The Economics of Blockchain
Incentives, Fees, and Digital Money
Day 3 of 5

Prof. Jörg Osterrieder

BSc Seminar: Digital Finance

Spring 2026

PhD Seminar Series: Digital Finance Research

Where We Are

Days 1–2 Recap

- What are cryptocurrencies and why they matter
- How DeFi replaces banks with smart contracts
- AMMs: swap tokens without a middleman
- Impermanent loss and liquidity provision

Today's Roadmap

1. How blockchains really work (consensus)
2. Transaction fees: the EIP-1559 revolution
3. MEV: the invisible tax on your trades
4. CBDCs and stablecoins: the future of money

The $50 Invisible Tax

True Story (Happens Thousands of Times Per Day)

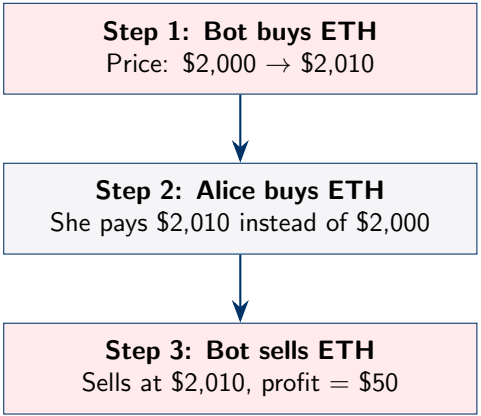
Alice wants to swap $5,000 of USDC for ETH on Uniswap. She submits her transaction and waits for confirmation...

- A **bot** spots Alice's pending transaction in the public queue
- The bot *jumps in front*, buys ETH first, pushing the price up
- Alice's trade executes at a *worse* price — she gets less ETH
- The bot immediately sells, pocketing the difference: **$50 profit**

This is called a "sandwich attack."

Alice never sees it happen. No notification. No receipt. Just less ETH.

How a Sandwich Attack Works

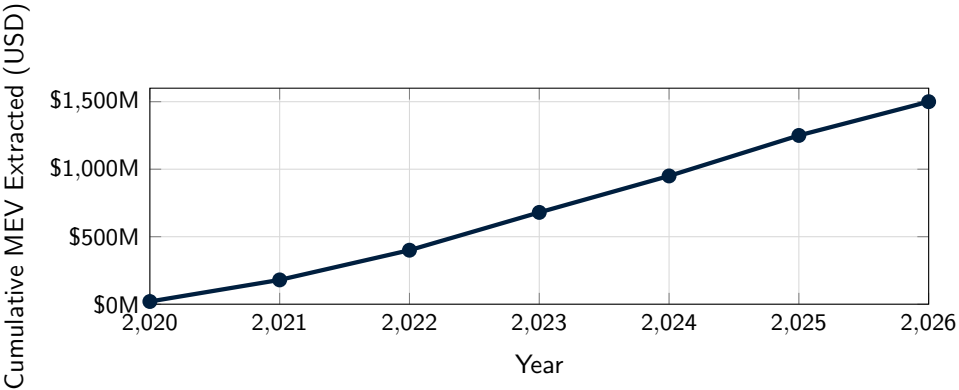


Before Alice's trade

Alice's trade (worse price)

After Alice's trade

How Big Is This Problem?



Over $1.5 billion extracted from users on Ethereum alone.

Source: Flashbots MEV-Explore, EigenPhi (illustrative estimates).

Is This Fair?

"It's Theft!"

- Users lose money without consent
- Bots exploit information asymmetry
- Ordinary users can't compete
- Undermines trust in DeFi

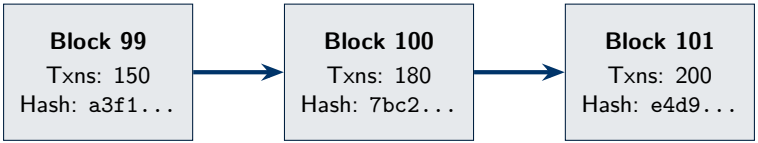
"It's Just Market Forces"

- Arbitrage improves price efficiency
- Liquidations keep lending safe
- Anyone can run a bot (permissionless)
- Traditional finance has the same issue

Today we'll build the tools to analyze this debate.

Outline

Quick Recap: Blocks, Chains, Hashes



Each block contains a fingerprint (hash) of the previous block

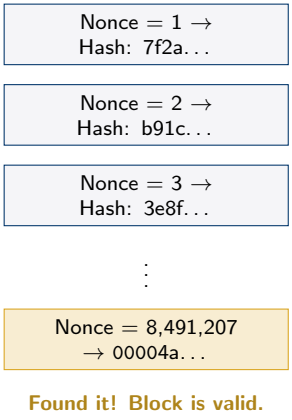
- A **blockchain** is a linked list of transaction batches
- **Hash**: a one-way fingerprint — change one byte and the hash changes completely
- Tampering with an old block breaks every subsequent hash $\Rightarrow$ **immutability**

Proof of Work: Mining for Blocks

The process:

1. Collect pending transactions into a block
2. Try random numbers ("nonces") until the hash starts with enough zeros
3. First miner to find a valid nonce wins
4. Winner broadcasts block, gets **reward + fees**

Think of it like a lottery:
Buy more tickets (computing power) $\Rightarrow$ higher chance of winning.



Why Is Proof of Work Secure?

The 51% Attack

To cheat (e.g., reverse a transaction), an attacker needs **more than 50%** of the network's total computing power.

- Bitcoin's network: millions of specialized machines worldwide
- Cost to attack Bitcoin: estimated **$10+ billion** in hardware alone
- Plus ongoing electricity costs
- **Economic logic**: it's more profitable to play by the rules

Key Insight

Blockchains don't assume people are honest. They make *cheating more expensive than cooperating*. This is **incentive design** at work.

Proof of Stake: Lock Tokens Instead of Burning Electricity

The idea: instead of solving puzzles, validators lock up ("stake") their own tokens as collateral.

- Validators are chosen proportional to their stake
- If they cheat $\Rightarrow$ their staked tokens are **destroyed** ("slashing")
- Honest behavior $\Rightarrow$ earn staking rewards (currently $\sim$3–5% APY on Ethereum)

Ethereum's Transition (Sep 2022, "The Merge")

Ethereum switched from PoW to PoS, reducing energy consumption by $\sim$99.95%.

Security comes from economic skin-in-the-game, not electricity.

PoW vs. PoS: Side-by-Side

	Proof of Work	**Proof of Stake**
Used by	Bitcoin	Ethereum (since 2022)
Security from	Computing power	Staked capital
Energy use	Very high	Very low
Hardware	Specialized ASICs	Standard computer
Punishment	Wasted electricity	Tokens destroyed
Rewards	Block reward + fees	Staking yield + fees
Min. to attack	51% of hashrate	33% of staked ETH
Centralization risk	Mining pools	Large stakers

Both rely on the same principle: make cheating more expensive than cooperating.

Outline

The Old System: Gas Auctions

Before August 2021, Ethereum used a first-price auction for transaction fees:

- You bid a gas price. Miners pick the highest bids first.
- **Problem 1**: You don't know what others are bidding
- **Problem 2**: Bid too low $\Rightarrow$ your transaction is stuck
- **Problem 3**: Bid too high $\Rightarrow$ you overpay massively
- During NFT drops or market crashes: **"gas wars"** with fees of \$50–\$500+

The Result

Fees were unpredictable, unfair, and wildly volatile. Average users were priced out during peak times.

EIP-1559: A Smarter Fee System

Key idea: replace the guessing game with a transparent, auto-adjusting price.

Base Fee (set by the protocol)

- Adjusts automatically every block
- Goes UP when blocks are full
- Goes DOWN when blocks are empty
- Everyone pays the same base fee

Priority Tip (set by the user)

- Optional tip to the validator
- Want faster inclusion? Tip more
- Small amount, usually \$0.01–\$0.50
- Like tipping a waiter for faster service

Total fee = Base Fee + Priority Tip

The Base Fee Adjustment Formula

How the base fee updates each block

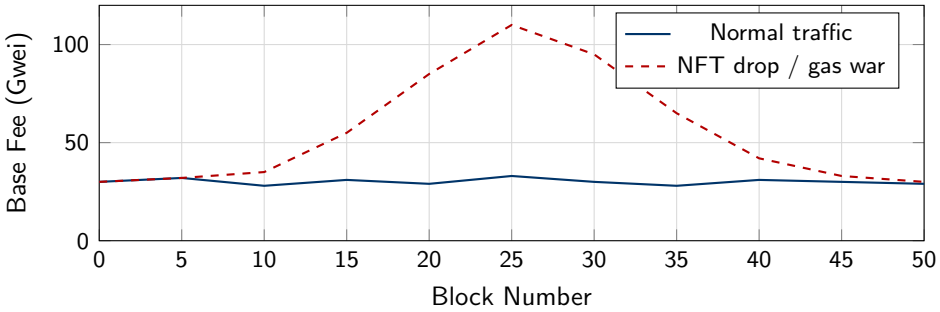
$$\text{base}_{t+1} = \text{base}_t \times \left(1 + \frac{1}{8} \times \frac{\text{gas\_used}_t - \text{target}_t}{\text{target}_t}\right)$$

Intuition:

- Target = 50% full block (15M gas out of 30M max)
- Block is **full** (30M gas): base fee increases by $\frac{1}{8} = 12.5\%$
- Block is **empty** (0 gas): base fee decreases by 12.5%
- Block is **at target** (15M gas): base fee stays the same

This is a negative feedback loop: high demand $\rightarrow$ higher fees $\rightarrow$ less demand $\rightarrow$ lower fees. Like a thermostat for block space!

Visualizing Base Fee Dynamics

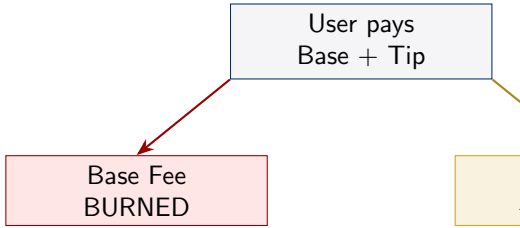


The base fee **spikes during congestion** but **automatically recovers**. No human intervention needed — pure algorithmic adjustment.

The Burn: Base Fee Is Destroyed

The key twist: the base fee is not paid to validators. It is **burned** — permanently removed from ETH supply.

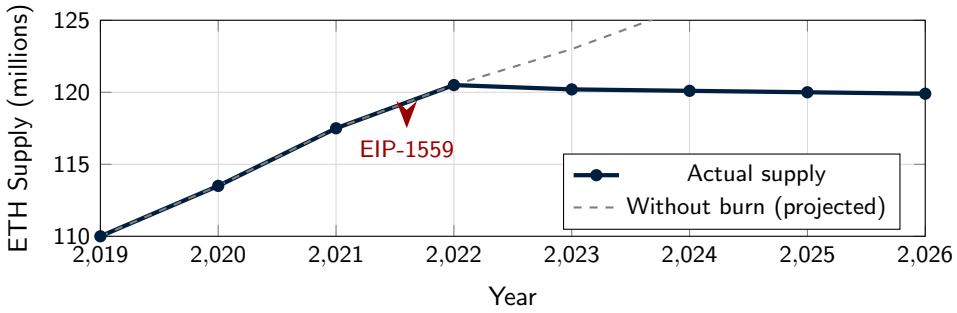
- **Why burn?** Prevents validators from artificially stuffing blocks to raise fees they'd collect
- Creates a **deflationary pressure** on ETH
- More network usage ⇒ more ETH burned
- On busy days, more ETH is burned than created



Since EIP-1559

Over 4 million ETH burned (∼$10B+).

ETH Supply: Before and After EIP-1559



Since "The Merge" + EIP-1559, ETH supply is roughly **flat to declining**.

Outline

Three Types of MEV

MEV = Maximal Extractable Value: profit from reordering transactions in a block.

Arbitrage	Liquidations	Sandwich Attacks
Price differs across DEXs. Bot buys low, sells high. **Verdict:** mostly helpful (aligns prices)	Borrower's collateral drops below threshold. Bot triggers liquidation. **Verdict:** necessary (keeps lending safe)	Bot front-runs AND back-runs a user's trade. **Verdict:** harmful (pure extraction)

Why Does MEV Exist?

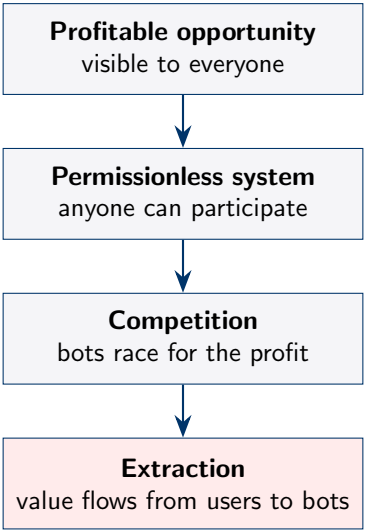
The Root Cause

Validators (block producers) have **complete control** over transaction ordering within their block.

1. Transactions sit in a public waiting room (the **mempool**)
2. Anyone can see pending transactions before they execute
3. Validators choose which transactions go first
4. If reordering creates profit $\Rightarrow$ someone will capture it

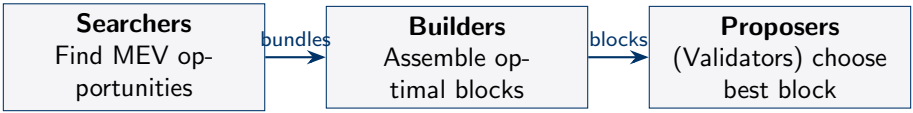
Analogy: Imagine a stock exchange where the person processing your order can *see* it and trade ahead of you. That's what the mempool enables.

Simple Game Theory: Why MEV Is Inevitable



Flashbots: Making MEV Transparent

If you can't eliminate MEV, can you make it fairer?



Proposer–Builder Separation (PBS):

- Separates "who builds the block" from "who proposes it"
- Creates a competitive market for block construction
- MEV revenue is shared more broadly instead of going to one party
- Users can submit to private channels to avoid being sandwiched

Debate: Is MEV a Bug or a Feature?

MEV Is a Bug

- Harms ordinary users
- Creates an uneven playing field
- Reduces trust in DeFi
- Can destabilize consensus (if MEV $>$ block reward)

MEV Is a Feature

- Arbitrage keeps prices efficient
- Liquidations maintain system health
- MEV funds validator security
- Mirrors traditional market making

Reality: MEV is a structural property of decentralized systems.
The goal is to *minimize harm* and *distribute value fairly*.

Outline

What Is a Stablecoin?

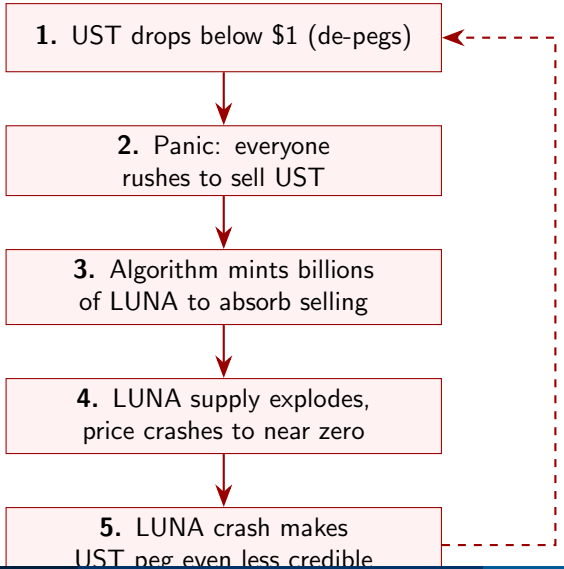
A cryptocurrency designed to maintain a **stable value**, usually pegged to \$1.

Type	Example	Backed By	Market Cap
Fiat-backed	USDT, USDC	Dollars in bank accounts	\$130B+
Crypto-backed	DAI	ETH & other crypto (over-collateralized)	\$5B
Algorithmic	(UST)	Algorithm + paired token	\$0 (collapsed)

Why do stablecoins matter?

- Bridge between crypto volatility and real-world pricing
- Enable DeFi lending, trading, and payments
- \$150B+ market: larger than many countries' money supply

Case Study: The Terra/Luna Death Spiral



CBDCs: Digital Money from Central Banks

Central Bank Digital Currency (CBDC)

A digital form of a country's fiat currency, issued and backed by the central bank. Think: digital cash, not crypto.

How it works:

- Issued by central bank (ECB, Fed, etc.)
- Runs on government infrastructure
- Not mined or staked
- Value guaranteed by the state
- Could replace physical cash

Examples in progress:

- Digital Euro (ECB, pilot phase)
- Digital Yuan (China, live in cities)
- e-Naira (Nigeria, launched 2021)
- FedNow (US, real-time payments)

CBDC vs. Cryptocurrency: Key Differences

	CBDC (e.g., Digital Euro)	Crypto (e.g., Bitcoin)
Issuer	Central bank	No one (decentralized)
Supply	Controlled by policy	Fixed or algorithmic
Privacy	Limited (gov. can track)	Pseudonymous / private
Stability	Stable (pegged to fiat)	Highly volatile
Permission	Permissioned	Permissionless
Finality	Instant	Minutes to hours
Trust model	Trust the government	Trust the math

They solve different problems: CBDCs modernize payments; crypto offers censorship resistance.

CBDCs: Promises and Risks

Potential Benefits

- **Faster payments**: instant, 24/7, cross-border
- **Financial inclusion**: 1.4 billion unbanked adults
- **Lower costs**: no card network fees
- **Better monetary policy**: direct stimulus possible
- **Reduce tax evasion**: traceable transactions

Potential Risks

- **Privacy**: government sees all transactions
- **Bank disintermediation**: why keep deposits at banks?
- **Digital bank runs**: one click to move all money
- **Censorship**: accounts can be frozen
- **Cybersecurity**: single point of failure

The Global CBDC Race

Status	Countries	Count
Launched	Nigeria, Bahamas, Jamaica, China (pilot)	11
Pilot phase	EU, India, Brazil, Russia, Australia	30+
Development	USA, UK, Canada, Japan, South Korea	40+
Research	Most others	60+
Total exploring		**130+**

Key fact: 98% of global GDP is represented by countries exploring CBDCs.

Source: Atlantic Council CBDC Tracker [3].

Outline

Hands-On Exercise

Simulating EIP-1559 Base Fee Dynamics

Python / Jupyter Notebook

Exercise: Simulate the EIP-1559 Base Fee

Goal: Implement the base fee adjustment formula and see how it responds to different demand scenarios.

Setup (provided in notebook)

- Initial base fee: 30 Gwei
- Target gas per block: 15,000,000
- Maximum gas per block: 30,000,000
- Simulate 100 blocks

Your tasks:

1. Implement the update formula in Python
2. Run two scenarios: normal traffic vs. gas war
3. Plot the base fee evolution
4. Calculate total ETH burned

Two Demand Scenarios

Scenario A: Normal Traffic

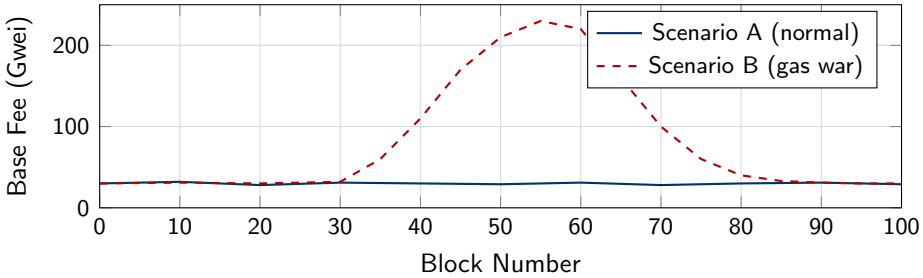
- Gas usage fluctuates randomly around the target
- gas_used $\sim \mathcal{N}(15\text{M},\ 3\text{M}^2)$
- Clipped to $[0, 30\text{M}]$

Scenario B: Gas War

- Blocks 1–30: normal traffic
- Blocks 31–60: NFT drop! Everyone wants in
- gas_used $= 28\text{M}$ (near max)
- Blocks 61–100: demand returns to normal

Prediction before coding: What do you think will happen to the base fee in each scenario? Sketch your guess on paper first!

Expected Output: Base Fee Over 100 Blocks



Your plot should look similar. The base fee is **self-correcting**!

Calculate: How Much ETH Was Burned?

For each block:

$$\text{ETH burned}_t = \text{base\_fee}_t \times \text{gas\_used}_t \quad \text{(convert from Gwei to ETH: divide by } 10^9\text{)}$$

Sum over 100 blocks:

$$\text{Total ETH burned} = \sum_{t=1}^{100} \text{ETH burned}_t$$

Questions to answer

- How much more ETH is burned in the gas war scenario?
- At what block does the burn rate peak?
- How quickly does the base fee return to normal after the gas war ends?

Compare: First-Price Auction vs. EIP-1559

	First-Price Auction	**EIP-1559**
Fee predictability	Low (must guess)	High (base fee is visible)
During congestion	Extreme overpaying	Gradual, bounded increase
Fee volatility	Very high	Smoothed by adjustment
Fairness	Favors sophisticated users	Same base fee for all
Revenue to validators	100% of fees	Only the tip
Supply effect	None	Deflationary (burn)

EIP-1559 is not perfect — it doesn't solve MEV or prevent all fee spikes — but it makes fees *much* more predictable and fair.

Discussion Questions

1. **Should MEV be banned?**
   - Is it even *possible* to ban in a permissionless system?
   - What would be the unintended consequences?
2. **Would you use a CBDC?**
   - What if it offered you 2% interest directly from the central bank?
   - Would you trust the government with your transaction history?
3. **Stablecoins vs. CBDCs:** can they coexist, or will regulators eventually ban private stablecoins?

Summary & Day 4 Preview

Today's Key Takeaways

- PoW and PoS: different paths to the same goal (making cheating expensive)
- EIP-1559: auto-adjusting fees + ETH burn
- MEV: an unavoidable consequence of transparent mempools
- CBDCs: government digital money, very different from crypto
- Terra/Luna: why algorithmic stablecoins are fragile

Reading: [2], Ch. 5–7; [1]

Day 4 Preview: AI Meets Finance

- What is machine learning, really?
- Can AI predict crypto prices?
- Overfitting: the #1 trap
- Decision trees and random forests
- Hands-on: build a BTC predictor

References I

[1] Vitalik Buterin. *Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform*. 2014.

[2] Campbell R. Harvey, Ashwin Ramachandran, and Joey Santoro. *DeFi and the Future of Finance*. Wiley, 2021.

[3] International Monetary Fund. *CBDC: Progress and Further Considerations*. Tech. rep. International Monetary Fund, 2024.