Blockchain Game Theory
Incentives, Fees, and MEV
Day 3 of 5

Prof. Jörg Osterrieder

MSc Seminar: Digital Finance

Spring 2026

Where We Are

Days 1–2 Recap

- Crypto market structure and price dynamics
- DeFi and AMM mechanics (constant-product formula)
- Impermanent loss quantification
- Option pricing on crypto underlyings

Today's Roadmap

1. Consensus as a strategic game
2. EIP-1559 mechanism design
3. MEV: formalization and attacks
4. Tokenomics and staking economics

The $25M FBI Sting: Operation Token Mirrors

October 2024

The FBI created a **fake cryptocurrency token** (NexFundAI) as a sting operation, complete with a fabricated website and market-making activity.

- Undercover agents posed as token promoters
- Target: firms offering **wash-trading** and **pump-and-dump** services for token issuers
- Defendants manipulated prices, created fake volume, and front-ran their own clients
- Result: **18 individuals charged**, $25M+ in assets seized

Why This Matters for Today

Front-running is not just a legal problem—it is an *economic equilibrium* embedded in blockchain architecture.

From Crime to Mechanism: MEV at Scale

Operation Token Mirrors exposed manual manipulation.

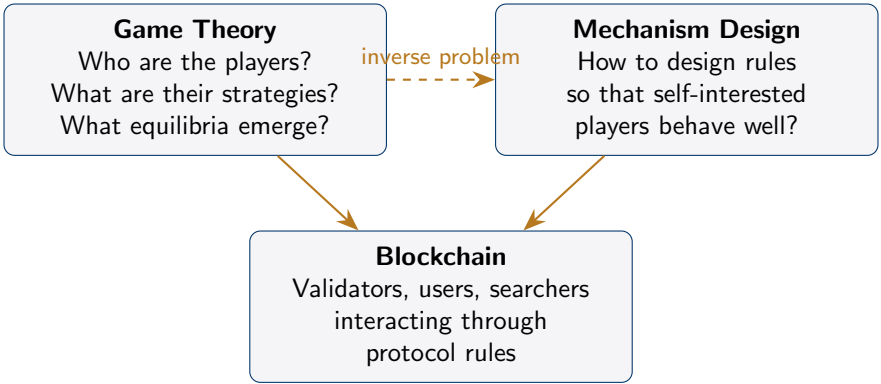
But on Ethereum, *automated* extraction happens every block:

- Bots scan the mempool for profitable reordering
- Validators can choose which transactions to include
- This is **Maximal Extractable Value (MEV)**

MEV by the Numbers (2023–24)

- $600M+ extracted annually on Ethereum
- 30–40% of blocks contain MEV
- Average sandwich profit: $3–$50 per attack
- Largest single MEV: $3.4M (March 2023)

The Game-Theoretic Lens



Today we use **both** to understand fees, consensus, and MEV.

Today: Formalize with Game Theory and Mechanism Design

1. Consensus Games
2. EIP-1559 Fee Mechanism
3. MEV Formalization
4. Tokenomics
5. Hands-On: EIP-1559 Simulation & MEV Analysis

Validators as Strategic Players

Setup. A blockchain has n validators, each with stake fraction α_i (PoS) or hash-rate fraction (PoW).

- **Players:** validators $i \in \{1, \ldots, n\}$
- **Strategy set:** $S_i = \{\text{honest}, \text{ deviate}\}$
- **Honest:** follow the protocol (propose valid blocks, attest correctly)
- **Deviate:** selfish mining, double-spend attempts, MEV extraction beyond protocol norms

Definition 1 (Normal-Form Game)

A consensus game is $\Gamma = (N, \{S_i\}_{i \in N}, \{u_i\}_{i \in N})$ where $u_i : S_1 \times \cdots \times S_n \to \mathbb{R}$ is validator i's payoff.

Payoff Structure

Parameters:

- R: block reward (e.g., 2 ETH post-Merge + tips)
- c_i: operating cost for validator i
- α_i: stake fraction $\Rightarrow$ probability of being selected
- p: penalty (slashing) for detected deviation

Expected single-round payoffs:

$$u_i(\text{honest}) = \alpha_i \cdot R - c_i$$

$$u_i(\text{deviate}) = \alpha_i \cdot (R + \Delta) \cdot (1 - q) - c_i - q \cdot p$$

where Δ is the extra MEV from deviation and q is the detection probability.

Key insight: Honest is optimal when $q \cdot p > \alpha_i \cdot \Delta \cdot (1 - q)$.

The Honest-vs-Selfish Mining Game

Consider two pools (A and B) each controlling roughly 50%.

		Pool B	
		Honest	Selfish
Pool A	Honest	$R/2 - c$ $R/2 - c$	$R/2 - c - L$ $R/2 + \Delta - c'$
	Selfish	$R/2 + \Delta - c'$ $R/2 - c - L$	$R/2 + \Delta' - c'$ $R/2 + \Delta' - c'$

= Pool A payoff

= Pool B payoff

L: loss from competitor's selfish mining (orphaned blocks).
$\Delta' < \Delta$: when both deviate, competition erodes gains.

Nash Equilibrium Analysis

Definition 2 (Nash Equilibrium)

A strategy profile $s^* = (s_1^*, \ldots, s_n^*)$ is a Nash equilibrium if no player can unilaterally improve their payoff:

$$u_i(s_i^*, s_{-i}^*) \geq u_i(s_i', s_{-i}^*) \quad \forall\, i,\ \forall\, s_i' \in S_i$$

Without slashing:

- If $\Delta > 0$, *deviate* weakly dominates *honest*
- Unique NE: (Selfish, Selfish) — a Prisoner's Dilemma!
- Both pools earn less than under mutual honesty

With slashing (p large, q high):

- Deviation payoff drops: u_i(deviate) decreases by $q \cdot p$
- When $q \cdot p > \Delta$: (Honest, Honest) becomes the unique NE

Mechanism design goal: Set p and q so honesty is incentive-compatible.

Proof of Stake: Slashing as Enforcement

Ethereum PoS parameters (post-Merge):

- Minimum stake: 32 ETH ($\approx$ \$60k at current prices)
- **Inactivity leak:** gradual penalty for offline validators
- **Slashing:** $\geq 1/32$ of stake destroyed for provable misbehavior
- **Correlation penalty:** if many validators misbehave simultaneously, penalty scales up to *entire stake*

Incentive Compatibility Condition

Validator i prefers honesty when:

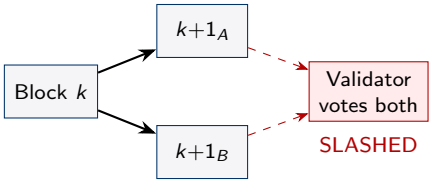
$$\underbrace{\alpha_i \cdot R}_{\text{honest reward}} > \underbrace{\alpha_i \cdot (R + \Delta) \cdot (1 - q)}_{\text{deviation reward}} - \underbrace{q \cdot p}_{\text{expected penalty}}$$

Ethereum targets $q \approx 1$ for detectable deviations (e.g., double voting).

The Nothing-at-Stake Problem

Problem (naïve PoS):

- In PoW, mining on two forks costs double the electricity
- In naïve PoS, voting on *both* forks is free
- Rational strategy: validate every fork $\Rightarrow$ guaranteed reward
- Consensus breaks: no fork is ever abandoned



Solutions:

1. **Slashing conditions:** punish validators caught voting on conflicting forks (Casper FFG)
2. **Finality:** once 2/3 of validators attest, block is irreversible ($\leq$ 2 epochs $\approx$ 12.8 min)

The Fee Market Problem: First-Price Auctions

Pre-EIP-1559 (before August 2021):

- Users bid gas prices; miners include highest bidders
- This is a **first-price sealed-bid auction**
- Problem: users don't know others' bids $\Rightarrow$ strategic overbidding

Consequences:

1. **Gas wars:** during NFT mints, fees spiked to \$100+ per transaction
2. **Bid uncertainty:** users either overpay or get stuck for hours
3. **Revenue volatility:** miner income wildly unpredictable

Mechanism Design Challenge

Design a fee mechanism that is *incentive-compatible* (truthful bidding), *efficient* (right transactions get in), and *budget-balanced*.

EIP-1559: Base Fee + Priority Tip [ ]

Two-part fee structure:

$$\text{Fee}_t = \underbrace{b_t}_{\text{base fee}} + \underbrace{\delta_i}_{\text{priority tip}}$$

- b_t: algorithmically determined, same for all transactions in block t
- δ_i: user-chosen tip to incentivize inclusion priority
- Base fee is **burned** (destroyed), not paid to validator
- Validator receives only the tip δ_i

User's Decision

If your maximum willingness to pay is v_i, set:

- Max fee: v_i (cap on total fee)
- Priority tip: $\delta_i = v_i - b_t$ (if $v_i > b_t$, else don't transact)

Truthful bidding is a (weakly) dominant strategy!

Base Fee Dynamics

Update rule (block target $g^* = 15\text{M}$ gas, max $= 30\text{M}$):

$$\boxed{b_{t+1} = b_t \cdot \left(1 + \frac{1}{8} \cdot \frac{g_t - g^*}{g^*}\right)}$$

- g_t: actual gas used in block t
- Block full ($g_t = 30\text{M}$): $b_{t+1} = b_t \cdot 1.125$ $\quad (+12.5\%)$
- Block empty ($g_t = 0$): $b_{t+1} = b_t \cdot 0.875$ $\quad (-12.5\%)$
- Block at target ($g_t = g^*$): $b_{t+1} = b_t$ $\quad$ (no change)

Properties:

- Exponential adjustment $\Rightarrow$ fast convergence to market-clearing price
- Maximum $\pm 12.5\%$ per block $\Rightarrow$ bounded volatility
- Long-run equilibrium: blocks are on average 50% full

Incentive Compatibility

Theorem 3 (Roughgarden, 2021)

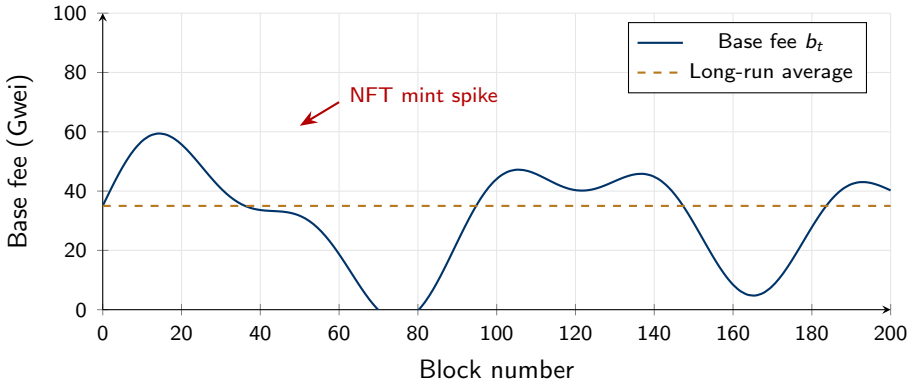
Under EIP-1559, for any user i with value v_i and base fee $b_t < v_i$:

1. *Bidding tip $\delta_i = v_i - b_t$ is a weakly dominant strategy*
2. *No user benefits from misreporting their value*
3. *The mechanism is off-chain agreement proof: no coalition of users and a single validator can profitably deviate*

Intuition:

- The base fee is "take it or leave it" — you can't negotiate it down
- Your tip only determines priority *within* a block
- Since base fee adjusts to clear the market, your bid doesn't affect future prices

Base Fee Over 200 Blocks



When demand surges, base fee rises; as demand normalizes, it returns to equilibrium. The 12.5% cap per block smooths transitions.

The Burn: Base Fee Destruction

Pre-EIP-1559: All fees go to miners $\Rightarrow$ ETH supply only grows.

Post-EIP-1559: Base fee is burned $\Rightarrow$ ETH is destroyed every block.

$$\text{Net ETH issuance} = \underbrace{\text{Block rewards}}_{\text{issuance}} - \underbrace{b_t \cdot g_t}_{\text{burn per block}}$$

Issuance rate (PoS):

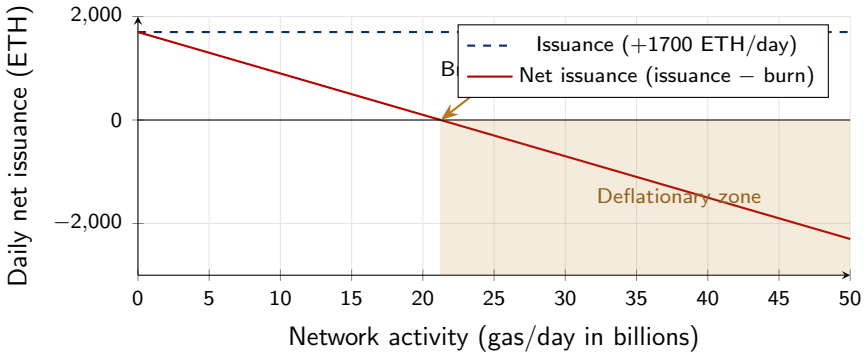
- $\approx$ 1,700 ETH/day (staking rewards)
- Fixed by protocol, depends on total staked

Burn rate:

- $\approx$ 1,500–3,000 ETH/day (varies with demand)
- High activity $\Rightarrow$ deflationary
- Low activity $\Rightarrow$ mildly inflationary

Since the Merge (Sept 2022), total ETH supply has *decreased* by $\approx$400k ETH.

Is ETH Deflationary?



ETH becomes deflationary when daily gas consumption exceeds the breakeven point—typically during high DeFi/NFT activity.

Checkpoint

Quick Question

If all blocks are exactly at target gas for 10 consecutive blocks, what happens to the base fee?

Checkpoint

Quick Question

If all blocks are exactly at target gas for 10 consecutive blocks, what happens to the base fee?

Answer

The base fee **stays constant**. The update rule multiplies by $(1 + \frac{1}{8} \cdot \frac{g - g^*}{g^*})$. When $g = g^*$ (actual gas equals target), the adjustment factor is exactly 1, so $b_{t+1} = b_t$. The base fee only changes when blocks are above or below target utilization.

MEV: Formal Definition

Definition 4 (Maximal Extractable Value)

Given a set of pending transactions $\mathcal{T}$ and the set of all valid orderings $\Pi(\mathcal{T})$:

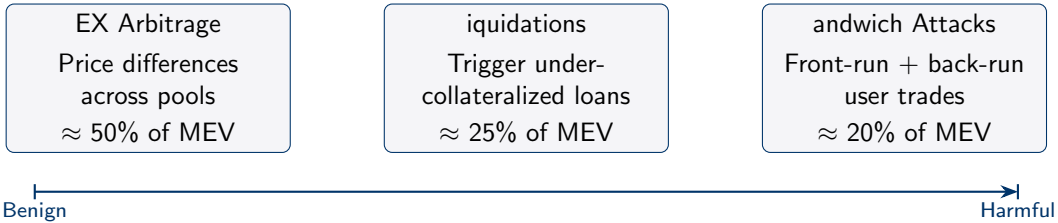
$$\text{MEV} = \max_{\pi \in \Pi(\mathcal{T})} V(\pi) - V(\pi_{\text{default}})$$

where $V(\pi)$ is the total value captured by the block producer under ordering π, and π_{default} is the canonical ordering (e.g., by arrival time).

Key insight: The block producer has a *monopoly* on transaction ordering within their block.

- They can insert, reorder, or censor transactions
- MEV ≥ 0 by definition (producer can always choose π_{default})
- In practice, **searchers** find MEV and share profits with producers

Types of MEV Extraction



- **Arbitrage:** corrects mispricings, arguably beneficial
- **Liquidations:** maintains DeFi solvency, competitive market
- **Sandwiches:** pure value extraction from users, no social benefit

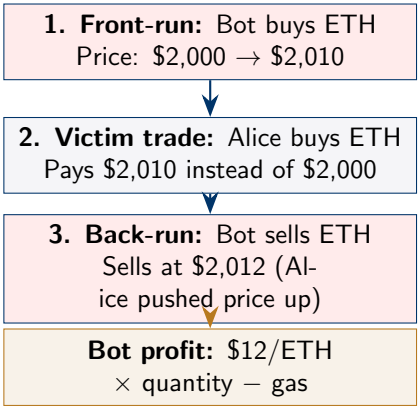
AMM Refresher (from Day 2)

- Constant product: $x \cdot y = k$
- A trade of Δ tokens changes price: larger trades $\Rightarrow$ more slippage
- Price impact: slippage $\approx \frac{\Delta}{x+\Delta}$

Why this matters for MEV

Sandwich attacks exploit the deterministic price impact of AMM trades. Understanding the constant-product formula is essential for what follows.

Sandwich Attack: Step by Step



Ordered *before* Alice

Alice's slippage = bot's profit

Ordered *after* Alice

Sandwich Attack Profitability

For a constant-product AMM with reserves (x, y) and fee γ:

Bot's profit from sandwiching a trade of size Δ_{user}:

$$\pi_{\text{sandwich}} = \underbrace{s \cdot \Delta_{\text{user}}}_{\text{slippage captured}} \times \underbrace{(1-\gamma)^2}_{\text{AMM fee loss}} - \underbrace{g_{\text{front}} + g_{\text{back}}}_{\text{gas costs}}$$

where s is the price impact of the user's trade:

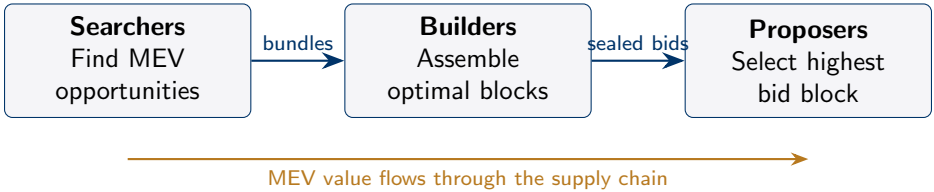
$$s = \frac{\Delta_{\text{user}}}{x + \Delta_{\text{user}}}$$

Profitability conditions:

- Larger $\Delta_{\text{user}} \Rightarrow$ more slippage to capture
- Smaller pool (lower x) $\Rightarrow$ higher price impact
- Higher gas costs $\Rightarrow$ minimum trade size threshold
- Typical breakeven: user trade $>$ \$5,000 on medium-sized pools

Proposer-Builder Separation (PBS)

Problem: Validators who extract MEV themselves become centralized (economies of scale in MEV search).



PBS separates:

- *Block building* (competitive, specialized) from *block proposing* (random, decentralized)
- Proposers don't need MEV expertise; they just pick the highest bid
- Currently via MEV-Boost (out-of-protocol); enshrined PBS in development

Token Valuation: Equation of Exchange [ ]

Fisher's equation adapted for crypto:

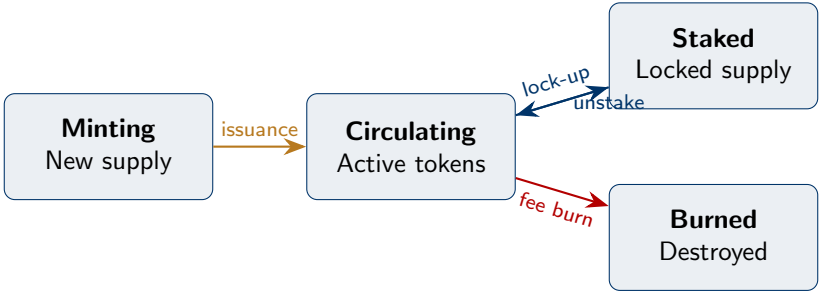
$$\boxed{M \cdot V = P \cdot Q} \qquad \Longrightarrow \qquad P_{\text{token}} = \frac{P \cdot Q}{V \cdot S}$$

- M: total token market cap
- V: velocity (how often each token changes hands per period)
- $P \cdot Q$: platform GDP (total value of on-chain economic activity)
- S: circulating supply
- $P_{\text{token}} = M/S$: price per token

Key levers:

1. **Grow GDP:** more usage $\Rightarrow$ higher token value
2. **Reduce velocity:** staking, lock-ups $\Rightarrow$ tokens held longer
3. **Reduce supply:** burns, halving schedules

Token Supply Dynamics



$$S_{\text{effective}} = S_{\text{minted}} - S_{\text{burned}} - S_{\text{staked}}$$

- **Inflationary:** Bitcoin (pre-2140), most PoS chains
- **Deflationary pressure:** ETH burns, BNB quarterly burns
- **Velocity sinks:** staking, DeFi lock-ups, governance voting

Staking: Lock Supply, Earn Yield

Why do validators stake?

$$\mathbb{E}[\text{Annual yield}] = \frac{R_{\text{total}}}{S_{\text{staked}}} \approx \frac{\text{issuance} + \text{tips} + \text{MEV}}{S_{\text{staked}}}$$

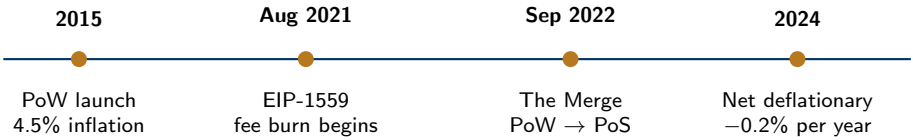
ETH staking (2024 data):

- ≈28M ETH staked (23% of supply)
- APY: ≈3.5–4.5%
- Higher MEV periods ⇒ higher APY
- Liquid staking (Lido) ≈32% share

Economic effects:

- Reduces circulating supply ⇒ lower velocity
- Security budget: total staked × slash risk
- More stakers ⇒ lower yield ⇒ equilibrium
- Liquid staking tokens (stETH) re-enter DeFi

Case Study: ETH Monetary Policy Shift



PoW era (2015–2022):

- 2 ETH/block × 6,400 blocks/day
- ≈4.5M ETH/year issuance
- Always inflationary

PoS + EIP-1559 (2022–):

- Issuance cut by ∼90%
- Burn often exceeds issuance
- ETH now has "ultrasound money" properties

Hands-On Session

EIP-1559 Simulation & MEV Analysis

Python notebooks on course platform

Exercise: Simulate EIP-1559 Base Fee (1/5)

Step 1: Implement the base fee update.

Python skeleton

```
def update_base_fee(b_t, gas_used, gas_target=15_000_000):
    """EIP-1559 base fee update rule."""
    delta = (gas_used - gas_target) / gas_target
    b_next = b_t * (1 + delta / 8)
    return max(b_next, 1)  # floor at 1 Gwei
```

Tasks:

1. Simulate 500 blocks with gas demand drawn from $g_t \sim \mathcal{N}(15\text{M}, 5\text{M}^2)$, clipped to $[0, 30\text{M}]$
2. Plot the base fee trajectory
3. What happens when you increase demand to $\mathcal{N}(25\text{M}, 3\text{M}^2)$?

Exercise: Demand Shock Analysis (2/5)

Step 2: Simulate an NFT-mint demand shock.

- Blocks 1–200: normal demand $\mathcal{N}(15\text{M}, 5\text{M}^2)$
- Blocks 201–250: spike to $\mathcal{N}(28\text{M}, 2\text{M}^2)$
- Blocks 251–500: return to normal

Questions:

1. How quickly does the base fee respond to the shock?
2. How many blocks until the base fee returns to pre-shock levels?
3. Compute total ETH burned during the spike vs. normal period.
4. What would happen under the old first-price auction?

Expected result: Base fee peaks $\sim$5–10$\times$ above normal, then decays exponentially over $\sim$30–50 blocks.

Exercise: MEV Extraction Analysis (3/5)

Step 3: Quantify sandwich attack profitability.

Setup

```
# Constant-product AMM
x, y = 10_000_000, 10_000_000  # $10M each side
fee = 0.003  # 0.3% Uniswap fee
gas_cost = 50  # $ per transaction
```

Tasks:

1. Implement a swap function for a constant-product AMM
2. Compute sandwich profit for user trades of \$1k, \$5k, \$10k, \$50k, \$100k
3. Find the minimum trade size where sandwich is profitable

Exercise: Pool Depth vs. Profit (4/5)

Step 4: Sensitivity analysis.

Vary pool liquidity and user trade size:

- Pool sizes: \$1M, \$5M, \$10M, \$50M, \$100M
- User trades: \$1k to \$500k (log scale)
- Gas costs: \$10, \$50, \$100

Create a heatmap:

- x-axis: user trade size (log)
- y-axis: pool liquidity (log)
- Color: sandwich profit (\$)
- Contour line at profit $= 0$ (breakeven)

Key finding: Shallow pools are sandwich goldmines. Deep pools protect users.

Exercise: Game-Theoretic Analysis (5/5)

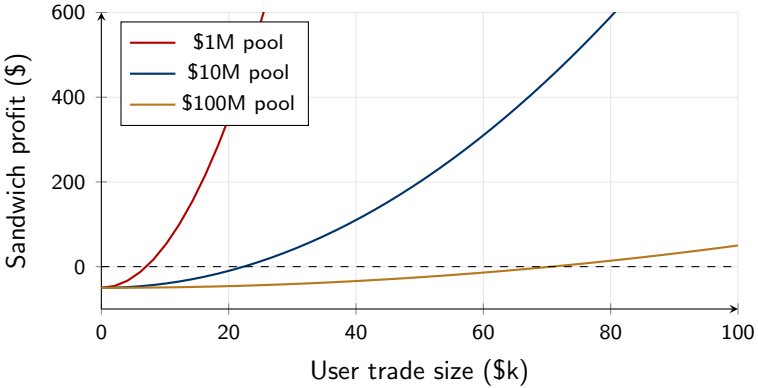
Step 5: Model the validator's MEV decision.

1. Construct the 2-player game from Slide 9 with specific values: $R = 2$ ETH, $\Delta = 0.5$ ETH, $c = 0.01$ ETH
2. Find the Nash equilibrium for $p \in \{0, 0.5, 1, 5\}$ ETH
3. Plot: minimum slashing penalty to sustain honesty as a function of MEV opportunity Δ
4. Under what conditions does the Prisoner's Dilemma structure break?

Discussion

Should MEV be eliminated, redistributed to users, or treated as a fee? What are the implications for CBDC transaction ordering?

Sandwich Profitability: Key Results



Profit scales quadratically with trade size and inversely with pool depth. Gas cost creates a minimum viable trade size.

Discussion: MEV Redistribution and CBDC Design

MEV Redistribution Proposals:

1. **MEV-Share:** searchers return a portion of MEV to users
2. **Order flow auctions:** users sell their order flow
3. **Encrypted mempools:** prevent front-running entirely
4. **Fair ordering:** Chainlink FSS (first-come-first-served)

CBDC Fee Design:

- Central banks will design transaction fees
- Should CBDCs have EIP-1559-style dynamic fees?
- Who orders CBDC transactions — and can they extract MEV?
- Privacy vs. ordering transparency trade-off

Debate Prompt

"MEV is an unavoidable consequence of decentralized transaction ordering. Attempting to eliminate it creates worse outcomes than managing it."

Day 3 Summary

1. **Consensus games:** validators are strategic players; slashing enforces honest equilibria
2. **EIP-1559:** replaced first-price auctions with incentive-compatible base fee + tip
3. **MEV:** value from transaction ordering; sandwich attacks, arbitrage, liquidations
4. **PBS:** separates block building from proposing to preserve decentralization
5. **Tokenomics:** equation of exchange, supply dynamics, staking as velocity sink

Day 4 Preview: ML for Crypto Markets

From game theory to data science: building, evaluating, and deploying machine learning models for crypto prediction. Backtesting pitfalls, SHAP explainability, and realistic expectations.

References I

[1] Lin William Cong, Ye Li, and Neng Wang. "Tokenomics: Dynamic Adoption and Valuation". In: *Review of Financial Studies* 34.3 (2021), pp. 1105–1155.

[2] Tim Roughgarden. "Transaction Fee Mechanism Design". In: *ACM SIGecom Exchanges* 19.1 (2021).