At EthGlobal2024, Konrad, Orest, and I worked together on designing and coding an LVR mitigation solution using UNIV4 hooks and SUAVE. In this summary, I'll share our work and insights.

Concept

LVR

Unlike market makers, liquidity providers (LPs) solely provide liquidity to AMMs and aren't concerned with price discovery. This task is outsourced to cross-domain arbitrageurs, who profit from the price difference between markets in different domains until their prices reach equilibrium.

Price discovery occurs more frequently in faster domains, leaving LPs in the slower domains to essentially offer their assets at a stale price. The value extracted from these stale quotes is known as Loss Versus Rebalancing (LVR).

On Ethereum, the Proposer-Builder separation (PBS) block auction leaves most of the LVR in the pockets of the proposers. To challenge the status quo, we aimed to capture LVR at the protocol level instead and redistribute it to LPs, incentivizing deeper on-chain liquidity.

While there are many proposed directions to capture LVR, we decided to auction off the right to execute the first swap in the block for the given pool.

Why Vickrey auction?

The value of an LVR opportunity is uniquely tied to the pool, particular block, and external market conditions. No one is in a better position to evaluate its value than the trading firms capturing it.

An auction provides a way to extract this information from them, essentially allowing a more knowledgeable party to do value discovery for you.

For our use case, we deemed the Vickrey auction most appropriate (given enough competition) as it incentivizes participants to bid honestly and not shade their bids. This simplifies the auction process for bidders and can potentially result in more value captured.

Why SUAVE?

Although conducting an auction on-chain is theoretically possible, it is unfeasible for our use case. We needed a system that was cheaper and faster and offered privacy. The latter is crucial for implementing Vickrey's auction, which is a sealed bid.

A centralized off-chain auction solves this, but it lacks credibility and robustness, requiring trust in the operator. The operator's role would be left only to those backed by a reputable firm or with a great reputation themselves—that has its costs.

SUAVE provides an alternative solution with confidentiality, cheap execution, and fast coordination while remaining credible and requiring trust in the TEE rather than a centralized operator.

UNIV4

UniV4 lets you leverage hooks to create your own rules for the pool without having to go through the hassle of designing your own DEX from scratch. This suited our usecase as our intention was not to redesign the DEX completely but to focus on a particular problem and solve it on top of already established DEX design.

Design

Flow

1. Arbitrageurs(ARBERs) lock assets used for bidding in a UNIV4-controlled contract.
2. ARBERs identify cross-domain opportunities.
3. ARBERs participate in the Vickrey auction on SUAVE.
4. The SUAVE auction is resolved with the winning bid and signature announced publicly.
5. The winning ARBER uses this signature to unlock the pool and execute their strategy.
6. Proceeds are later distributed to affected LPs.

[

image

1654×834 109 KB

](https://collective.flashbots.net/uploads/default/original/2X/2/27e996178dd60f2d95624eeeb2218f67d73c8635.png)

Mechanism

Vickrey auction on SUAVE

- The auction contract hosts a hidden signer.
- Auctions are specific to block and pool.
- Anybody can initiate an auction for any future block
- For every bid, funds locked in the Registry must be sufficient.
- The auction is supposed to be resolved with enough time for the bidder to include the trade in the block and for the block to travel to the proposer.
- The winning bid with a signature is publicly submitted to the SUAVE chain.
- If no bidder participates in the auction, SUAVE contracts emit a general signature allowing anyone to unlock the pool.

UNIV4

- Allows the first swap in a block if a valid signature is provided.
- Subsequent swaps don't require a signature.

Registry

- Validates the auction's winning signature.
- Deducts the winning bid from the winner's balance.
- Redistributes the proceeds to LPs.

[

image

2096×1202 250 KB

](https://collective.flashbots.net/uploads/default/original/2X/5/55808669257a2ea381bbcf37a454f7c81c759104.png)

Implementation

We've developed a POC that includes the Registry, UNIV4 hook, and SUAVE auction contract. Challenges with UNIV4 deployment on L1 led us to use L2 (Arbitrum Sepolia), temporarily bypassing block restrictions and timed auction resolution.

Nevertheless, an auction resolution mechanism using beacon node API is already implemented and can be enabled once UNIV4 hook deployment on Holsky/Sepolia L1 becomes viable. With this method, the auction for block N can only be resolved X seconds before the expected timestamp of block N.

Check out our repo

Further work

Future steps include deploying on an L1 testnet and addressing technical challenges, such as preventing bid spamming, distributing proceeds fairly, and preventing auction abuse.

Instead of L1 deployment, we also considered an L1 block emulator that optimistically maps slot and block number to the timestamp inside L2 runtime.

Auction resolution

One of the critical aspects of the described design is the timing of auction resolution.

Latency considerations are essential for executing cross-chain opportunities. If the auction ends too soon, less value is captured by the system; but if the auction ends too late, the winning bidder risks missing the block.

[

image

2236×1422 282 KB

](https://collective.flashbots.net/uploads/default/original/2X/7/74f790d6bfaa273258dc401eff5b84a1082bcb4e.jpeg)

Eden’s bids dashboard - block 19420415

Furthermore, even when there is no LVR opportunity (no auction participant), aggregators and solvers still need to provide a valid signature to unlock the pool for their retail trades. Obtaining the signature is again tied to the auction resolution. This can be problematic, as these entities need to start accounting for the chance the retail trades might not make it to the next block. Could this and related complexities associated with the system dissuade them from using it?