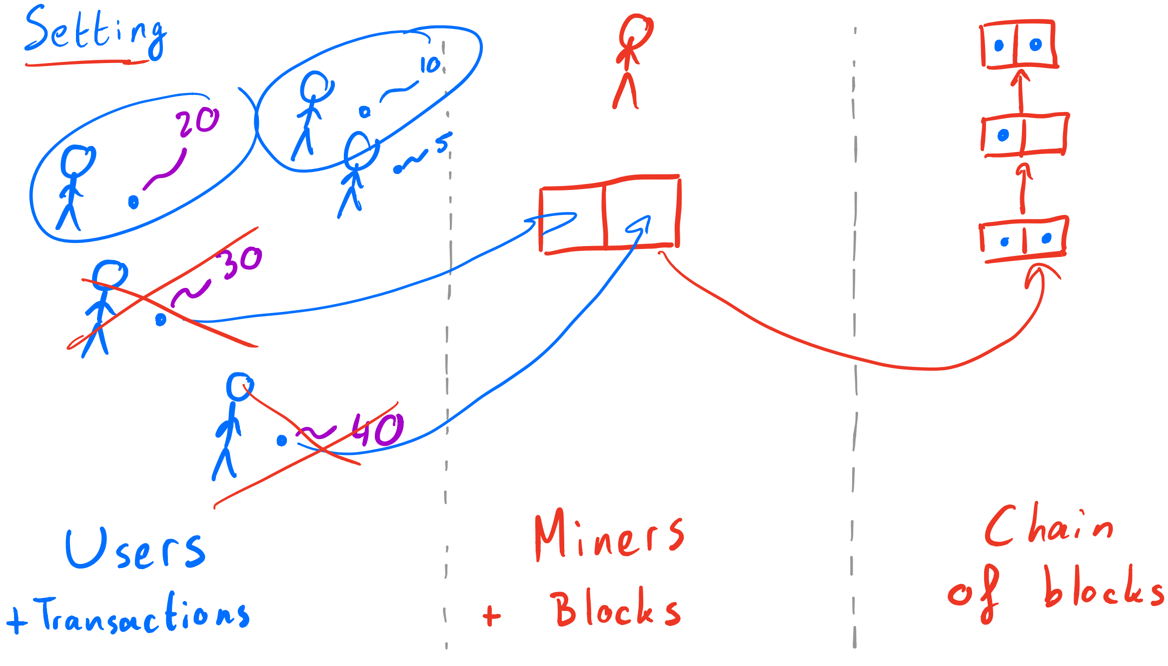
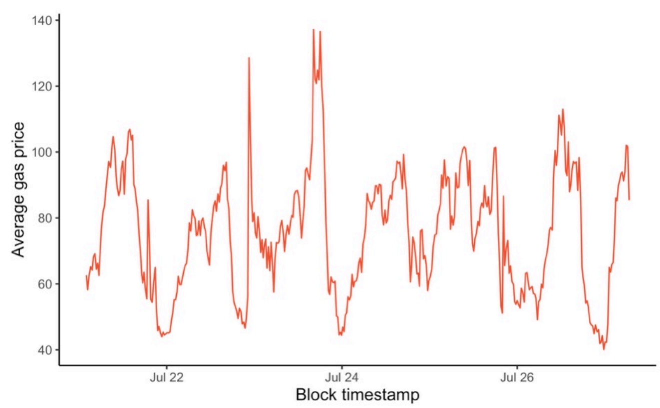
Ethereum's new transaction fee market design

EIP 1559

Setting



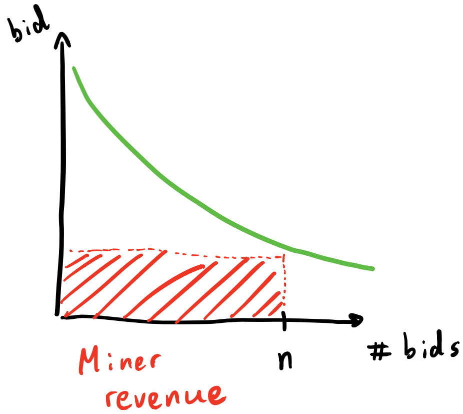
- → Transaction inclusion is a first-price auction.
- → User bids $ ⟷ Miner receives $
- → FPA ⟷ inefficient!



Why 2nd price auctions fail

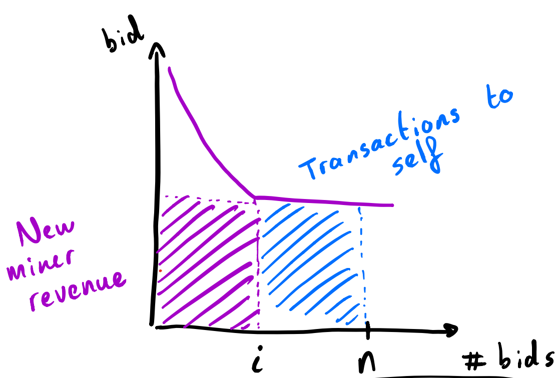
→ Bids in a block are $f_1, \ldots, f_n$

→ Miner receives $n \times f_n$



Problem!

→ Miner can include transactions to itself and increase n-th bid!



Given bids

$f_1 \geq f_2 \geq \ldots \geq f_n$

Miner optimises

$\max_{f_i} \; i \times f_i$

\$1,000,000	→	\$1,000,000
\$ 0.10		\$1,000,000
\$ 2 × 0.10		\$1,000,000

→ Waste of space!

+ not incentive compatible

The key problem

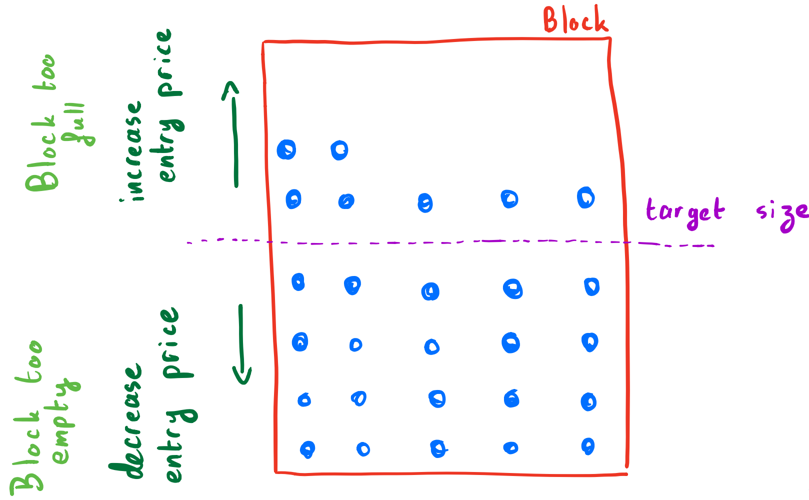
> As long as miners decide the entry price, they can game it

→ Solution : make the entry price "objective".

Dynamic congestion pricing

- Same idea as ERP
- Observe congestion
- When congestion high, raise entry price
- When congestion low, lower entry price

Congestion observability



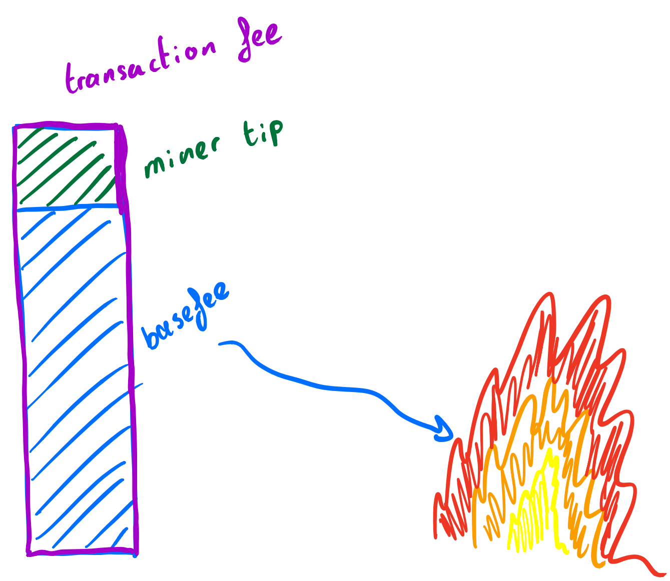
Entry price = "basefee"

What users pay

Users must pay at least basefee.

Can add a tip to the miner

Basefee is burnt!



Comparing both paradigms

FPA

- Users observe historical prices (+ current mempool state) and decide on f
- Inclusion depends on other bidders directly, unpredictable

Bad UX!

EIP 1559

- Users observe basefee
- Tip can be set as small compensation
- Most likely included in the next block. Good UX!
  (except during spikes!)

Simulations

Stationary case

→ Demand is a Poisson process, $\mathbb{E} = \lambda$

→ Users have a value $v \sim F_v$
cost for waiting $c \sim F_c$

$$v - 5c - f_{\uparrow}$$

basefee + tip

→ See notebook

Strategic users

During demand shifts, users want to get ahead.

Enter a bidding war

→ See notebook

Questions

→ Efficiency of the mechanism
vs. FPA, another 2nd price proposal, "escalator"

→ Miner collusion
Artificially keep basefee low to obtain larger tips

→ Equilibrium behaviour of the users
how to set tip / max fee

→ Approximations (wallet defaults)

→ Miner transaction pool behaviour