title

: A2MM: Mitigating Frontrunning, Transaction Reordering and Consensus Instability in Decentralized Exchanges

Date

: July 23, 2021

topics

: market structure, MEV, trade optimizations

A2MM - Critique and Defects

This article examines the claims and findings of a research paper titled, "A2MM: Mitigating Frontrunning, Transaction Reordering and Consensus Instability in Decentralized Exchanges*.

In this paper, the authors claim to have implemented a MEV defeating AMM. My thoughts on their claims of A2MM as an alleged solution are as follows.

When A2MM receives a swap transaction for a market with the assets X and Y, A2MM atomically performs optimal routing and arbitrage among the considered AMM, minimizing subsequent arbitrage transactions.

Validating their claims

The paper actually comes to the conclusion that batch processing

reduces MVI the most as it brings the spreads between markets closer. This principle is how A2MM works:

It asses the size of the trade, if it's enough to balance the price among the underlying liquidity market, then it executes the trade.

Axioms

Theorem 1

: Routing optimization aims to level the asset price on multiple AMMs and can be solved by greedily routing transaction volume.

Theorem 2

: An optimal strategy (Soptimal) performing routing and arbitrage among N AMMs on market (X , Y) is equivalent to a batch execution strategy (Sbatch). Sbatch consists of at most N swaps (SwapXtoY or SwapY toX). Both Soptimal and Sbatch change the states of AMMs from

TLDR: close the spread via flash loans.

'Faulty' Assumptions

specific objections to their methodology

Assumes only one

DEX on chain

In this work, we take the stance that a blockchain should ideally only operate one AMM smart contract, to increase the financial efficiency, reduce network layer and block-space overhead, and consequently increase blockchain throughput as well as security.

Assume's that all AMM swaps initiate at the A2MM's user interface (frontend)

They assume that all trading happens on their frontend interface

Off chain optimization and routing

[

1090×454 30.3 KB

](https://global.discourse-cdn.com/standard10/uploads/manifold/original/1X/ff22db1b8f938a5393a937bf3a77435cd285c097.png)

sic

. A2MM contract with a user interface while using Uni- and Sushiswap's liquidity pools. Upon receiving a swap request, A2MM derives the routing and arbitrage parameters on the fly on-chain.

A2MM only mitigates two-point arbitrage overhead

41% of the on-chain arbitrages are two-point arbitrages. Therefore, we estimate that A2MM will decrease about 41%

This only holds true if you assume you are the only exchange on chain, which they do assume in their paper.

Solves Sandwich attacks by aggregating volume across venues

By routing the trading volume onto multiple AMM exchanges, A2MM aggregates the MVI (minimal profitable victim input) thresholds among the underlying liquidity pools. In the simple case, where two AMM markets have the same liquidity and pricing formula, A2MM's accumulative MVI threshold is 2× the MVI of a single AMM.

Miner Network performance

Uncle rates have gone down. They dispute this fact, which makes no sense as we have seen post-Berlin block times coming down and miner profits going up (as a result of faster block times). Uncle rate has gone down, read this article by f2pool: The Secret Weapon F2Pool Used to Tackle Its Uncle Rate | by bloXroute Team | bloXroute Labs | Medium

Additional thoughts

The paper neglects the fact that trade execution times (re: settlement) will occur slower. Also does not match against RFQ systems (e.g. 0x, AirSwap, etc), so is not assessing true market price on chain.

Disclaimer