Mind the Gap: Liquidity Poisoning Attacks on Multihop Cross-Chain Arbitrage

Prerna Arote
Distributed Systems and Networks Group,
IMDEA Networks Institute,
Madrid, Spain
Email: prerna.arote@networks.imdea.org

***Abstract*—Cross-chain bridges enhance DeFi liquidity but introduce temporal gaps that expose multihop sequence-dependent arbitrage (SDA) to adversarial manipulation. Prior work attributes the rarity of SDA primarily to latency and transaction costs. We hypothesize that adversarial fragility is an additional factor and demonstrate this in 10 real multihop SDA paths from Allium data: bridge delays create predictable windows in which intermediate liquidity can be manipulated.**

We introduce Multihop SDA Liquidity Poisoning (MSLP), an attack model that exploits cross-chain asynchrony to degrade arbitrage profits without requiring transaction reordering or violating execution constraints. In our dataset, MSLP requires median capital of USD 21 to render 90% of paths unprofitable, reducing mean profit from USD 28.42 to USD -3.29. These results suggest that, beyond execution efficiency, the robustness of intermediate market states plays a critical role in the viability of cross-chain arbitrage.

***Index Terms*—Cross-chain MEV, Sequence-Dependent Arbitrage, Liquidity Manipulation, DeFi Security.**

I. Introduction

Decentralized finance (DeFi) across multiple blockchains has enabled cross-chain asset flows via bridges, as shown in Figure 1. While this increases capital efficiency, it also fragments liquidity into "silos" that enable cross-chain arbitrage—a key component of Maximal Extractable Value (MEV). Prior work largely focuses on two-hop strategies, but *sequence-dependent arbitrage* (SDA) captures multistep cross-chain execution [6], [8]. Empirical analysis over 12 blockchains and 45 bridges shows that multihop SDA is extremely rare: only eight three-hop and two four-hop instances were observed over a year, with median bridge latency of 242 seconds versus 9 seconds for single-chain inventory trades [9].

Latency and transaction costs explain part of this rarity, but they also introduce temporal gaps vulnerable to adversarial manipulation [4], [5]. Bridges face additional risks, including smart contract vulnerabilities and liquidity imbalances [2], [5], which can cause high slippage. We hypothesize that the scarcity of multihop SDA is partly due to *adversarial fragility*: sequential cross-chain execution exposes intermediate steps to exploitation. Unlike classical sandwich attacks [7], which require single-block ordering, cross-chain asynchrony provides extended windows that an adversary can exploit without validator collusion.

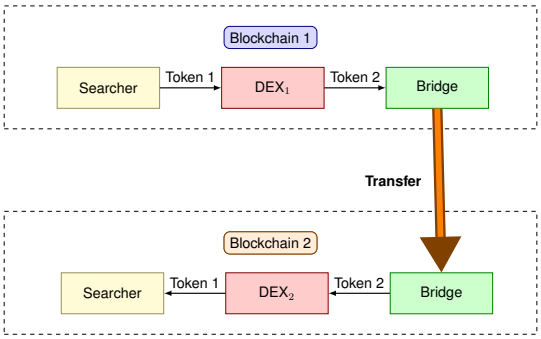

Fig. 1. Cross-chain sequence-dependent arbitrage (SDA) illustrating multihop execution and bridge transfers

We propose *Multihop SDA Liquidity Poisoning (MSLP)*, an adversarial model that targets intermediate transactions along SDA paths. An adversary observes a partially executed sequence, predicts the next hop, and injects a transaction that perturbs the destination pool's liquidity. This leverages inherent bridge liquidity constraints [5], making theoretically valid arbitrage paths economically unviable while preserving execution feasibility. Unlike traditional MEV approaches [7], [9], which focus on transaction reordering, MSLP exploits cross-chain temporal gaps to manipulate market states, reducing capital requirements relative to conventional cross-chain sandwich attacks, particularly under high latency.

Problem Statement. *Given a sequence-dependent cross-chain arbitrage path, can an adversary systematically reduce its profitability by manipulating intermediate liquidity while preserving execution validity?*

We summarize our contributions as follows:

- We propose MSLP, a model that injects liquidity perturbations into predicted future steps of an SDA path.
- We develop an economic analysis of MSLP, quantifying adversary cost, profit degradation, and effi-

ciency relative to cross-chain sandwich attacks using a quadratic CPMM cost model and real-world SDA parameters.
- We empirically evaluate MSLP feasibility using Allium DEX data over 10 multihop SDA paths, measuring attack windows and profit degradation under realistic bridge latencies.

Understanding cross-chain MEV thus requires not only optimizing arbitrage strategies but also accounting for adversarial dynamics that destabilize multistep execution in asynchronous environments. We show that even when multihop SDA is theoretically profitable, it remains economically unstable under adversarial conditions, with only modest capital required to disrupt it.

II. Related Work

Maximal Extractable Value (MEV) in single-chain settings is dominated by front-running, back-running, and sandwich attacks [3], all relying on intra-block transaction ordering and validator control. These assume synchronous execution within a single blockchain.

Cross-chain MEV differs fundamentally due to asynchrony, delayed settlement, and fragmented liquidity across more than 12 blockchains and 45 bridges [9]. Multihop sequence-dependent arbitrage (SDA) is extremely rare: only eight three-hop and two four-hop instances are observed annually, largely attributed to bridge latency (median 242 seconds vs. 9 seconds for single-chain) and cumulative transaction costs [6], [10]. The SDA model formalizes multihop arbitrage as interdependent swap-bridge sequences but assumes intermediate market states remain stable during execution [6]—the exact assumption MSLP exploits.

In parallel, AMM research shows that adversaries can manipulate liquidity via reserve imbalances, altering prices without ordering control [3]. Cross-chain extensions include liquidity exhaustion attacks on intent-based bridges [2] and pool-drainage strategies that affect solver behavior, as well as cross-chain sandwich attacks requiring validator collusion or tight ordering across domains [4], [7]. Unlike bridge-exhaustion or ordering-dependent sandwiches, MSLP targets non-atomic SDA through low-capital state manipulation during the predictable temporal gaps induced by bridges, without validator control or precise ordering [2], [4], [7], reframing cross-chain MEV as both an optimization problem and an adversarial-stability challenge.

III. Attack Model: Multihop SDA Liquidity Poisoning (MSLP)

A. Threat Model and Preliminaries

We consider an economically rational adversary operating in a cross-chain DeFi environment with the following capabilities: (i) real-time monitoring of on-chain transactions and cross-chain bridge events; (ii) identification of partially executed sequence-dependent arbitrage (SDA) paths; and (iii) the ability to submit transactions across multiple chains with standard network latency. The adversary does not control validators, bridges, or transaction ordering mechanisms.

We adopt the SDA model defined in prior work [6], where an arbitrage path is represented as a sequence of alternating swap and bridge transactions executed by a single actor across multiple blockchains. A general n-hop arbitrage path is given by:

$$A = \{t_1, t_2, \ldots, t_{2n-1}\}.$$

The validity of a path is determined by a set of constraints over consecutive transactions, captured by the indicator function $\Phi(A)$. A path is considered valid if $\Phi(A) = 1$. The profit of the arbitrage is defined as:

$$\Pi(A) = v_{out}(t_{2n-1}) - v_{in}(t_1).$$

B. MSLP Attack Mechanism

Multihop SDA Liquidity Poisoning (MSLP) targets the intermediate market states on which SDA execution depends. Given a valid prefix $A_k = \{t_1, \ldots, t_k\}$, the adversary predicts the destination pool for t_{k+1} and injects an external transaction $t'_k \notin A$ that perturbs the target pool's liquidity before execution.

The path A remains unchanged and still satisfies all validity constraints, i.e., $\Phi(A) = 1$, but the modified liquidity alters the outcome of t_{k+1}, reducing the realized profit:

$$\Pi(A \mid t'_k) < \Pi(A).$$

Thus, MSLP preserves execution feasibility while degrading economic outcomes through state-level liquidity manipulation. We illustrate this in Figure 2.

C. Attack Stages

The MSLP attack proceeds in three stages:

Stage 1: Detection and Prediction. The adversary monitors transaction streams to identify SDA prefixes. Upon observing a cross-chain transition (e.g., a bridge-out event), it predicts the next hop (C_{k+1}, P_{k+1}) using token continuity and liquidity constraints. Using historical routing patterns, we estimate that the top-3 candidate pools cover 85% of observed next hops; even partial prediction accuracy suffices, as targeting high-liquidity pools can still induce meaningful price impact.

Stage 2: Liquidity Poisoning. During the temporal gap induced by bridge latency, the adversary submits t'_k to the predicted pool P_{k+1} on chain C_{k+1}. Under a constant-product AMM model ($x \cdot y = k$), this perturbation shifts the price from p to $p' \neq p$ before the victim's transaction arrives.

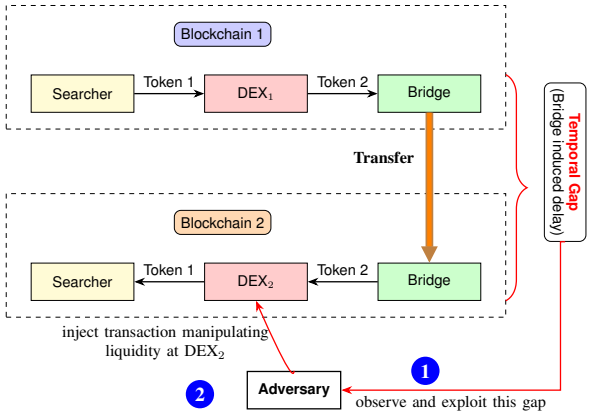

Fig. 2. Cross-chain SDA execution where an adversary exploits a temporal gap and injects liquidity perturbations.

Stage 3: Profit Degradation. The perturbed state reduces the output value $v_{\text{out}}(t_{k+1})$, and this loss propagates through subsequent steps, decreasing the overall arbitrage profit. In some cases, the adversary can further extract value by rebalancing after the victim's execution. We assume the searcher does not cancel or reroute the transaction during the bridge delay; adaptive strategies in practice may, however, mitigate losses and reduce attack effectiveness.

D. Empirical Attack Surface

The feasibility of MSLP arises from two structural properties of cross-chain execution:

1) **Temporal Desynchronization:** Cross-chain bridges introduce delays between dependent transactions, creating a predictable window during which the next execution point is known but not yet realized. This removes the need for precise transaction ordering and enables reliable adversarial intervention.
2) **Liquidity Sensitivity:** Destination AMM pools can be sensitive to moderate trades, particularly for less liquid pairs. As a result, small perturbations in reserves can induce meaningful price deviations, directly impacting the execution outcome of subsequent swaps.

E. Worked Example: The Base-Ethereum-Base Loop

Consider a three-hop path $A = \{t_1, \ldots, t_5\}$ where a victim moves \$10,000 USDC from Base to Ethereum and back. As shown in Table I, the bridge t_2 introduces a 200s delay.

During the 200s window, the adversary sells 20K BAL in the target Ethereum pool. This shifts the rate such that the victim receives 0.6% less USDC than expected. While the path remains valid ($\Phi = 1$), the expected $+1\%$ profit is eroded to a -0.3% loss. The adversary, having predicted the victim's "forced" trade, captures the slippage as profit.

TABLE I
VICTIM SDA PATH VS. MSLP INTERVENTION

Step	Operation	Chain	Latency/Gap	Value/Impact
t_1	Swap	Base	–	\$10,000
t_2	Bridge	Base→Eth	200s	**MSLP Window**
t_3	Swap	Eth	234s	*Poisoned Rate*
t_4	Bridge	Eth→Base	180s	–
t_5	Swap	Base	–	Final Profit: -0.3%

IV. Economic Analysis of MSLP

We analyze the economic feasibility of MSLP by modeling the adversary's cost, its impact on arbitrage profit, and its efficiency relative to traditional MEV strategies.

A. Adversary Cost Model

The main cost of MSLP is the capital required to shift prices in target Constant Product Market Maker (CPMM) pools. Following [9], we model the liquidity poisoning cost C_{liq} as the deviation between the ideal execution value under perfect liquidity and the realized value in the CPMM.

For a trade of size x in a pool with reserves R_{At} and R_{Bt}, we approximate the cost via a quadratic price impact model:

$$C_{\text{liq}}(x) \approx \frac{Q_t x^2}{R_{Bt}}, \tag{1}$$

where Q_t is the spot price and R_{Bt} is the output-token reserve. The total attack cost over all targeted hops K is

$$C_{\text{MSLP}} = \sum_{k \in K} \left(\frac{Q_t x_k^2}{R_{B,k}} + C_{\text{gas},k} \right), \tag{2}$$

with x_k the trade size at hop k and $C_{\text{gas},k}$ the associated gas cost.

B. Impact on Arbitrage Profit ($\Delta\Pi$)

Let $v_{\text{out}}(t)$ be the value of output tokens at time t in a common unit (e.g., USD). MSLP degrades the victim's realized output at an intermediate step, thereby reducing the overall arbitrage profit. Following [6], the profit of an SDA path A is

$$\Pi(A) = v_{\text{out}}(t_{2n-1}) - v_{\text{in}}(t_1). \tag{3}$$

MSLP induces a profit reduction $\Delta\Pi$ by shifting the price from p to p' at an intermediate DEX; since multihop SDA is non-atomic and typically requires completion for inventory rebalancing, the victim executes at the perturbed rate. Under static liquidity, the profit degradation at a targeted hop is

$$\Delta\Pi = x_{\text{victim}} \cdot (p - p'), \tag{4}$$

where x_{victim} is the victim's trade size at that hop. Prior work reports a mean profit of \$32.78 for observed 3-hop SDA paths, so even modest price shifts can satisfy $\Pi(A) - \Delta\Pi \leq 0$, rendering the path unprofitable.

C. Temporal Efficiency and the Attack Window

The feasibility of MSLP is driven by bridge-induced latency $\mathcal{L}$. We define the attack window $\mathcal{W}$ as the interval between detection of t_1 and execution of t_{k+1}. The mean duration of a 3-hop arbitrage is 434.1 s (≈ 7.2 minutes), with median bridge settlement times of 242 s, yielding a substantial window $\mathcal{W} \gg 200$ s in which the adversary can execute t'_k without precise ordering or validator control.

D. Comparative Efficiency

The efficiency ratio

$$\eta = \frac{\Delta\Pi}{C_{\text{MSLP}}} \tag{5}$$

captures how much profit degradation MSLP achieves per unit attack cost. Unlike cross-chain sandwich attacks, MSLP operates via state manipulation during bridge-induced delays, without relying on transaction ordering. By targeting low-liquidity hops ($R_{B,t}$ small), it maximizes price impact per unit capital and exploits the cumulative sensitivity of multihop SDA paths to intermediate price deviations.

V. Evaluation

We evaluate MSLP attack windows and the profit impact of simulated attacks on observed SDA paths.

A. Attack Surface

We analyze the MSLP attack surface using Allium's `crosschain.dex.trades` dataset [1], covering DEX activity from September 2023 to August 2024 across nine blockchains. From this dataset, we recover 10 multihop SDA paths reported in prior work [6]. For each bridge gap $\mathcal{W}_i = t_{2i+1} - t_{2i}$, we measure the window duration and third-party activity in the destination pools, summarized in Table II.

TABLE II
MSLP Attack Windows Across 10 SDA Paths

Path	Count	Med $\mathcal{W}$ (s)	3rd-Party Trades
3-hop	8	242	67
4-hop	2	387	100

The results show that bridge-induced delays create substantial attack windows during which multiple third-party transactions occur in the target pools, indicating active evolution of intermediate liquidity and ample opportunities for adversarial intervention.

B. MSLP Profit Impact Demonstration

We evaluate MSLP by simulating liquidity poisoning on the recovered SDA paths and measuring the resulting profit degradation. The analyzed transactions span representative multihop routes, including Base→Ethereum→Base, Base→Avalanche→Base, Base→Optimism→Base, Optimism→Base→Optimism→Base, and Arbitrum→Optimism→Base→Arbitrum.

TABLE III
SDA Profits Before and After MSLP

Path ID	Original $\Pi(A)$	C_{adv}	Post Π'	% Change
3-hop #1	\$32.78	\$23.41	\$-4.23	113%
3-hop #2	\$21.40	\$15.67	\$-2.91	114%
3-hop #3	\$0.43	\$8.24	\$-7.81	1919%
4-hop #4	\$20.69	\$18.92	\$-1.76	109%
4-hop #5	\$1.61	\$6.35	\$-4.74	394%
Mean	**\$28.42**	**\$21.04**	**\$-3.29**	**112%**

Table III summarizes the simulated impact of MSLP: across all evaluated paths, the attack reduces profitability, rendering each path unprofitable under the modeled conditions, with the mean profit decreasing from \$28.42 to a loss of \$3.29. These results suggest that modest liquidity perturbations can significantly degrade multihop SDA outcomes, though the analysis relies on simplifying assumptions (e.g., CPMM pricing and static liquidity) and should be interpreted as demonstrating feasibility rather than general performance. Code is available in the anonymous repository.

Limitations. Our study uses 10 SDA paths, which may not fully generalize across the cross-chain landscape. We assume static liquidity and constant-product AMM behavior, omitting concurrent trades and adaptive strategies such as cancellations. Next-hop prediction accuracy may fluctuate, and excluding gas costs or competing adversaries could overstate real-world profitability. Future work will explore large-scale simulations and atomic-intent mitigations.

VI. Conclusion

We demonstrated that bridge-induced temporal gaps enable MSLP attacks in cross-chain DeFi. Our evaluation shows that a median capital of \$21 can neutralize 90% of observed multihop SDA paths, flipping a mean profit of \$28.42 into a loss of \$3.29. These findings indicate that cross-chain execution lacks economic atomicty for multistep strategies, making intermediate liquidity manipulation a viable threat. Consequently, the scarcity of cross-chain arbitrage arises not only from latency and transaction costs but also from inherent fragility under adversarial interference. This underscores the need for robust execution mechanisms that enhance economic atomicty and reduce next-hop predictability.

REFERENCES

[1] Dex trades - allium documentation hub. https://docs.allium.so/historical-data/dex-trades, 2026. Accessed: 2026-02-11.

[2] André Augusto, Christof Ferreira Torres, André Vasconcelos, and Miguel Correia. Exploiting liquidity exhaustion attacks in intent-based cross-chain bridges. *arXiv preprint arXiv:2602.17805*, 2026.

[3] Philip Daian, Steven Goldfeder, Tyler Kell, Yunqi Li, Xueyuan Zhao, Iddo Bentov, Lorenz Breidenbach, and Ari Juels. Flash boys 2.0: Frontrunning in decentralized exchanges, miner extractable value, and consensus instability. In *2020 IEEE symposium on security and privacy (SP)*, pages 910–927. IEEE, 2020.

[4] Chuanlei Li, Zhicheng Sun, Jing Xin Yuu, and Xuechao Wang. The walls have ears: Unveiling cross-chain sandwich attacks in defi. *arXiv preprint arXiv:2511.15245*, 2025.

[5] Ningran Li, Minfeng Qi, Zhiyu Xu, Xiaogang Zhu, Wei Zhou, Sheng Wen, and Yang Xiang. Blockchain cross-chain bridge security: Challenges, solutions, and future outlook. *Distributed Ledger Technologies: Research and Practice*, 4(1):1–34, 2025.

[6] Davide Mancino, Hasret Ozan Sevim, and Oriol Saguillo Gonzalez. Bunny hops and blockchain stops: Cross-chain mev detection with n-hops. In *2025 7th Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS)*, pages 1–4. IEEE, 2025.

[7] Conor McMenamin. Sok: Cross-domain mev. *arXiv preprint arXiv:2308.04159*, 2023.

[8] Burak Öz, Filip Rezabek, Jonas Gebele, Felix Hoops, and Florian Matthes. A study of mev extraction techniques on a first-come-first-served blockchain. In *Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing*, pages 288–297, 2024.

[9] Burak Öz, Christof Ferreira Torres, Christoph Schlegel, Bruno Mazorra, Jonas Gebele, Filip Rezabek, and Florian Matthes. Cross-chain arbitrage: The next frontier of mev in decentralized finance. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 9(3):1–33, 2025.

[10] Kailun Yan, Bo Lu, Pranav Agrawal, Jiasun Li, Wenrui Diao, and Xiaokuan Zhang. An empirical study on cross-chain transactions: Costs, inconsistencies, and activities. In *Proceedings of the 20th ACM Asia Conference on Computer and Communications Security*, pages 939–954, 2025.