

# A02: Block Building Simulation

## Build the Most Profitable Block

Prof. Joerg Osterrieder

(c) Joerg Osterrieder 2025-2026

Spring 2026

## Learning Objectives

- Understand how miners/validators select transactions for blocks
- Calculate fee density (fee per unit of block space)
- Experience the knapsack optimization problem firsthand
- Analyze trade-offs between transaction fee and block space

## Assignment Details

- Time: 40 min activity + 5 min presentation
- Format: Teams of 3–4 students
- Difficulty: Medium
- Points: 50
- Key constraint: Maximum block size = 10 units

## Core Challenge

You have 20 transactions but only 10 units of space. Maximize your total fees!

[digital-ai-finance.github.io/crypto-economics/assignments/A02\\_block\\_building/instructions.html](https://digital-ai-finance.github.io/crypto-economics/assignments/A02_block_building/instructions.html)

A02:

## Steps

- 1 Receive your set of 20 transaction cards
- 2 Each card shows: TX ID, Fee (ETH), Size (units)
- 3 Spread all cards on the table
- 4 Familiarize yourself with the range of fees and sizes
- 5 Assign roles: calculator, recorder, strategist

## Materials

- 20 transaction cards
- Block template worksheet
- Hash calculation guide

## Read Carefully

Each transaction has a different fee AND a different size. A high fee doesn't mean it's the best choice if it takes up too much space.

[digital-ai-finance.github.io/crypto-economics/assignments/A02\\_block\\_building/instructions.html](https://digital-ai-finance.github.io/crypto-economics/assignments/A02_block_building/instructions.html)

A02:

## Steps

- 1 For each transaction, calculate the fee-per-unit ratio:  $\text{Fee} \div \text{Size}$
- 2 Sort transactions by fee density (highest ratio first)
- 3 Select transactions greedily: add highest-density first
- 4 Track remaining block space (must not exceed 10 units)
- 5 Consider: can you swap a large TX for two smaller, more profitable ones?
- 6 Record your final block on the block template

## Fee Density Formula

- $\text{Fee Density} = \text{Fee} \div \text{Size}$
- Example:  $0.05 \text{ ETH} / 2 \text{ units} = 0.025 \text{ ETH/unit}$
- Higher density = more profitable per unit of space

## The Knapsack Problem

This is a classic optimization problem. The greedy approach (highest density first) is good but not always optimal. Can you do better?

## Steps

- 1 Compute your simplified block hash: sum the last digit of each included TX ID
- 2 Record the hash on your block template
- 3 Exchange blocks with another team for cross-verification
- 4 Verify: Are all transactions valid? Does total size  $\leq 10$ ? Is the hash correct?
- 5 Report any discrepancies

## Simplified Hash Example

- TX IDs: 103, 207, 415
- Last digits:  $3 + 7 + 5 = 15$
- Block hash = 15

## Why Hashing?

In real blockchains, the hash commits the block producer to a specific set of transactions. Changing even one transaction changes the hash entirely.

## Presentation Structure

- 1 Strategy explanation (2 min) — how did you select transactions?
- 2 Results (1 min) — total fees collected, space used, hash
- 3 Trade-offs (1 min) — what did you leave out and why?
- 4 Q&A (1 min)

## Key Concepts to Mention

- Fee density vs. absolute fee
- Opportunity cost of large transactions
- Greedy vs. optimal solutions
- Real-world miner/validator incentives

## Deliverables

- Completed block template with selected transactions
- Calculated hash
- Presentation of strategy and results

[digital-ai-finance.github.io/crypto-economics/assignments/A02\\_block\\_building/instructions.html](https://digital-ai-finance.github.io/crypto-economics/assignments/A02_block_building/instructions.html)

A02:

## Miner Incentives

- Miners maximize revenue by selecting high-fee transactions
- Block space is scarce — every unit has an opportunity cost
- Real Ethereum blocks: gas limit  $\approx$  30M gas
- MEV (Maximal Extractable Value) adds another dimension

## The Knapsack Problem

- NP-hard optimization problem in computer science
- Items have weight (size) and value (fee)
- Greedy by density is a good approximation
- Exact solutions require dynamic programming

## Real-World Implications

- Users compete for block space by raising fees
- EIP-1559 changed Ethereum's fee market (base fee + priority fee)
- During congestion, low-fee transactions wait in the mempool

## Assignment Page

[digital-ai-finance.github.io/crypto-economics/assignments/A02\\_block\\_building/instructions.html](https://digital-ai-finance.github.io/crypto-economics/assignments/A02_block_building/instructions.html)

## All Assignments

[digital-ai-finance.github.io/crypto-economics/assignments/index.html](https://digital-ai-finance.github.io/crypto-economics/assignments/index.html)

**Grab your transaction cards and start building your block!**

You have 40 minutes for the simulation plus 5 minutes for your team presentation.

---

(c)

Joerg Osterrieder 2025-2026