Offchain Labs is developing a proof-of-work mechanism to disincentivize massive sybil attacks by MEV searchers on the Arbitrum sequencer. In this post, I describe issues with the proposed mechanism and suggest alternative mechanisms that more efficiently achieve the goals of the proposal.

Introduction to PR 1504

Skip this section if you are already familiar with the proposal.

Yesterday, February 23, Arbitrum core devs publishedPull Request 1504, titled “Add a relay client connection nonce.”

In the Flashbots discord, PlasmaPower stated that the intention of the mechanism is “to stop incentivizing tons of connections to our feed.” Ben Burgess stated that the relay currently serves between 100,000-150,000 [1] connections at a time.

The proposal implements a change such that, when a new block is available, the relay will first deliver it to the 25 feed subscribers with the lowest “nonce.” [2] After an artificial 50 millisecond delay, the relay will deliver the block to all other subscribers. The idea is that, to minimize latency, searchers will no longer have an incentive to sybil attack the relay. Instead, they will spend tens of thousand of dollars, daily, on GPUs and electricity for mining low nonces. By my rough estimate, MEV searchers on Arbitrum currently profit roughly $5-20 million annually.

Context

Currently, MEV searchers open thousands of concurrent subscriptions to the Arbitrum Sequencer Relay’s WebSocket feed. They may even do so from numerous IPs to evade rate limiting. The current system incentivizes searchers to do so because messages are propagated to subscribers sequentially and, importantly, in random order.

Solution Guidelines

While avoiding turning this into a formal mechanism design problem, we can surmise some general guidelines:

- The mechanism should reasonably minimize the load on the sequencer.
- The mechanism should be easy to implement, while a more sophisticated longer-term auction design is considered. See Footnote [3] below.
- Imposing an artificial delay on a subset of connections (i.e. non-searchers) is acceptable.
- We do not want to affect the FIFO nature of the sequencer, as this opens a can of worms.

I also assume (and hope) that, if the alternative is spending millions of dollars annually on GPUs and electricity, all stakeholders would prefer to direct these funds to a charity or the Arbitrum ecosystem.

For environment and reputational reasons, I assume that we prefer to avoid PoW mechanisms, if possible.

Issues

There are several critical issues with the proposed solution and obviously better alternatives.

The primary issue is how economically wasteful

this mechanism is.

MEV searchers on Arbitrum currently profit roughly $5-20 million annually. The current proposal would have a large portion of these funds spent on mining proof-of-work hashes.

- A simple auction design could instead direct these funds to a charity or the Arbitrum ecosystem.
- Less importantly, MEV searchers’ time is now spent on optimizing PoW algorithms instead of… other things? Such as increasing profits to donate even more to charity or the Arbitrum ecosystem.

As mentioned above, PR 1504 is also quite harmful to the environment. Pathos aside, it is (subjectively) not in the interest of Arbitrum’s ecosystem to associate themselves with promoting PoW.

Alternative Mechanism Designs

Suggestions for alternative auction designs that are more economically efficient and satisfy the previously discussed constraints, especially simplicity of implementation and FIFO ordering.

1. Pay X

ETH to some address be in the priority queue for 24 hours. Very simple to implement. There's still an incentive to open multiple connections, but it's not cheap to do so. X

can be adjusted daily. Subscriptions to the feed can optionally contain a header with an ECDSA signature from an EOA that has paid in the past 24 hours.

1. Searchers submit blind bids before Sunday 12:00 UTC to an email address, web form, or smart contract. Top 5 bidders pay the 6th's bid. Give the winning bidders API keys or have them authenticate via ECDSA signatures. This is extremely simple to implement. Credit to @snoopy_mev

on twitter.

Footnotes

[1] Simultaneous broadcast is a well-studied problem. Every financial exchange and video game has solved this problem before.

[2] Keccak hash of the current date + a salt phrase.

[3] As PlasmaPower stated, PR 1504 is intended to be a temporary solution until Offchain Labs designs a more sophisticated transaction ordering solution, as discussed here.