Cost of Feudalism
Towards a Theory of Maximal Extractable Value (MEV)

Tarun Chitra

Joint work w/ Theo Diamandis (MIT), Kshitij Kulkarni (Berkeley)
MEV: “You know it when you see it”

Maximal extractable value comes in many shapes and sizes

- Sandwich attacks 🥪
- Liquidations 🌊
- Arbitrage 🧛
- NFT mint front-running 🎨
- Cross-chain 🕰️

Source: Eigenphi.io
What is MEV?

- Maximal Extractable Value (MEV) is any *excess value* captured by validators
  - **Reordering** transactions to adversarial, non-FIFO ordering
  - Strategically **adding** txns before and/or after other users’ txns
  - **Eliding** txns from particular bundles

- Currently managed via **off-chain auctions** (e.g. Flashbots)
  - **Pro**: Ensures low spam from strategic users rendering the network unusable for non-strategic/passive users
  - **Con**: Adds centralization vector since auction isn’t {credible, verifiable}
Transaction Flows in Blockchains

Transaction mempool
\[ \Delta_1, \ldots, \Delta_N \]

MEV Searcher
\[ [\Delta^x, b^x] = f(\Delta_1, \ldots, \Delta_N) \]

Validator

Blockchain

Block 1

Block 2

...
But how do we know when value captured via MEV is excessive?
Describing Value Flows in MEV is difficult

- Do we do optimize for the welfare of users or the revenue of validators?
  - User welfare is important for network success
  - Validator revenue is important for economic security
    - Competing goals for any decentralized network
- What pieces are missing in our description of MEV?
  - User utilities: How much users intrinsically value a particular txn(s)
  - User payments: Set of transaction fees that users are willing to pay
  - Allocation: How an auctioneer (Flashbots, Proposer in PBS) allocates block space to users
- Components are dependent on the applications involved
  - E.g. utilities for NFT minters and DeFi traders are very different
Formalizing Value in MEV

- **Formalism:**
  - Allocation of block space to users: \( x_1, \ldots, x_n \in \{0,1\} \)
    - \( x_i = 1 \) if the \( i \)th user’s transaction makes it in
  - Utilities, payments of users: \( u_1, \ldots, u_n, p_1, \ldots, p_n \)
    - Note: \( p_i = p_i(x_1, \ldots, x_n) \)

- **Social Welfare:**
  \[
  SW(x) = E[u_1(x_1)x_1 + \cdots + u(x_n)x_n]
  \]

- **Revenue:**
  \[
  Rev(p) = E[p_1 + \cdots + p_n]
  \]

- **Equilibria:** \((x^*, p^*) \in Eq = \{(x,y) \in (\arg\max_x SW(x), \arg\max_p Rev(p))\}\)
  - If \( p_i^* \neq p_i(x^*) \), then we do not have a consistent equilibria for user welfare maximization and revenue maximization!
    - Happens all the time in MEV!
Quantifying MEV at the system level

- Quantifying how good or bad a set of equilibria via approximation ratios, such as the Price of Anarchy

\[
\text{POA} = \frac{\sup_{x^*, p^* \in \text{Eq}} \sum_{i=1}^{n} p_i^* - p_i(x^*)}{\inf_{x^*, p^* \in \text{Eq}} \sum_{i=1}^{n} p_i^* - p_i(x^*)}
\]

- In words: How many times more is the worst case deviation between optimal price and the user’s demanded price relative to the best case?

- **POA = \(O(1)\) is good, POA = \(o(n)\) is okay, POA = \(\Omega(n)\) is bad**
  - This quantity depends deeply on how you define the utilities and prices paid, which is application-specific and not uniquely defined
Price of Anarchy in examples:

Braess's Paradox
Braess’s Paradox: A Tale of 4 Cities

- Traffic network with 4 cities, 4 roads
- Two roads have travel times dependent on the percentage of traffic on that road (x)
- Assume each driver is myopic and selfish
  - Given that the two paths take $1 + x$ time to travel from S to D, the expected time is $3/2$

Equilibrium:

\[ 1 + \alpha^* = 1 + 1 - \alpha^* \]

\[ \Rightarrow \alpha^* = \frac{1}{2} \]

Net cost:

\[ \frac{1}{2} \left( 1 + \frac{1}{2} \right) + \frac{1}{2} \left( 1 + \frac{1}{2} \right) \]

\[ = \frac{1}{2} + \frac{1}{4} + \frac{1}{2} + \frac{1}{4} = \frac{3}{2} \]
Adding a road worsens congestion

- You might think that adding an extra road will improve congestion — but that’s not always the case!
- Adding a road that takes 0 time to traverse between the middle cities, if everyone is greedily optimizing, leads to congestion on the same path!

Equilibrium:

\[
\alpha^* + 1 - 2\alpha^* + 1 = \alpha^* + 1 - 2\alpha^* + 1 - 2\alpha^* \\
\implies \alpha^* = 0
\]

Net cost:

\[
1 \cdot (1 + 1) = 2
\]
At this point, you might think...

MEV is *always* bad
Application Specificity can give MEV positive externalities

- **Not all applications are the same**
  - MEV allows for strategic users to make a profit while simultaneously improving the welfare of non-strategic users

- **Routing across multiple contracts is hard for non-strategic users**
  - Finding optimal routes for trading has been hard for non-strategic users
    - Reliance on 3rd party services like 1inch, Matcha, Gem, etc.
    - Algorithmic Game Theory has studied selfish routing for decades: does any of it apply to MEV?

- **Example**: Braess’s Paradox for selfish routing
  - “Sometimes adding more capacity can slow the network down if the incentives aren’t tuned correctly”
The **Inverse Braess Paradox**

Now add some small congestion cost on the middle link ($\varepsilon$, due to MEV)

**Counterintuitively**: these costs actually *improve* the overall network flow!

e.g. by disincentivizing bad selfish behavior

Equilibrium:

$$\alpha^* + 1 - 2\alpha^* + 1 = \alpha^* + 1 - 2\alpha^* + (\varepsilon + 1)1 - 2\alpha^*$$

$$\implies \alpha^* = \frac{\epsilon}{2(\varepsilon + 1)} \to \frac{1}{2} \text{ as } \varepsilon \to \infty$$

Net cost:

$$\to \frac{3}{2} \text{ as } \varepsilon \to \infty$$
Inverse Braess Paradox for CFMMs (e.g. Uniswap)

Replace ‘travel times’ in the road network w/ CFMM price impact function

- Travel routes are token trades e.g. A→B→C
- Congestion is many users trying to trade on the same link
- Sandwich attacks are equal to adding congestion on a link $\varepsilon$

Shockingly even w/ sandwiches:
PoA = $O(1)$

**Theorem 2.** Suppose that $f(\kappa, \mu, \eta), g(\kappa, \mu, \eta) \in O((1 + (\alpha \beta \kappa)^{O(1)})^{1/diam(G)}$. Then there exists a function $C(\kappa, \alpha, \beta, \mu, \eta)$ that is constant in the size of the network graph $G$ such that

$$\text{PoA}(\Delta) \leq C(\kappa, \alpha, \beta, \mu, \eta)$$

(23)

Towards a Theory of Maximal Extractable Value I: CFMMs
Kulkarni, Diamandis, C, 2022
If MEV is not so bad, how do we harness it for good?
Redistributing MEV to validators can let you lower inflation!

Redistributing MEV (e.g. sharing a percentage of captured MEV pro-rata with validators) increases the stickiness of staked/delegated capital.

![Graph showing inflation rate and epoch time relationship](image.png)
Multidimensional MEV Auctions can improve allocative efficiency

Now: Ethereum bundles MEV into one auction — Uniswap arbitrage competes with NFT minting

Future: Rollups or apps run their own auctions that get aggregated (*hierarchical PBS*)

α-leak: C, Kulkarni, Ferreira (unpublished, 2022) prove that disaggregation can sometimes improve auction efficiency

*Unbundle* MEV to improve social welfare in MEV auctions
Theoretical Foundations of MEV are important!

- **Then**: MEV started at an emergent/unstudied phenomena
- **Now**: Design space for redistributing and optimizing MEV for users relies on theoretical understanding of networks
  - Surprising formalized truths evince that **MEV isn’t always bad**!
- Without algorithmic game theory and probability theory, it is hard to reason about such truths
  - Our papers are just the beginning!
- **Open Problems**:
  - **Optimal** auctions
  - Information theoretic **lower bounds**
  - Aggregation vs. Disaggregation effects
    - “What is the Coase theorem for MEV?”
Thank you!

P.S. We launched Aera today!
aera.finance

@tarunchitra
@gauntletnetwork