Economics of Blockchains

51% attacks, transaction fees, FruitChains

Zeta Avarikioti
georgia.avarikioti@tuwien.ac.at

December 17th, 2025

TU WIEN

S&P Security & Privacy

Overview

- Economic limits of blockchains
  - 51% attack
  - Why Bitcoin works?
- Mechanism design in blockchains
  - Transaction fees & EIP1559
  - FruitChains

Economic Limits of Blockchains

Why run a node?

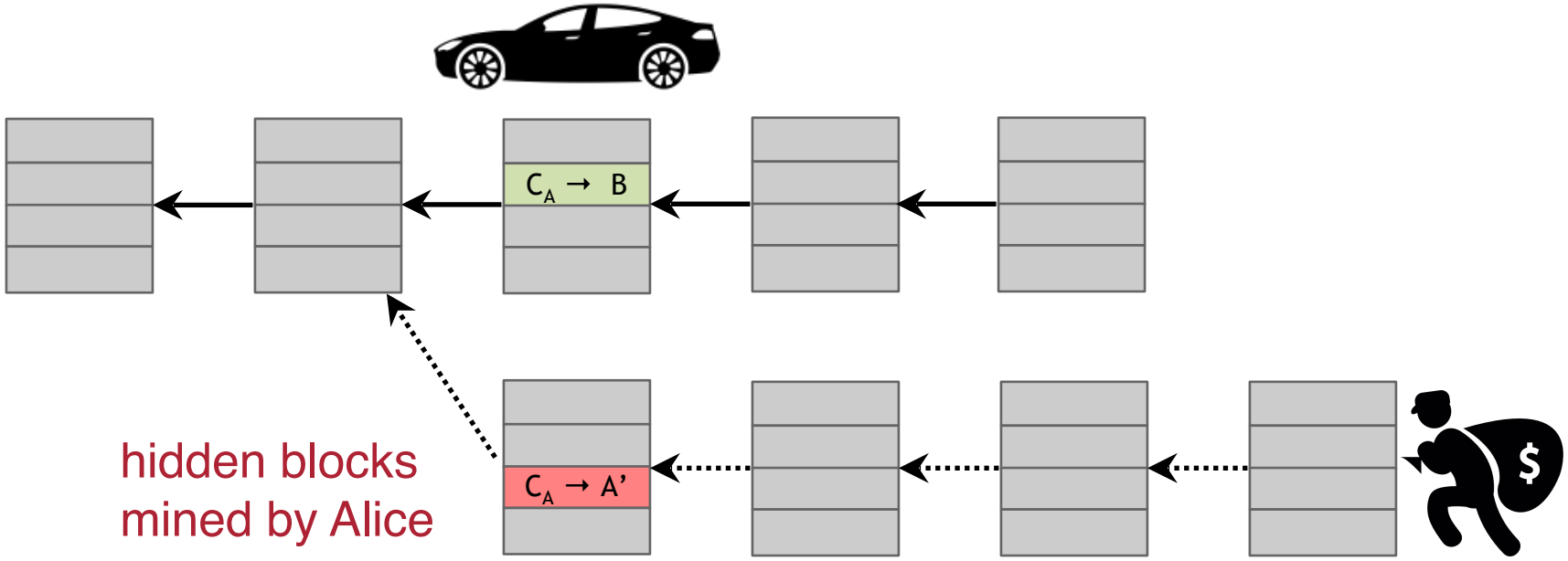
- Block rewards!

- Rational nodes aim to maximize their rewards
  - Difficulty adjustment imposes fixed chain growth
  - Hence, fixed rate of block rewards
  - Maximizing rewards by maximizing the fraction of blocks on the longest chain (chain quality)

Economic security of blockchains

The security bounds hold *only if*
we can financially motivate them!

51% attack



- c = cost of one hash (or coin) per time step
- n = # of honest hashes (per time step)
  - total honest investment = nc
  - Alice must invest *bnc* to reach a *b/(1+b)* fraction of hashrate
- R = rewards per time step in USD
  - rewards are given proportionally to hashrate (R/n)

Economic cost of 51% attack

- Equilibrium before the attack:

  Marginal cost of 1 hash = expected marginal reward of 1 hash

  c = R/n

- Mining is not a profitable!? Then why do miners mine?
  - Bullish, storage of value, etc.
- Cheaper electricity → c drops…what about the price?
- Attack cost for T time steps = bcnT = bRT

= **22.5M $** (b=1.5, T=60, R=3.125)

Economic cost of 51% attack

- Rewards
  - Alice gets RT rewards over T time steps
  - Net cost = (b-1)RT
- All transaction fees (assume 0)

= **7.5M $** (b=1.5, T=60, R=3.125)

Economic cost of 51% attack

- Hashrate went up!
- Alice produces b blocks (before difficulty adjustment)
  - Net cost = (b-b)RT = 0

Why no 51% attack?

= 0 $!!!

Why Bitcoin works?

- Market governance
- Miner commitment
- Suspending consensus

Market governance

- Price drop: D → new price = (1-D)* old price
- Rewards ($) = bRT(1-D) + V
- Cost ($) = (1+f(b))bRT

51% attack not profitable in Bitcoin iff

bRTD > V

Market governance

- D: “pick your poison” parameter!
  - D small
    - double-spending is easier
    - high fees to secure V
    - Bitcoin is either currency or a commodity
  - D large…great, no double-spend attacks!
    - sabotage attacks (Bitcoin worth is $1.8 trillion!)

51% attack not profitable in Bitcoin iff

bRTD > V

Miner commitment

- ASICs & Power Purchasing Agreements
  - non-repurposable asset & high cost
  - 50% upfront payment by miners
  - 24 months asset depreciation
    - 12 months' cost is 164,250 BTC ($13B = c)
    - **Anything that jeopardizes the value of these coins before delivering is highly destructive!**

51% attack not profitable in Bitcoin iff

$bRTD + Dc_A > V$

Miner commitment

- Example
  - 60% hashrate (b=1.5)
  - 20 blocks confirmation
  - 5% price drop

Non-repurposable expenses!

$$V < (1.5 \cdot 3.125 \cdot 20 \cdot 0.05 + 0.05 \cdot 0.6 \cdot 164{,}250) \cdot 80{,}000$$
$$\approx (5 + 4{,}930) \cdot 80{,}000 \approx 400M\$$$

51% attack not profitable in Bitcoin iff

bRTD + Dc_A > V

Suspending consensus

- **Users lead and miners follow!**
- Example: UASF movement 2017 (SegWit)
- **f** = probability users coordinate off-chain to *suspend consensus* (e.g. DAO attack on ETH)
  - Decreases potential reward
  - Increases potential V
- Double-edge sword: good for sabotage, bad for upgrades

51% attack not profitable in Bitcoin iff

$bRT(1-f(1-D)) + Dc_A > V$

Key takeaways

- Security mainly due to non-repurposability of expenses
- If the price is very robust, the commitment must be larger
- Block rewards are useful → secondary market needed
- Confirmation time adds little to security → months
- Block confirmation may help in detecting attacks (slower block discovery)
- Derivative markets, smart contracts (DeFi), payment channels, rollups can be harmful (increase V)
- Social intervention may be useful against attacks (but slows down progress)

Mechanism Design

In Blockchains

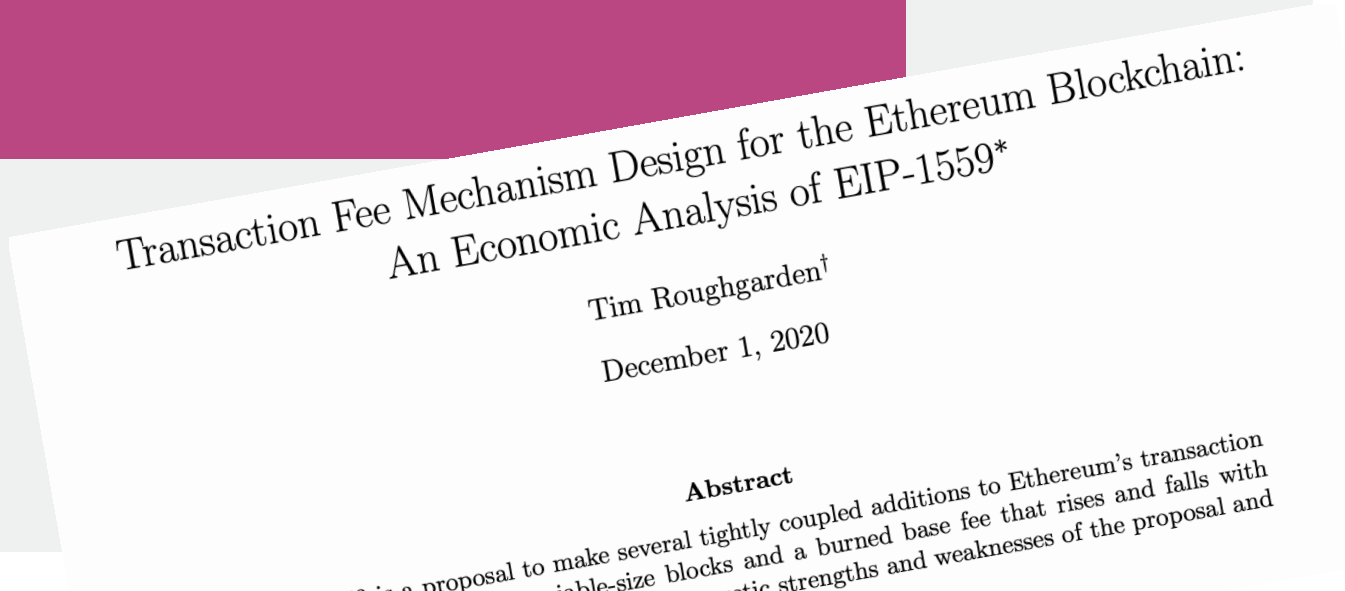
What is game theory?

- Game theory
  - given a game determine the equilibrium outcomes
  - Prisoner's dilemma, tragedy of the commons, etc.
- Mechanism design ("inverse game theory")
  - Given an intended outcome, design a game so that it becomes an equilibrium
  - E.g., use rewards, fines, etc.
- Technology transfer from mechanism design to blockchain protocol design?

Incentive-compatible blockchains

- Block ordering
  - Transaction fees
  - Block rewards (myopic, non-myopic analysis)
- Transaction ordering — we saw it in Lecture 7!
  - Maximal Extractable Value (MEV)

Transaction Fees



Transaction Fee Mechanism Design for the Ethereum Blockchain:
An Economic Analysis of EIP-1559*

Tim Roughgarden†

December 1, 2020

Abstract

proposal to make several tightly coupled additions to Ethereum's transaction
iable-size blocks and a burned base fee that rises and falls with
tic strengths and weaknesses of the proposal and

Why do we have transaction fees?

- Scarce resource (demand >> supply)
  - ETH Supply: block size=12.5M gas (2021)
    - Gas = measure of computational costs of tx
    - ~ 600 simple transfers
    - ETH block generation time ~ 12 sec
  - ETH demand: much more (Think DeFi!)

Goal of tx fees is **economic efficiency**:

differentiate high- vs. low-value tx so the block is packed with the highest value txs.

- **Remark:** scarcity & economic efficiency = tx revenue (even if you don't care about it)

Easy vs. difficult price estimation

- Typical tx fees: like buying a house (first-price auction)
  - Users propose their own fees
- New proposal in ETH (EIP 1559): make it as easy as Amazon (posted-price)
  - Acceptable fee typically set in advanced by the protocol
  - Better user experience

Key idea #1: The base fee

- Each block has a *base fee*, only determined by the previous blocks.
  - Acts as reserve price (if tx fee < base fee → ineligible for inclusion)
  - Users and miners cannot manipulate the base fee (like in Vickrey)

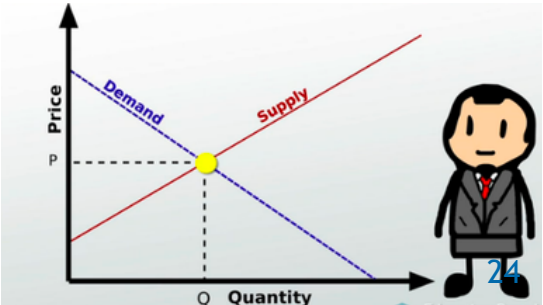
- Who gets the base fee revenues?
  - Base fee *cannot* be given to the block's miner
    - Miners and users can collude off-chain to evade the base fee (equivalent to being 0)
    - Example: 50% burned and 50% to miner, then user gives off-chain 75% if miners access its tx with 0 fee and they both gain 25%!
    - The base fee must be orthogonal to miner utility

No one - *burn* the base fee! (EIP 1559)

  - Pay it forward to miners of future blocks (alternative)

Key idea #1: The base fee

- How is the base fee set?
  - Goal: base fee = “market clearing price”
    - Price when demand = supply
    - Supply = # tx fitting a block
      (e.g., 1MB in BTC, 12.5M gas in ETH)
    - Demand = # tx willing to pay this price to be included
  - Demand is changing: Tatonnement
    - ↑ fee when demand >> supply
    - ↓fee when supply >> demand
  - But demand (excluded txs) is not included on chain!!!



Key idea #2: Variable size blocks

- Have **target** block size to control the average
  - 15M gas for ETH
- **Maximum** block size is double
  - 30M gas ETH

- **Tatonnement:** ↑ base fee after larger than target block size, ↓base fee after smaller than target block size
  - EIP 1559: max increase/decrease ±12.5% (linearly interpolate)

- **Benefits:** can borrow capacity from near future to
  - Reduce variability in market price
  - Absorb short demand spikes
  - Reduce confirmation delays

Key idea #3: Tips

- **Problems**
  - Why not mine empty blocks since the base fee is burned?
  - Excessively low base fees & high spikes → demand > target → how will the miner decide which tx to include?

- **Tips**
  - Users decide, transferred to the miner
  - First price auction on top of base fee!
  - Users also specify a fee cap (=max willingness to pay)
    - Final price = min{base fee + tip, fee cap}

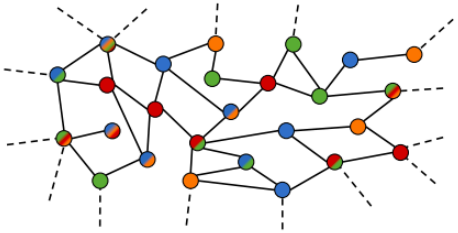
- **(Hopefully) Benefits**
  - Base fee not very low → all tx willing to pay fit in the block → **no competition** → small tip is enough!
  - Base fee very low → first price auction

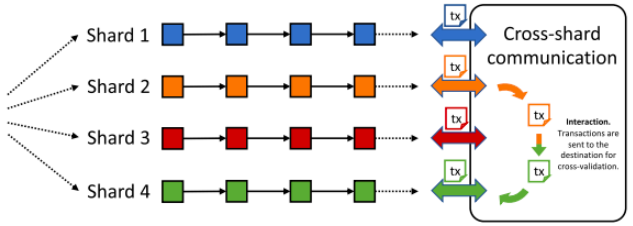
Guarantees

- Miners incentivized to follow the protocol
  - cf. Second price auctions fail miserably - trusted third party necessary, else miners manipulate auction with fake txs
- No incentive for miners and users to collude off-chain
  - cf. Base fee revenues to miners
  - cf. Burn some fees from first price auction (no base fee)
- Easy fee estimation (ex post NE)
  - If base fee not excessively low
  - If users are rational

Will tx fees be lower?

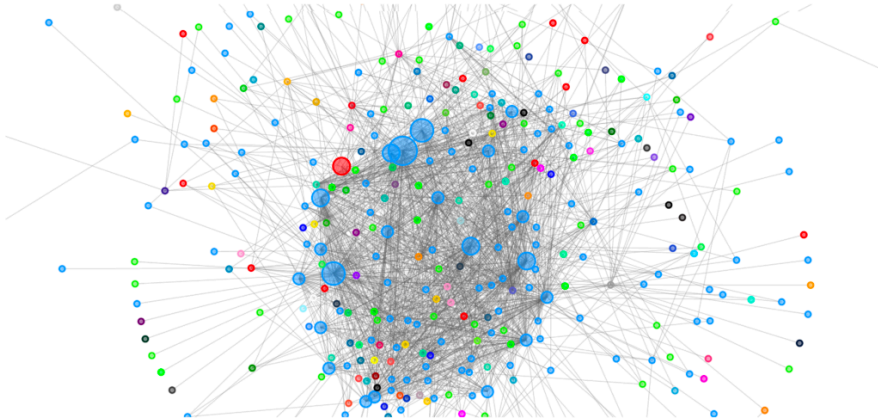
No ☹ That is a scalability problem, not a mechanism design problem!





Increase supply (e.g., sharding)

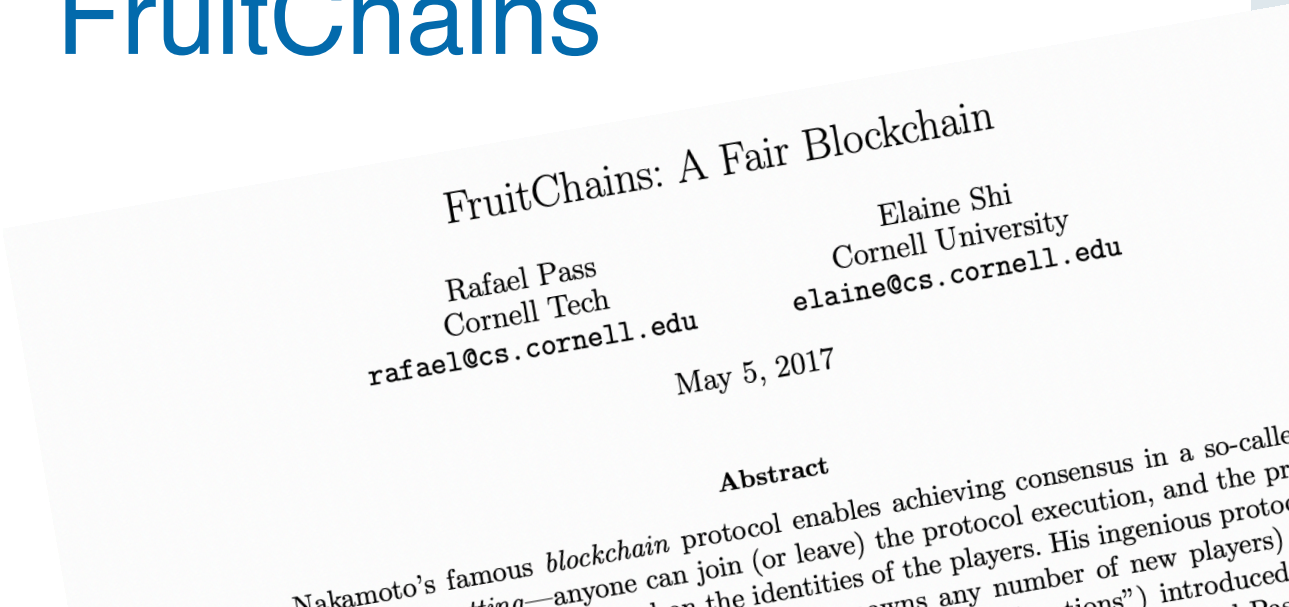
Decrease demand (e.g., L2 solutions)



Open directions

- **Security:** Miner may manipulate protocol over longer time scales
  - Good news: Similar incentives to now, e.g., selfish mining attack
  - Bad news: Miners may have more to gain by colluding due to the burning base fee
- **Base fee:** Pay base fee forward instead of burning it
  - Favors miners over stakeholders
  - Favors constant inflation and variable security
- **Tips:** Use hard-coded tips
  - Good news: Better UX
  - Bad news: When base fee very low, expected tip market will arise off-chain

FruitChains



FruitChains: A Fair Blockchain

Rafael Pass
Cornell Tech
rafael@cs.cornell.edu

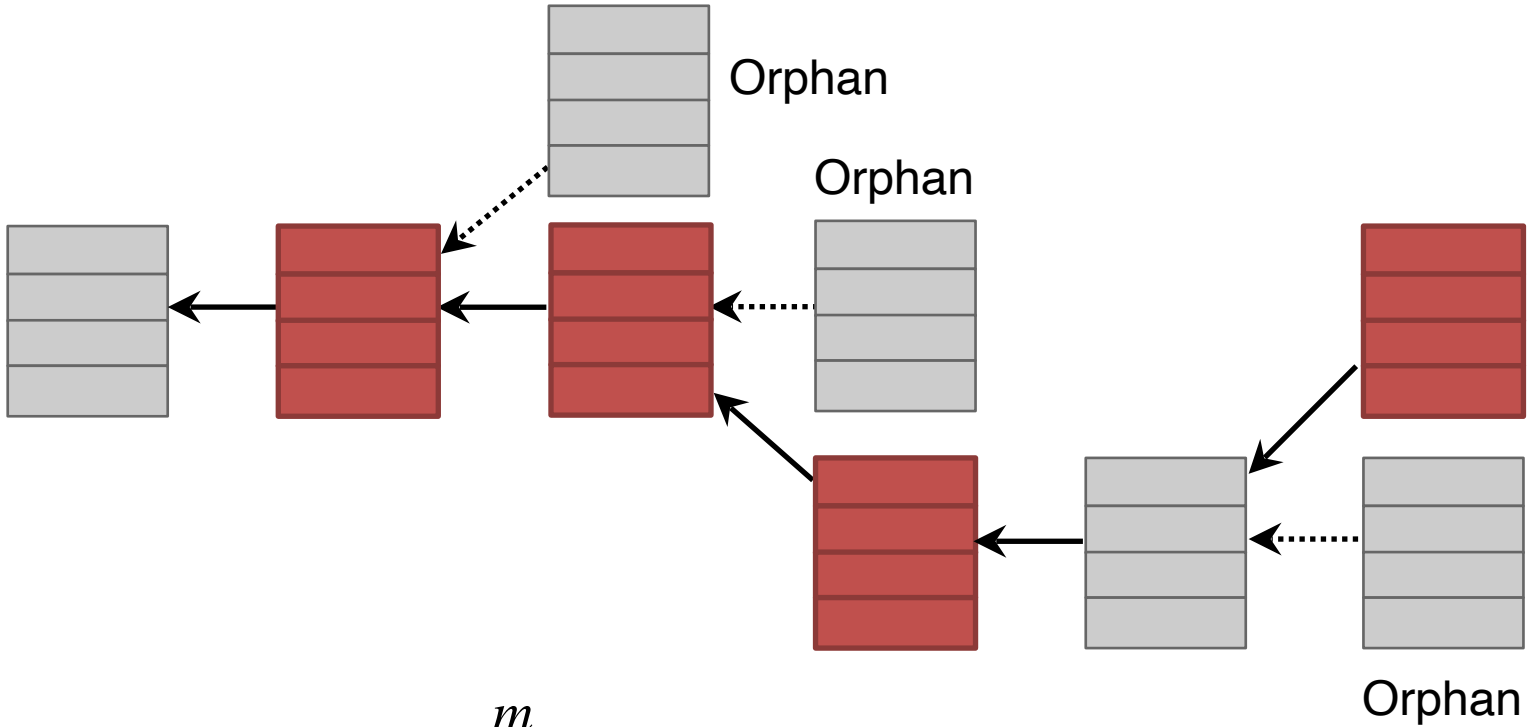
Elaine Shi
Cornell University
elaine@cs.cornell.edu

May 5, 2017

Abstract

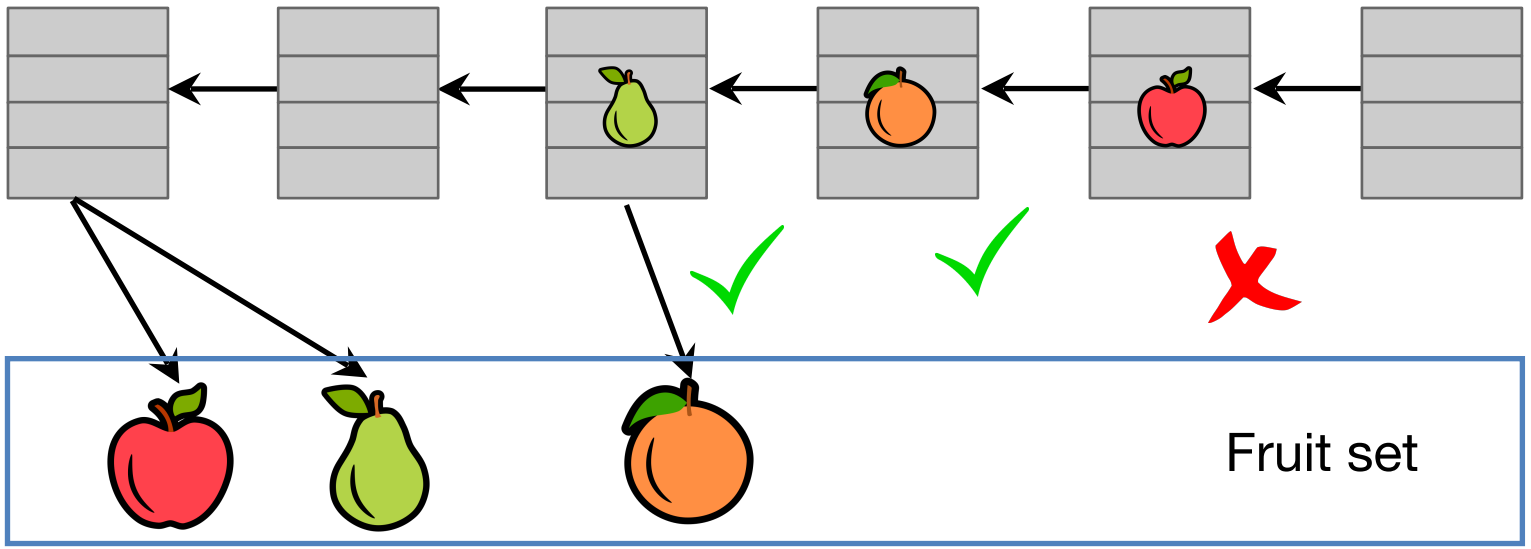
Nakamoto's famous *blockchain* protocol enables achieving consensus in a so-calle
setting—anyone can join (or leave) the protocol execution, and the pr
on the identities of the players. His ingenious protoc
grows any number of new players)
tions") introduced

Selfish mining (revisited)



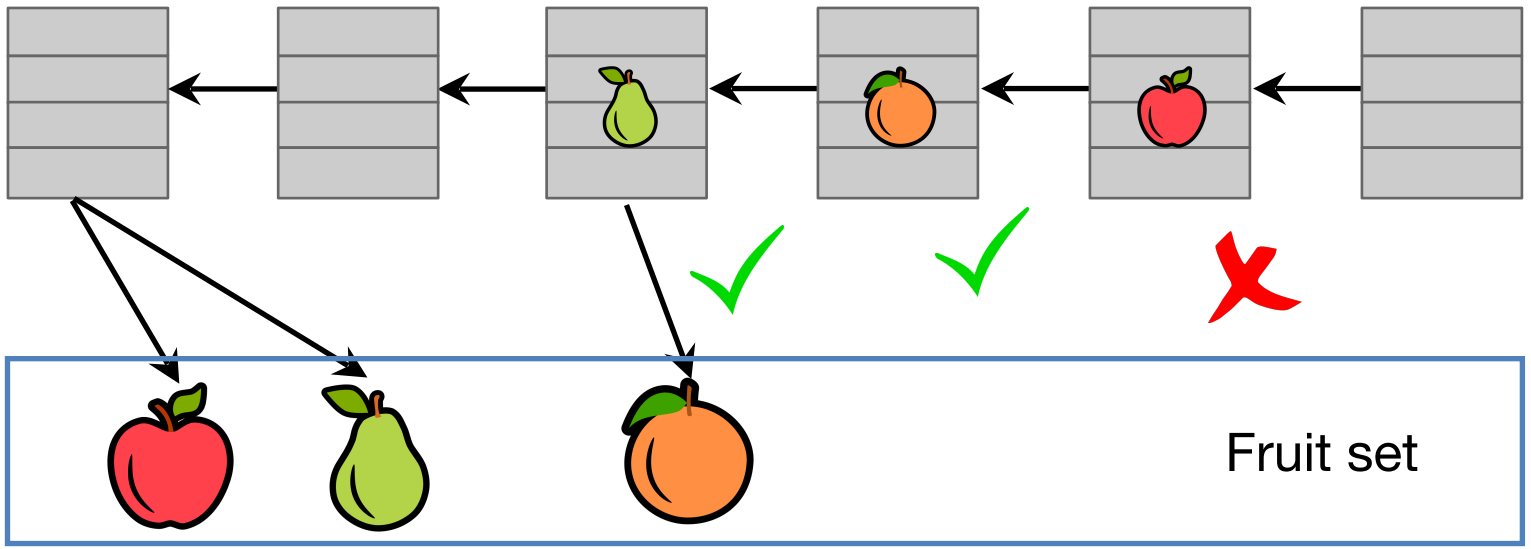
- Chain quality $CQ \geq 1 - \frac{m_a}{m_h}$, m_a, m_h are the adversarial & honest mining rates
- Can we design a protocol that is fair, i.e., $CQ \approx \frac{m_h}{m_h + m_a}$?
- Use the idea of on-chain simulation of block reward allocation in mining pools

FruitChains



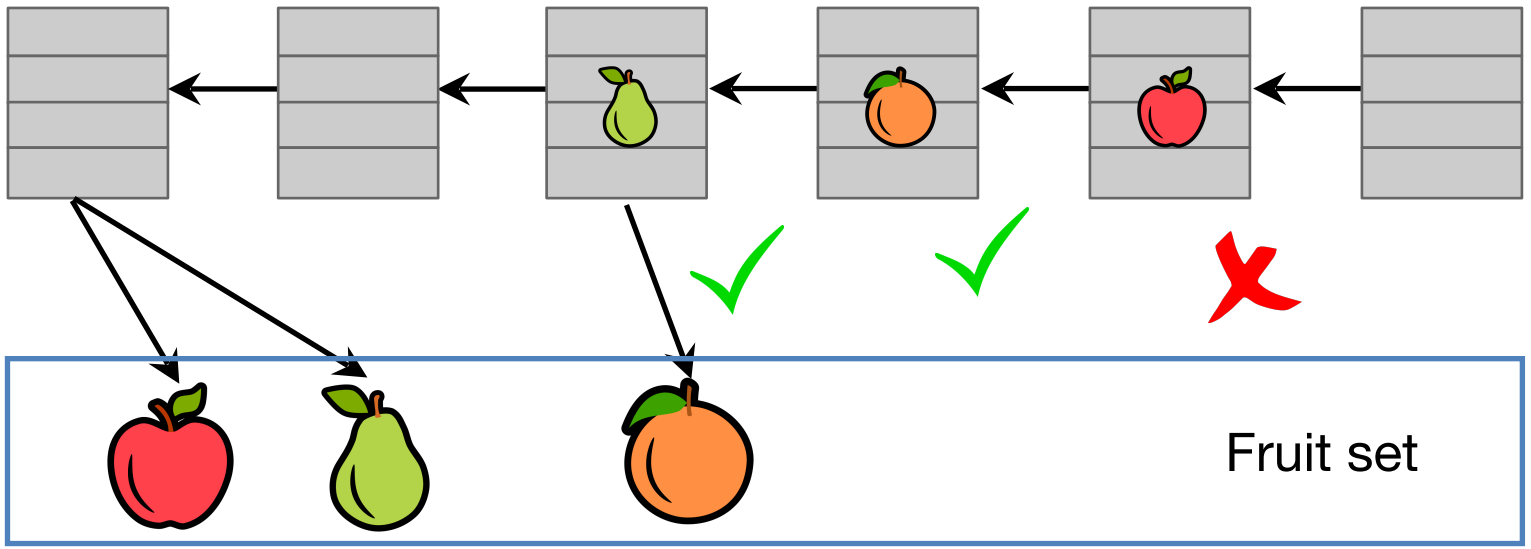
- Protocol key ideas:
  - Miners mine blocks and fruits (2-in-1)
  - Fruits are mined with lower difficulty (like the partial solutions in mining pools)
  - Fruits contain txs and “hang” from blocks
  - Blocks contain only recent fruits (not txs)
  - Ledger: Extract distinct fruits (only first occurrence counts); then order fruits by first block inclusion; finally extract the tx order

FruitChains



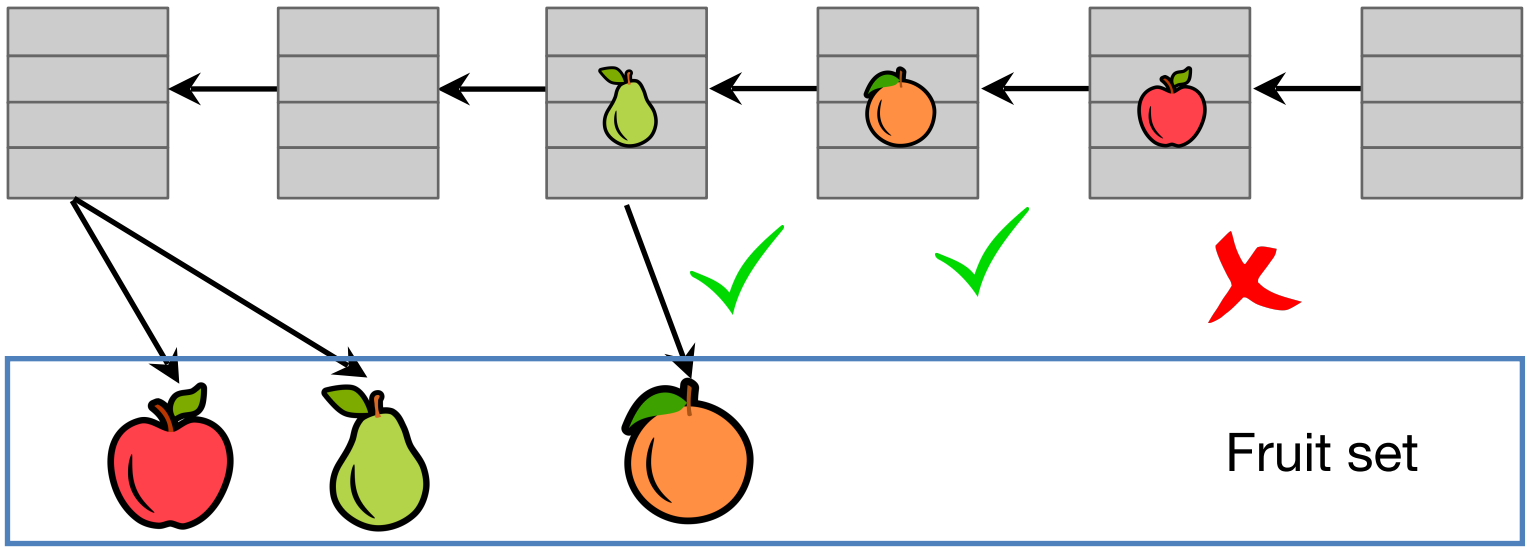
- Why does it work?
  - Chain quality is measured in fruits (not blocks)
  - An honest leader will include all recent fruits in their buffer
  - There will be an honest leader elected soon (while honest fruits are still recent)
  - So the adversary cannot gain by excluding honest fruits
  - We only include recent fruits to avoid a different attack:
    - The adversary may withhold fruits
    - Then suddenly release many of adversarial fruits and tamper with CQ

FruitChains



- From fairness to incentive compatibility
  - The fruit rewards and tx fees are evenly distributed to the T(k) preceding leaders
  - For collusions < 1/2 of the computational power, if the protocol is approx. fair then it is approx. Nash equilibrium.

FruitChains



- What problems does it solve?
  - Selfish-mining attack & unfair allocation of rewards
  - Centralization of mining pools
  - Transaction fees exacerbate instability
    - Miners may fork txs with high fees

Economics of Blockchains

51% attacks, transaction fees, FruitChains

Questions?

December 17th, 2025

TU WIEN

S&P Security & Privacy