W_{\text{total}} = \sum_{i=1}^{N} W_i \tag{12}$$

Step 3: Compute the New Target Liquidity (New targetBalance) Each pool's new target liquidity is determined by:

$$\text{New targetBalance}_i = \frac{W_i}{W_{\text{total}}} \times \text{Total Liquidity} \tag{13}$$

This formula ensures that while each pool's target is adjusted, the overall liquidity remains constant.

2.3.4 Example of Rebalancing 5 Pools

Consider a system with 5 pools and a total system targetBalance of $500k. The initial pool states are as follows:

Table 1: Pool States and 24-hour Flows (Currency: USDC)

Pool	Current Balance	targetBalance Value	Outflow (24h)	Inflow (24h)
A	$120k$	$100k$	$80k$	$60k$
B	$85k$	$100k$	$150k$	$140k$
C	$95k$	$100k$	$200k$	$180k$
D	$110k$	$100k$	$40k$	$50k$
E	$90k$	$100k$	$90k$	$70k$

Step 1: Calculate the LUR and LUR Score Using the formula:

$$LUR = \frac{\text{Inflow} + \text{Outflow}}{\text{targetBalance Value}} \tag{14}$$

and with (K=5):

$$LUR\_Score = 1 - \frac{LUR}{5 + LUR} \tag{15}$$

For each pool:

Table 2: LUR Calculations (Currency: USDC)

Pool	LUR Calculation	LUR Value	LUR Score
A	$(80k + 60k)/100k$	1.40	0.78
B	$(150k + 140k)/100k$	2.90	0.63
C	$(200k + 180k)/100k$	3.80	0.57
D	$(40k + 50k)/100k$	0.90	0.85
E	$(90k + 70k)/100k$	1.60	0.76

Step 2: Calculate NDR and NDR_Score For each pool, using:

$$NDR = \frac{\text{Outflow} - \text{Inflow}}{\text{targetBalance Value}} \tag{16}$$

and

$$\text{NDR\_Score} = \begin{cases} 1, & \text{if } NDR \leq 0 \\ 1 - NDR, & \text{if } NDR > 0 \end{cases} \tag{17}$$

Table 3: NDR Calculations (Currency: USDC)

Pool	NDR Calculation	NDR Value	NDR Score
A	$(80k - 60k)/100k$	0.20	0.80
B	$(150k - 140k)/100k$	0.10	0.90
C	$(200k - 180k)/100k$	0.20	0.80
D	$(40k - 50k)/100k$	-0.10	1.00
E	$(90k - 70k)/100k$	0.20	0.80

Step 3: Calculate the Liquidity Health Score (LHS) We combine the scores with a 70/30 weighting:

$$LHS = 0.7 \times LUR\_Score + 0.3 \times NDR\_Score \tag{18}$$

Table 4: LHS Calculations (Currency: USDC)

Pool	LHS Calculation	LHS Score
A	$0.7 \times 0.78 + 0.3 \times 0.80$	0.79
B	$0.7 \times 0.63 + 0.3 \times 0.90$	0.71
C	$0.7 \times 0.57 + 0.3 \times 0.80$	0.64
D	$0.7 \times 0.85 + 0.3 \times 1.00$	0.90
E	$0.7 \times 0.76 + 0.3 \times 0.80$	0.77

Step 4: Calculate Weighting Factors Using the weighting function (with $\alpha = 1$):

$$f(LHS) = 1 + (1 - LHS) \tag{19}$$

Table 5: Weighting Factors (Currency: USDC)

Pool	f(LHS) Calculation	f(LHS) Value	Weight Calculation	Weight
A	$1 + (1 - 0.79)$	1.21	$100k \times 1.21$	$121k$
B	$1 + (1 - 0.71)$	1.29	$100k \times 1.29$	$129k$
C	$1 + (1 - 0.64)$	1.36	$100k \times 1.36$	$136k$
D	$1 + (1 - 0.90)$	1.10	$100k \times 1.10$	$110k$
E	$1 + (1 - 0.77)$	1.23	$100k \times 1.23$	$123k$

Total Weight $= \$121k + \$129k + \$136k + \$110k + \$123k = \$619k$

Step 5: Calculate New targetBalance Values Each pool's new targetBalance is given by:

$$\text{New targetBalance}_i = \frac{W_i}{\text{Total Weight}} \times \text{Total Liquidity} \tag{20}$$

Table 6: New targetBalance Values (Currency: USDC)

Pool	Calculation	Final Allocation
A	$(121k/619k) \times 500k$	$97.8k$
B	$(129k/619k) \times 500k$	$104.2k$
C	$(136k/619k) \times 500k$	$109.9k$
D	$(110k/619k) \times 500k$	$88.9k$
E	$(123k/619k) \times 500k$	$99.4k$

These new targets ensure that pools with higher activity (lower LHS) receive increased liquidity, while more stable pools see a reduction.

2.3.5 Results

The rebalancing algorithm efficiently redistributes liquidity based on pool activity and stress levels. Looking at the changes in targetBalance values:

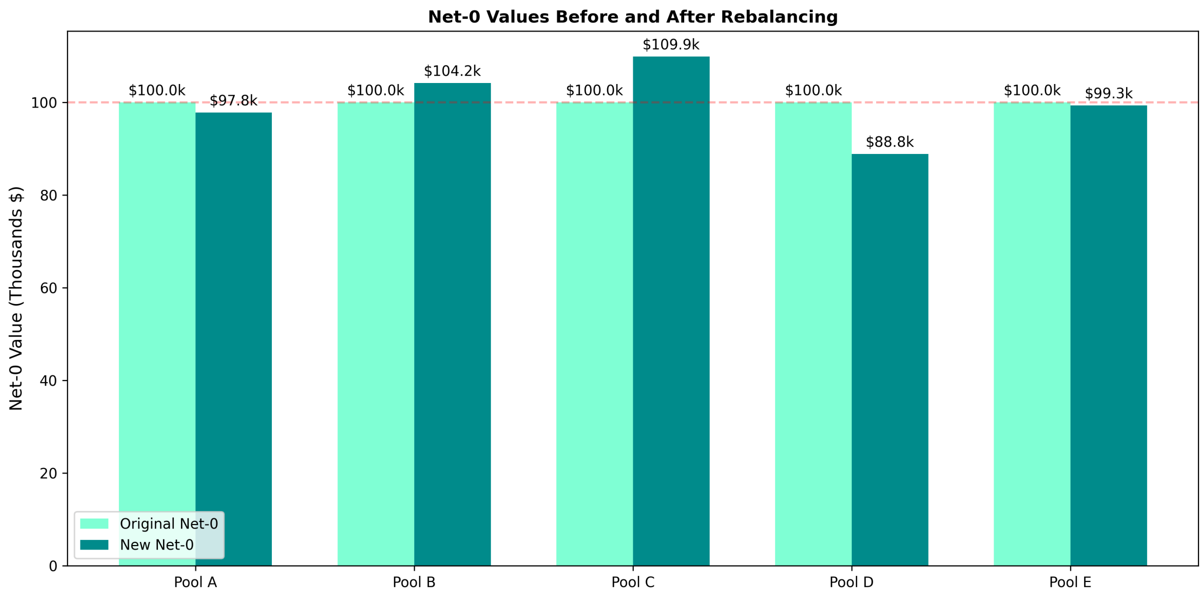

Figure 8: targetBalance Values Before and After Rebalancing

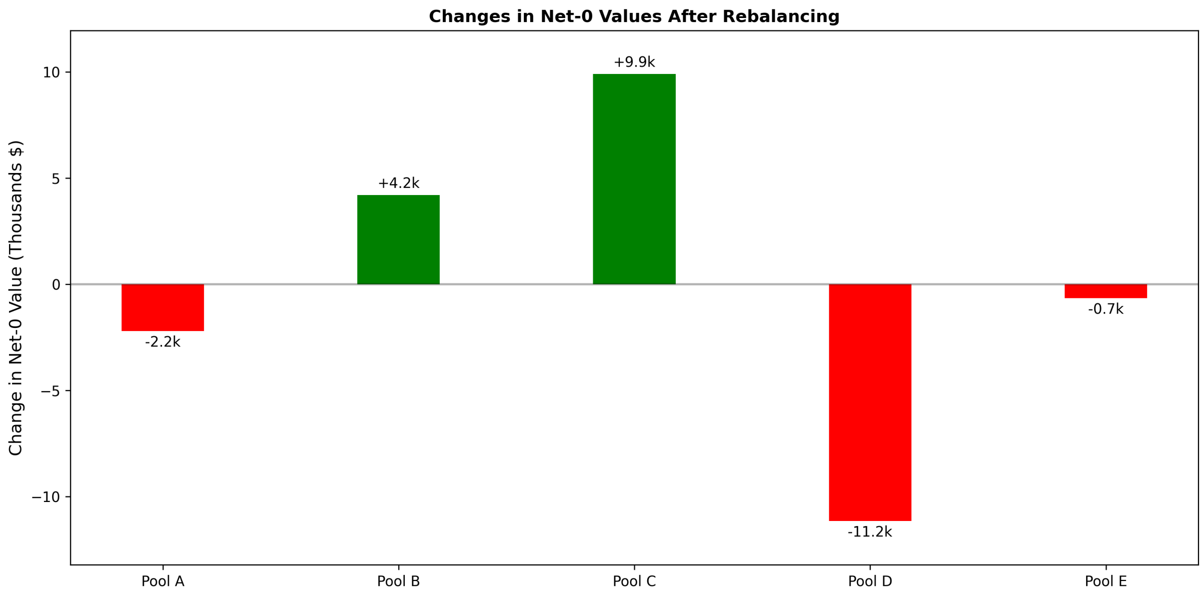

Figure 9: Changes in targetBalance Values After Rebalancing

- **Pool C** received the largest increase (+$9.9k) due to its high transaction volume ($380k total volume), indicating the algorithm correctly identified its need for additional liquidity.
- **Pool B** also gained liquidity (+$4.2k) as it showed significant activity ($290k total volume).
- **Pool D** saw the largest reduction (-$11.2k) as it demonstrated the lowest activity ($90k total volume) and had positive net inflow.
- **Pools A and E** experienced minor adjustments (-$2.2k and -$0.7k, respectively) reflecting their moderate activity levels.

2.3.6 Conclusion

The results of our rebalancing algorithm demonstrate its effectiveness in optimising liquidity across the system. The algorithm successfully accomplishes three key objectives. First, it accurately identifies which pools are experiencing stress from high transaction volumes, ensuring that pools under pressure are detected. Second, it carefully moves liquidity from pools with lower activity to those with higher demand, creating a more balanced system. Finally, it preserves the total amount of liquidity in the system while improving how that liquidity is distributed across different pools.

By using *LUR (Liquidity Utilisation Ratio)* and *NDR (Net Drain Rate)* as our key metrics, we can accurately measure both how active each pool is and its current stress level. These measurements then guide our rebalancing decisions, ensuring that liquidity flows to where it's needed most. The final allocation numbers confirm that our approach leads to meaningful and appropriate adjustments in pool liquidity levels.

3 Incentivisation Mechanism of Underserved chains

LBF's vision is to support all blockchain networks, ensuring extensive coverage across diverse chains. However, this broad support inherently results in varying activity levels across different chains, with some experiencing higher transaction volumes than others. Consequently, certain liquidity pools may remain in deficit for extended periods.

To address this, LBF implements a *dynamic bounty mechanism* that incentivises rebalancing activities, particularly targeting underserved chains. This system encourages *Rebalancers* to maintain balanced liquidity across all supported networks, regardless of transaction volumes or market activity. In return, they receive a bounty in the form of the *protocol native token*.

3.1 The Bounty Calculation Mechanism

The foundation of the bounty calculation mechanism is built on four essential components that work together to create a balanced and adaptable incentive system:

$$Reward = Deficit \times TimeFactor \times UtilisationFactor \times GovernanceMultiplier \tag{21}$$

3.2 LBF Progressive Reward Model

Based on the foundation bounty formula, we developed LBF's Progressive Reward Model. This model introduces a novel approach to incentivising liquidity rebalancing while maintaining protocol sustainability.

The core formula balances three critical dimensions:

$$B_{prog} = D_{prog} \times \underbrace{(T^p \times r)}_{\text{Time Weight}} \times \underbrace{\left[1 + \frac{(U - U_{\text{base}})}{100} \times k\right]}_{\text{Utilisation Factor}} \times \underbrace{G}_{\text{Governance}} \tag{22}$$

It is designed to address the dual challenges of immediate deficit resolution and long-term token emission control, this mechanism employs time-weighted urgency factors ($T^p \times r$) and utilisation-sensitive multipliers ($\left[1 + \frac{(U - U_{\text{base}})}{100} \times k\right]$) within a mathematically constrained framework.

Let's break down the formula:

1. **Reward** B: Reward in native token
2. **Base Deficit** D: Represents the immediate USDC value needed to balance the pool. This ensures rewards directly correspond to the protocol's liquidity requirements.
3. **Time Weight** ($T^p \times r$): Creates gradual early growth that accelerates over time, balancing urgency with emission control.
    - T: Hours since the deficit started

- p: Time exponent ($0 < p < 1$) controlling growth speed (0.5 by default)
- r: Growth rate coefficient

4. **Utilisation Factor** $[1 + (U - U_{\text{base}})/100 \times k]$: Increases rewards when utilisation exceeds safe levels, protecting pool health.
   - U: Current pool utilisation percentage, e.g. 80%
   - U_{base}: a threshold utilisation percentage where rewards escalate, e.g. 65%
   - k: Sensitivity factor
   - G: Governance multiplier
5. **Governance Multiplier** G: DAO-controlled parameter for system-wide reward adjustments.

As LBF will transition to a DAO-driven protocol, the DAO will be able to adjust the parameters to suit the needs of the community. The DAO control parameters are listed below. However, at the initial stage, the Concero team will set the parameters to ensure the protocol is sustainable and stable.

Table 7: DAO Control Parameters

Parameter	Purpose	Adjustment Range
r	Speed of time-based growth	0.1% – 10% per hour
U_{base}	Utilisation starting point	50% – 75%
k	Utilisation sensitivity	1 – 3
G	Global reward scale	0.5 – 1.5

3.3 Reward Model Comparison

Before coming up with the progressive reward model, we have tested the linear and exponential growth models for the bounty mechanism. The formula for the linear growth and exponential growth are as follows:

Linear Rate (r): Constant hourly growth coefficient

$$B_{lin} = D \times \underbrace{(1 + rT)}_{\text{Linear Growth}} \times \underbrace{\left[1 + \frac{(U - U_{base})}{100} \times k\right]}_{\text{Utilisation Factor}} \times G_{lin} \tag{23}$$

Growth Coefficient (λ): Continuous compounding rate

$$B_{exp} = D \times \underbrace{e^{\lambda T}}_{\text{Exponential Growth}} \times \underbrace{\left[1 + \frac{(U - U_{base})}{100} \times k\right]}_{\text{Utilisation Factor}} \times G_{exp} \tag{24}$$

In our testing, we found significant issues with traditional reward models that use either linear or exponential calculations. When rewards grow in a straight line (linear) or multiply exponentially over time, they create two main problems: they give out too many rewards too quickly in the beginning, and they continue growing steadily even after 24 hours, making them unsustainable in the long term.

To address these limitations, we developed a pioneering progressive reward model with three major improvements. At its core, our model uses a special mathematical approach called sub-linear temporal scaling, where rewards grow more slowly over time instead of increasing at a constant or exponential rate. This creates a better balance between rewarding quick responses and maintaining controlled, sustainable growth over longer periods. We retained the existing system that adjusts rewards based on participation levels, which helps maintain the health of the reward pool.

Additionally, we introduced four carefully chosen governance parameters that allow precise fine-tuning of the system: a base rate, a baseline utilisation level, a utilisation sensitivity and a governance factor.

This enhanced approach represents a significant advancement in Lanca's incentive system. By addressing the fundamental weaknesses of traditional models while maintaining stability and flexibility in governance, our progressive reward model offers a more sustainable and adaptable solution for the long-term success of the protocol.

3.4 Model Analysis

3.4.1 Comparative Analysis between Linear, Exponential and Progressive Model

To evaluate the performance of the three models, we set the following test parameters:

- Base deficit: $1,000
- Current pool utilisation: 85%
- Utilisation threshold: 65%
- Time range: 1 to 48 hours

3.4.2 Model Configurations

Table 8: Model Configurations

Model	Growth Mechanism	Key Parameters
Progressive	Square root scaling	$p = 0.5$, $r = 5\%$, $G = 0.5$
Linear	Constant hourly rate	$r = 5\%$, $G = 0.5$
Exponential	Compounding growth	$\lambda = 5\%$, $G = 0.5$

3.4.3 Key Findings

Our detailed analysis demonstrates the clear advantages of the progressive reward model for managing and incentivising Rebalancers. Through experimental testing over a 48-hour

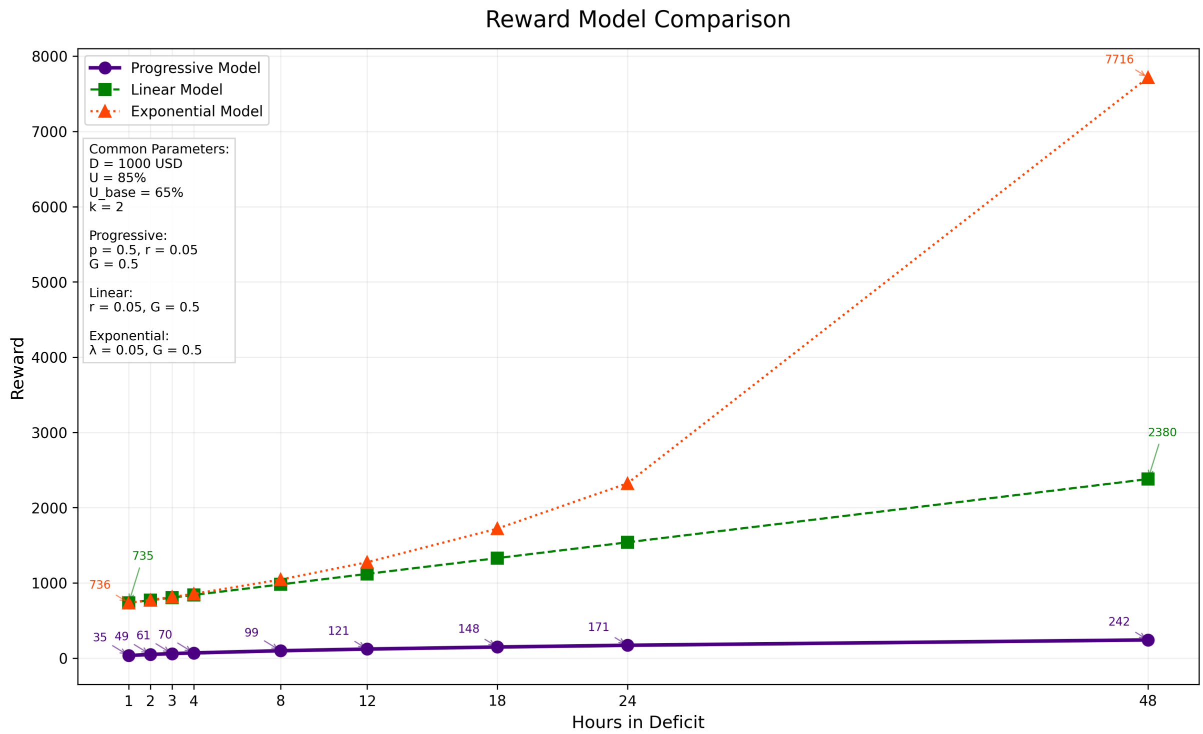

Figure 10: Reward Model Comparison: Linear, Exponential and Progressive Model

period, we observed distinct performance patterns that highlight the progressive model's superior ability to maintain controlled scaling compared to conventional approaches.

The progressive model implements a carefully calibrated reward structure that begins at 35 tokens (representing 3.5% of the base deficit) in the first hour and scales to 242 tokens (24.2% of deficit) by the 48-hour mark. This structured approach achieves remarkable efficiency, maintaining emissions at 96% below the exponential model while ensuring that rebalancing operations remain economically viable. A key strength of the model lies in its decreasing hourly growth rate, which declines from 6.54 tokens per hour to 2.08 tokens per hour, successfully encouraging prompt action while deterring speculative behavior.

From a protocol sustainability perspective, the progressive model exhibits exceptional risk management characteristics. While traditional models show concerning growth patterns—with exponential and linear models reaching multipliers of 7.72x and 2.38x respectively at 48 hours—our progressive model maintains a conservative 0.24x multiplier. This controlled growth approach successfully protects protocol reserves while preserving adequate incentives for timely liquidity provision. Notably, the protocol allows the DAO to adjust these rates based on community needs and market conditions.

The model's potency is driven by strategic mechanisms. First, it provides economically viable immediate rewards of 35 tokens within the first hour. Second, its anti-speculative design limits final rewards to just 6.9% of what an exponential model would provide. Third, it ensures predictable maximum outflows, capped at $\sqrt{48}$ multiplied by 35 tokens. These features, paired with the model's decreasing marginal reward rate, establish a comprehensive framework that aligns the interests of Rebalancers with the network's overall health requirements.

Our experimental results conclusively demonstrate that the progressive model successfully balances the dual imperatives of immediate liquidity provision and long-term protocol sustainability, marking a significant advancement in incentivisation strategies.

3.4.4 Flexibility of the Progressive Model

To further demonstrate the flexibility of the progressive model, we have simulated the reward curve for different settings of the growth speed control, utilisation alert level and global reward scale. The parameters are listed in the figure below.

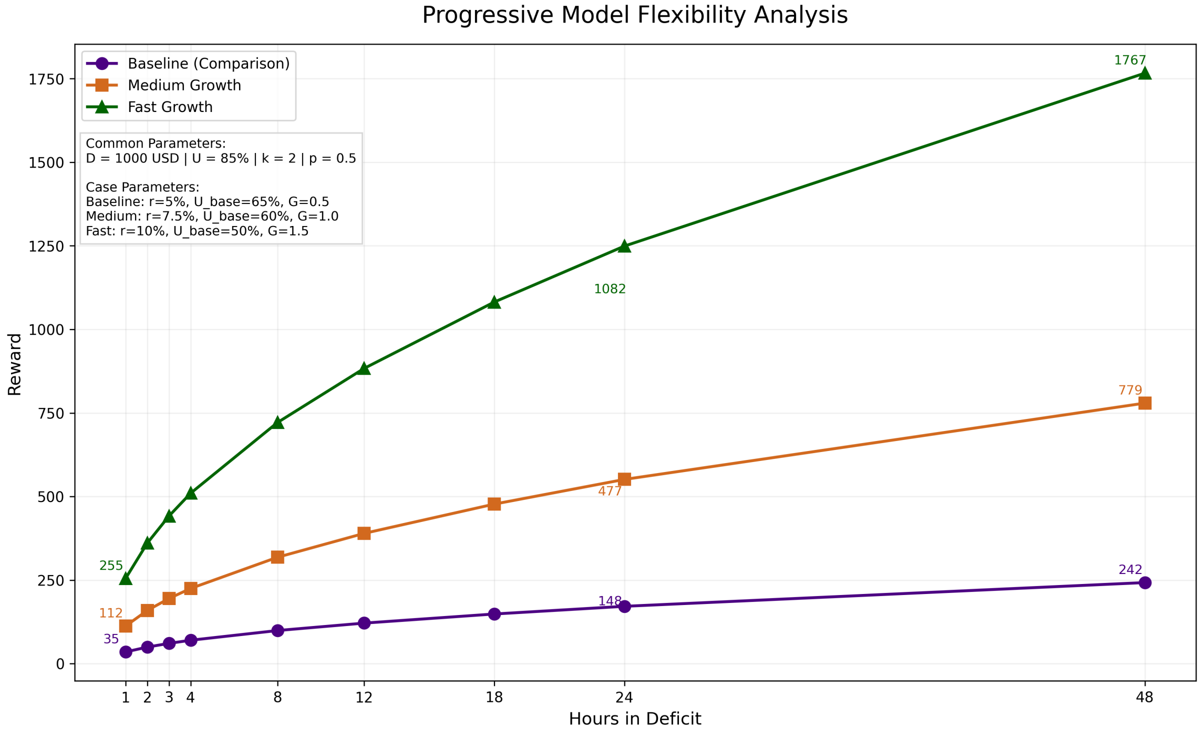

Figure 11: Flexibility of Progressive Model

3.4.5 Growth Reward and Growth Pattern

Table 9: Growth Reward and Growth Pattern

Configuration	1hr Reward	48hr Reward	Growth Pattern
Conservative	$35	$242	6.9x
Balanced	$112	$779	6.9x
High-Incentive	$255	$1,767	6.9x

The progressive model incorporates flexible configuration options that enable communities to adjust reward levels while maintaining predictable and controlled growth patterns. Through three key adjustable parameters, the system delivers precise control over incentive distribution while preserving its fundamental stability characteristics.

The first parameter, *Growth Speed Control*, allows adjustment of the base rate between 0.1% and 10%, proportionally scaling rewards across all time periods. Under conservative settings at 5%, a $1,000 deficit generates 35 tokens in the first hour. Increasing the rate to 7.5% triples early rewards to 112 tokens, while a 10% setting provides 255 tokens. Importantly, these adjustments maintain the same controlled growth curve regardless of the chosen rate.

The second parameter, *Utilisation Alert Level*, defines the threshold at which bonus rewards become active. Communities can set this between 50% and 75%, with lower thresholds providing earlier warning signals for developing liquidity shortages, while higher thresholds concentrate rewards on critical emergency situations. This flexibility allows protocols to fine-tune their response to market conditions.

The third parameter, *Global Reward Scale*, acts as a universal multiplier that can be set between 0.5x and 1.5x, proportionally adjusting all reward distributions. At maximum settings of 1.5x, 48-hour rewards can reach 1,767 tokens while preserving the same growth patterns observed in baseline configurations.

A notable characteristic of this system is that all configurations maintain an identical 6.9x growth ratio from 1-hour to 48-hour rewards. This consistency is achieved through three key design principles: the preservation of core square root temporal scaling, parameter adjustments that only affect reward magnitudes, and predictable protocol cost growth.

This carefully engineered flexibility enables communities to safely increase rewards during periods of high demand without risking unsustainable exponential growth. Through a democratic governance process, DAO members can adjust these settings via simple majority voting, ensuring the system remains responsive to market needs while maintaining long-term protocol sustainability.

4 Security and risk mitigation

The Lanca Bridging Framework's cross-chain liquidity management system introduces several security considerations that require comprehensive mitigation strategies.

4.1 Critical Risk Vectors

The framework's Parent-Child Pool architecture introduces notable security challenges, particularly related to centralisation risks. The Parent Pool, serving as the primary liquidity management hub, is a critical point of vulnerability where a breach could impact the entire network of Child Pools. This design requires strong security measures to prevent widespread issues.

Cross-chain communication dependencies facilitated through Concero V2 Messaging, introduce potential vulnerabilities in message verification and transaction settlement. These systems must maintain strict cryptographic integrity to prevent message manipulation that could disrupt asset flows between chains.

The IOU mechanism, while essential for liquidity rebalancing, presents a critical security challenge in its burn-and-mint process. To prevent unauthorised asset creation, the system must implement strict nonce validation for all cross-chain messages. A nonce, or 'number used once,' acts as a unique transaction identifier to prevent replay attacks. Without it, a malicious actor could capture a valid message for minting an IOU on a destination chain and resubmit it, tricking the protocol into minting duplicate IOUs. This would constitute a double-spend of the original burned asset, leading to severe value exploitation.

4.2 Mitigation Strategy

The framework implements a comprehensive security architecture combining technical controls, economic incentives, and operational safeguards:

Technical Controls: To mitigate the centralisation risk of the Parent Pool, all administrative functions are secured by a multi-signature (multisig) validation protocol. This could require a quorum of M-of-N independent keyholders to approve any critical changes, such as contract upgrades or parameter adjustments, preventing any single point of failure or unilateral malicious action. Cross-chain messaging implements threshold signature schemes and mandatory confirmation periods to ensure transaction integrity. Finally, all smart contracts undergo rigorous formal verification and extensive testing to mathematically prove their correctness and eliminate vulnerabilities before deployment.

Economic Security: Rebalancers might maintain bonded collateral subject to slashing conditions, creating strong disincentives for malicious behaviour. The dynamic bounty system adjusts incentives based on network conditions to maintain balanced liquidity distribution.

Operational Safeguards: Continuous monitoring systems track transaction patterns and liquidity movements to detect abnormal activity. The governance DAO maintains the ability to adjust security parameters rapidly in response to emerging threats.

This multi-layered security approach ensures the framework's resilience while maintaining operational efficiency in facilitating cross-chain liquidity management. Regular security assessments and protocol upgrades ensure the system remains protected against evolving threats in the blockchain ecosystem.

5 Implementation

The LBF implementation offers seamless, permissionless on-chain integration, enabling developers to easily add cross-chain bridging to their applications. By simplifying complex interactions, LBF fosters a flexible, decentralised, and secure bridging solution, boosting project liquidity and promoting growth across diverse blockchain networks.

6 Governance and tokenomics

LBF is a community-driven protocol, and the governance is vested in the DAO. The DAO is responsible for making decisions on the protocol's direction, including the protocol's parameters and upgrades. At the moment, there is no finalised governance and tokenomics model for LBF. However, the Concero team will be responsible for the initial governance of LBF.

7 Conclusion

The Lanca Bridging Framework directly confronts the critical challenge of capital inefficiency that limits the potential of today's multi-chain ecosystem. Its core innovation is a hybrid liquidity management system that combines the architectural strength of a Parent-Child Pool model with the dynamic agility of an IOU-based rebalancing mechanism.

This unique design ensures that liquidity is not fragmented or idle but is actively directed to where it is most needed, driven by autonomous Rebalancers who are economically incentivised to maintain network equilibrium. The result is a highly capital-efficient protocol that can support robust cross-chain transfers without demanding excessive liquidity. Secured by advanced cross-chain messaging and a multi-layered security strategy, LBF provides a resilient, scalable, and permissionless foundation for a truly unified blockchain landscape.

Acknowledgements

We express our gratitude to Chainlink Labs for their invaluable support and insightful feedback on the protocol architecture. We also extend special thanks to the core team at Concero for their significant contributions and valuable input to the whitepaper.

Author Contributions

A.B., O.K., N.G. and P.L. designed the architecture of the Lanca Bridging Framework. P.L. wrote the first draft of the whitepaper. O.K., A.B., and N.G. reviewed and edited the whitepaper. P.L. learned the technical concepts from A.B. and O.K., compiled the information, prepared the figures, and wrote the complete whitepaper.

References

[1] CoinGecko. *Top Layer 1 (L1) Coins by Market Cap.* https://www.coingecko.com/en/categories/layer-1. Accessed: 6 Feb. 2025. 2024.

[2] CoinGecko. *Top Layer 2 (L2) Coins by Market Cap.* https://www.coingecko.com/en/categories/layer-2. Accessed: 6 Feb. 2025. 2024.

[3] Swift. *Successful blockchain experiments unlock potential of tokenisation.* https://www.swift.com/news-events/news/successful-blockchain-experiments-unlock-potential-tokenisation. 2023.

[4] OKX. *Our Research Brief with Economist Impact is Live: Digital Assets as the New Alternative for Institutional Investors.* https://www.okx.com/learn/okx-economist-impact-report. Accessed: 6 Feb. 2025. 2024.