Overview

Some domains/chains, like Arbitrum, have expressed a preference for continuous-time sequencing. This is easy to implement with something like a shared sequencers. There is a large design space for interfaces with searcher/builder markets in such settings, which has remained largely unexplored. Any exploration of this design space would be valuable

Guiding Examples and Questions.

- The most extreme case is first-come-first-serve continuous-time ordering implemented with something likeThemis
- Arbitrum's Timeboost design represents a hybrid design between auctions and time-based ordering. Additional research on the design would be valuable, especially in understanding what kinds of games searchers will play to access opportunities.
- Another question is if more complex, but efficient designs are tenable (e.g. dynamic pricing curves, using more than simply a function of bid and time to order)
- Another question is if more complex, but efficient designs are tenable (e.g. dynamic pricing curves, using more than simply a function of bid and time to order)
- A continuous-time sequencer can still accept objects which resemble bundles that are the combination of multiple users' interaction (e.g. limit orders being cleared by a market maker). What are important differences between such settings where users can choose to submit to a coordination device that waits and aggregates user information or submit directly to a sequencer and block-based settings? Coming up with a good concrete example is part of the exercise

Relevant links

- Themis: Fast, Strong Order-Fairness in Byzantine Consensus
- [2306.02179] Buying Time: Latency Racing vs. Bidding in Fair Transaction Ordering
- Transaction ordering policy - Arbitrum Research
- Tarun's talk on fairness