

Thanks to Akaki Mamageishvili and participants at the “Make MEV, not (latency) wars” event for helpful feedback. A related presentation of the main idea and more “pictorial” view is available [here](#).

There is a justified attention on designing the market for blockspace in a way that it does not replicate traditional financial markets distortions. In particular, the now well-known latency arms race in which hundred millions of dollars have already been spent on network infrastructure to gain smaller and smaller advantages.

Will the blockspace market suffer from a similar trap? One where lots of wealth accrues to infrastructure & operations that have low/zero real utility, and that concentrates the market around a few big players that have capital & technology to compete in the arms race?

The answer is, it depends. On the market design and in particular on transaction ordering mechanisms. We provide two example designs, one time-based like traditional finance and the other bid-based like Ethereum L1. We show that the resulting incentives to invest in latency advantage are very different: constant in the former case and decreasing/finite in the latter. Then we present some open questions and we discuss the design space and fundamental trade-offs of hybrid designs.

FCFS designs for transaction ordering (vanilla FCFS)

The risk of falling into the trap is high.

This design makes block ordering similar to ordering in tradfi exchanges where there is continuous-time trading and FCFS ordering of transactions.

Consider a generic 12s interval and suppose that there are 12 mev opportunities in the interval that arrive at random times. The searcher that has the fastest network will win every single opportunity. The slightest latency advantage gives the searcher uniform monopoly power over the entire interval.

Auction designs for transaction ordering

The risk of falling into the trap is low.

In this design the transactions are ordered by the willingness-to-pay of the searcher, submitted in the form of a bid. Suppose there is a fast searcher with a latency of f milliseconds vs a slow searcher with a latency of $g > f$. It takes the fast searcher f ms to detect a new mev transaction and f ms to relay it to the block proposer. This means that the fast searcher will be able to extract up to $2f$ ms before the end of the slot and the slow searcher up to $2g$ ms. Call the advantage x ms.

Consider a 12s batch/slot interval. In this design the fast searcher has monopoly power over roughly $x/12000$. As an example, if x is 10ms this gives monopoly power to unilaterally extract only 0.08% of the time.

The same advantage gives the fast searcher unilateral extraction power 100% of the time in the other design, so the incentive to invest in fast network infra is reduced by a factor of >1000 in the 10ms advantage example.

Assuming a convex cost to acquire more advantage we can easily see that, since the gain grows linearly, it quickly becomes uneconomical to invest more. After initial inefficiency is competed away the incentive to invest more in latency improvements stops.

Screen Shot 2023-03-03 at 8.42.22 AM 1878×430 65.6 KB

This is in contrast with the FCFS design in which any small advantage grants uniform dominance. Under the same reasonable cost structure we can easily see that the incentives to gain a latency advantage never goes away.

Screen Shot 2023-03-03 at 8.43.03 AM 1878×412 61.5 KB

What about incentives to invest in latency for other participants?

- proposers: NA, they are already monopolists
- relays: since connecting is free we can assume that builders will connect to the faster relay, however since relays don't have revenue the economic incentive to invest in network infra seems low. For a malicious party that wants to subsidize this to monopolize the market the same argument as in the main section applies
- builders: low, again same incentives as the main argument, linear returns and convex costs, it quickly becomes more profitable to invest in other dimension of advantage (eg, acquiring private orders, devise more complex search strategies that extract closer to theoretical mev)

What about geographical distribution of mev opportunity origin-set (user wallets, CEXs) and destination-set (validators)?

There are arguments on whether geographical distribution of users and validators is a centralizing or decentralizing force for other participants in the supply chain. We only note that users and validators are themselves participants in the location

game. A searcher can invest to operate vertically integrated validators or subsidize them to relocate, same for user wallet providers (CEXs are harder to move).

The higher the level of geographical distribution of the network the higher the cost to acquire an average advantage of x milliseconds for a searcher. One can argue that now the searcher can first invest in centralizing the network (e.g., moving wallet providers and/or validators) and then invest in latency advantage at a lower unit cost. But other searchers can follow at a much lower cost by adjusting their location to the new geographical distribution.

This game is more complex than the previous one, so we need to be careful with our simplifications and the full analysis should consider key factors like (1) geographical distribution of MEV sources that are hard to move (i.e., CEXs) and their relative weight (2) intrinsic incentive to relocate closer to sources for other participants (3) opportunity cost of investing in latency and (4) the dynamics of repeated moves. Intuitively, it again seems that since gains from latency advantage are linear, for any reasonable convex cost structure, the incentive to invest in latency are significantly smaller in this design vs designs with continuous-time trading. However, a more thorough analysis of the relocation game is needed to say whether the resulting equilibrium network configuration is sufficiently decentralized.

Open questions

How does a geographically distributed demand/supply network change incentives?

How centralized is the equilibrium location of service providers (wallets/.../validators)? What is a reasonable measure of decentralization in this case?

In practice, in blockchain-based discrete-time auction designs, at what latency level (ms/.../ns) becomes uneconomical to invest more?

Hybrid designs

We have seen how two extremely different designs for transaction ordering lead to very different incentives for participants. The designs we have analyzed actually differ on two important dimensions:

- processing policy: continuous-time (streaming) vs discrete-time (batch)
- ordering policy: time-ordering vs bid-ordering

The FCFS is on one corner of spectrum, continuous-time processing and time-ordering, and the auction design is in the opposite corner.

Choosing a discrete-time batch processing policy opens up a wide ordering policy design space, virtually any combination of time-ordering and bid-ordering can be implemented. But the system has higher latency of settlement, equal to the length of the batch interval.

In the other case of continuous-time streaming, time-ordering is most natural and has no additional latency. Mechanisms that also allow to bid for inclusion can be introduced at the cost of some latency (such as the time-boost proposal that arbitrum is working on). However, the design space is more constrained.

In both cases above, a design with more latency in settlement introduces delay to user transactions and state updates which is welfare decreasing, but allows for more sophisticated bid-ordering mechanisms that can improve security, robustness, and mitigate the losses from MEV extraction which are welfare increasing. These are fundamental trade-offs to keep in mind while exploring this wide design space.