

Suppose you want to choose a transaction ordering policy for your new roll-up sequencer, L1 blockchain, DEX aggregator, or new financial exchange. What should you do?

Well, I don't know. But I have written a [new article](#), that should be helpful. It proposes a game-theoretic framework to study the relative performance of different transaction ordering policies.

It takes the perspective of a contest for earlier transaction execution by two competing players. For example, the two players could compete on an arb opportunity arising through a price discrepancy between an off-chain CEX and an on-chain DEX. How this competition plays out, depends on the transaction ordering policy in place: do we use FCFS, batch auctions, Arbitrum's [time boost](#), something else?

I want to make sense of several aspects of transaction ordering policies:

- Can we avoid latency competition by using bids to determine the ordering and inclusion of transactions? (not really)
- How does latency competition look like in a batch auction world? (not good, we have zero average profit for bidders in equilibrium when accounting for the cost of latency investment)
- What are the performance differences between batch auctions and hybrid formats such as the time boost proposal for Arbitrum? (depends on the parameterization, but in general they look very similar)

I think that the results are particularly relevant for roll-up sequencing but should also inform the broader transaction ordering discourse for L1s or off-chain aggregators.

I would be happy for any feedback, comments, suggestions, questions, opinions etc.