האוניברסיטה העברית בירושלים
THE HEBREW UNIVERSITY OF JERUSALEM

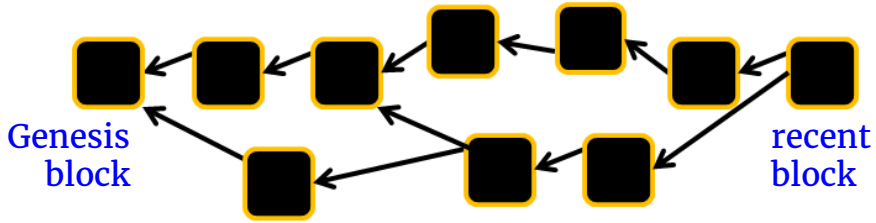
DAGlabs
A novel BlockDAG platform

Incentives and Trade-offs in Transaction Selection in DAG-based Protocols

Yonatan Sompolinsky and Yoad Lewenberg
Scaling Bitcoin, Stanford

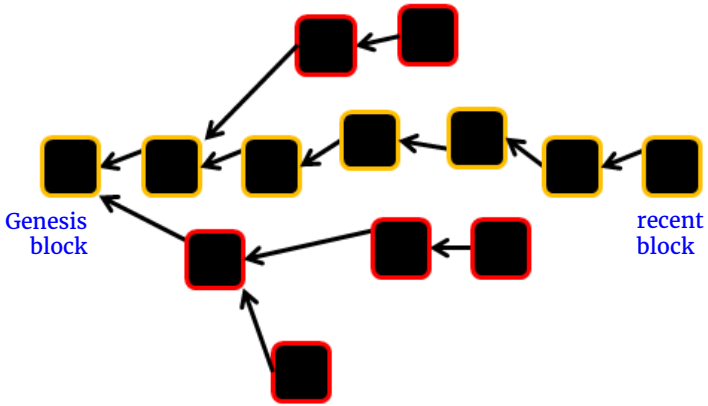
Background

- Directed Acyclic Graph of blocks (blockDAG)
- Inclusive Blockchain Protocols, Financial Crypto ‘15, Lewenberg, Sompolinsky, and Zohar
- Modification and scaling up of Layer 1
- Orthogonal to Layer 2 solutions

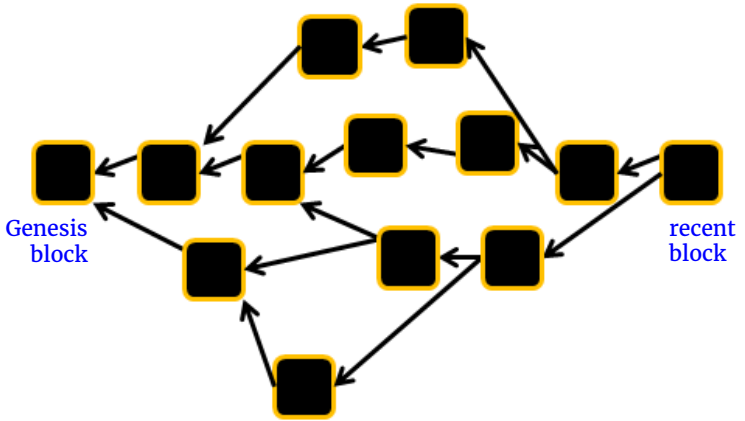


Blockchain vs BlockDAG

Chain paradigm:
1. maintain single chain
2. ignore the rest
3. forks rare



DAG paradigm:
1. maintain entire graph
2. consider all blocks
3. forks common

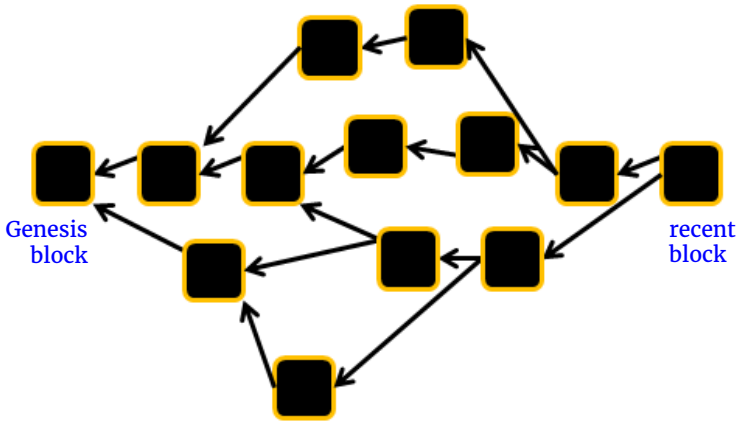


The BlockDAG paradigm

more information **possibly** implies:

- more security
- more scalability
- more fairness

DAG paradigm:
1. maintain entire graph
2. consider all blocks
3. forks common

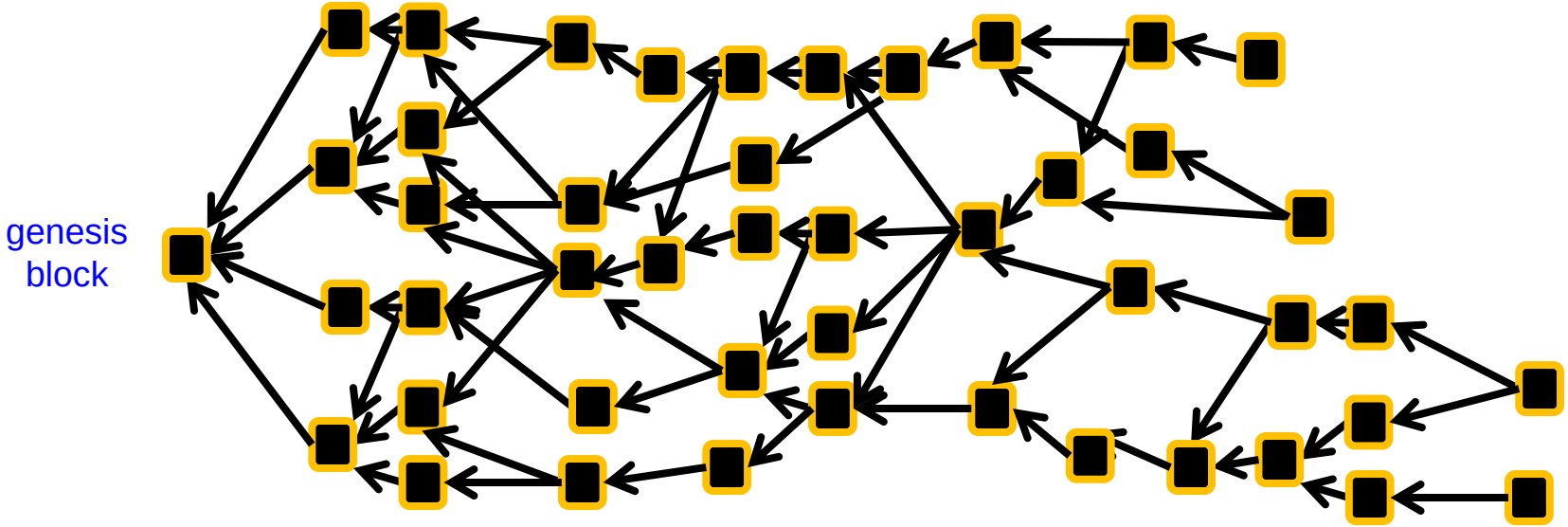


Road to scaling up Layer 1

speed up block rate (or size)

reference all blocks

extract consistency



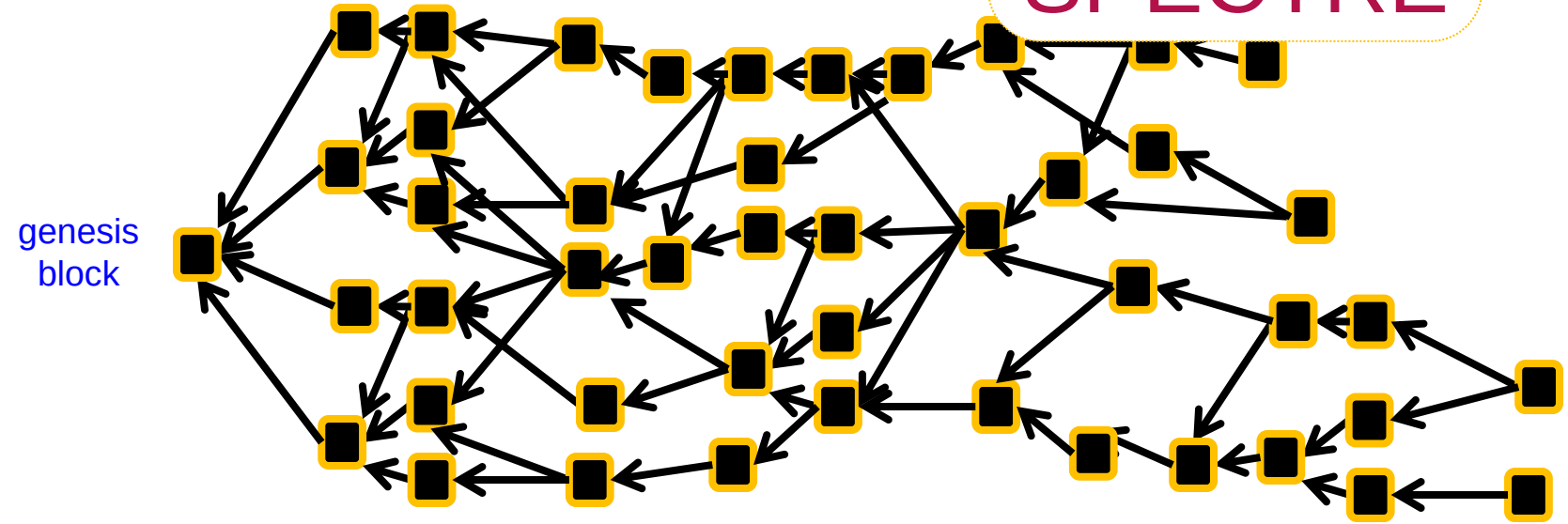
Road to scaling up Layer 1

speed up block rate (or size)

reference all blocks

extract consistency

SPECTRE



BlockDAG is (only) a paradigm

- DAG merely a framework, not a solution
- Not all blockDAGs are created equal
- DAG vs chain like highway vs one-lane road…

BlockDAG is (only) a paradigm

- DAG merely a framework, not a solution
- Not all blockDAGs are created equal
- DAG vs chain like highway vs one-lane road…

BlockDAG is (only) a paradigm

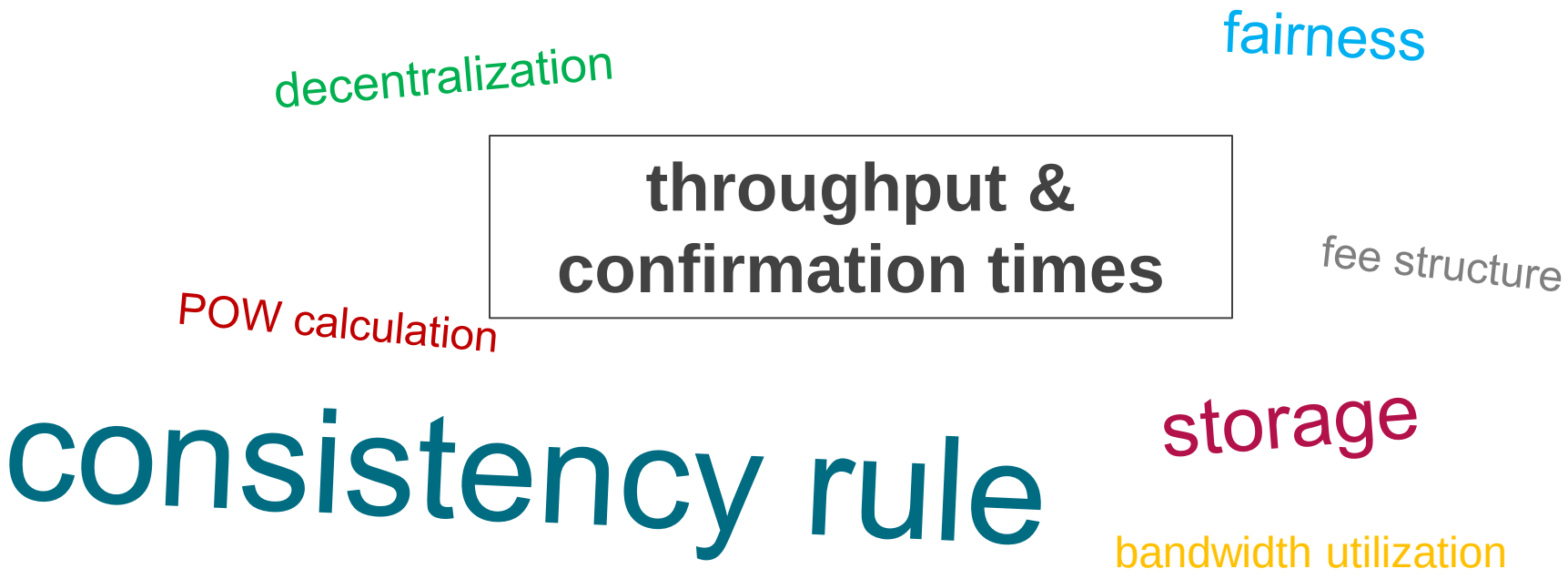
- DAG merely a framework, not a solution
- Not all blockDAGs are created equal
- DAG vs chain like highway vs one-lane road…

BlockDAG is (only) a paradigm

- DAG merely a framework, not a solution
- Not all blockDAGs are created equal
- DAG vs chain like highway vs one-lane road…

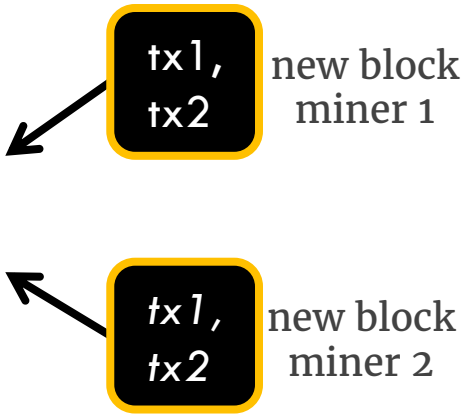
Scaling up Layer 1 -- challenges

– – –



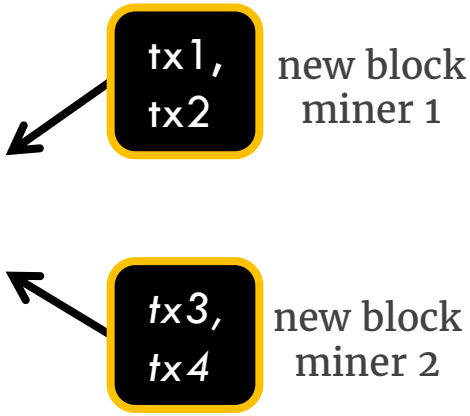
Two scenarios for DAG throughput

mempool: tx1>tx2>tx3>tx4>... (non-conflicting)



tx1,tx2 selected & approved
tx3,tx4 still in mempool

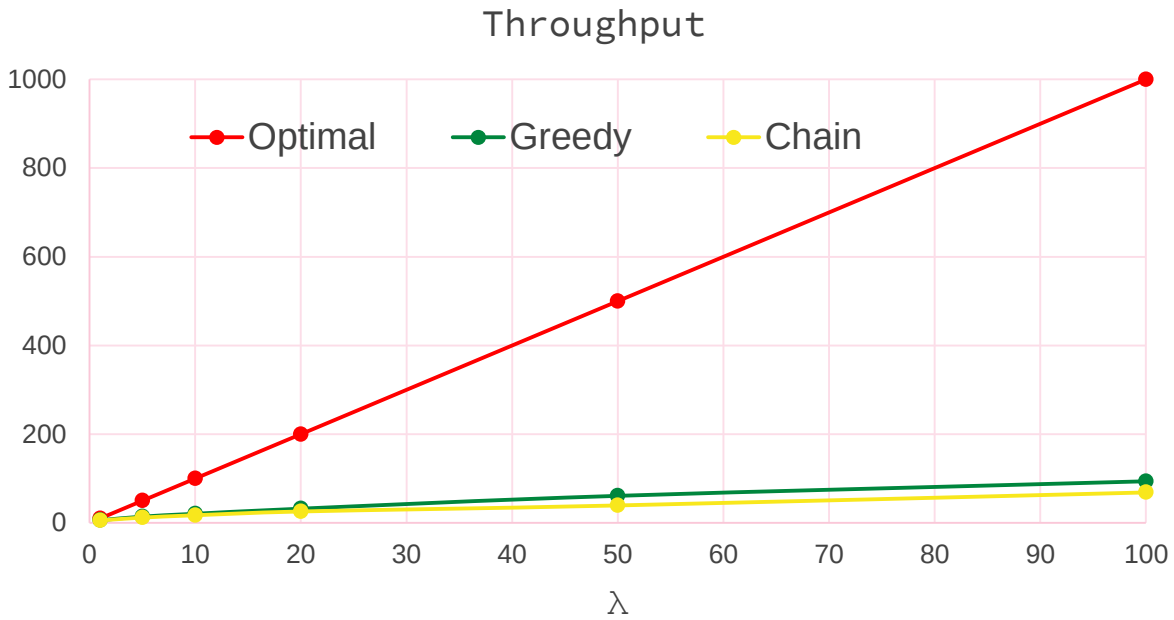
under utilization



tx1,tx2,tx3,tx4 selected & approved
(mempool cleared faster)

full utilization

Proposition 1: under naïve/greedy mining, DAG throughput ≈ chain throughput



Key observation / good news

Miners are incentivized to be avoid selecting the same txns, and to contribute to throughput increase.

Indeed, “collisions” result in loss of fees…

The Inclusive Game

mempool: tx1>tx2
players: miner1, miner2

	miner2 chose tx1	**miner2 chose tx2**
miner1 chose tx1	(0.5*tx1, 0.5*tx1)	(tx1, tx2)
miner1 chose tx2	(tx2, tx1)	(0.5*tx2, 0.5*tx2)

collision on tx1

collision on tx2

The Inclusive Game

pure strategy: select a txn

mixed strategy: select a txn using randomness

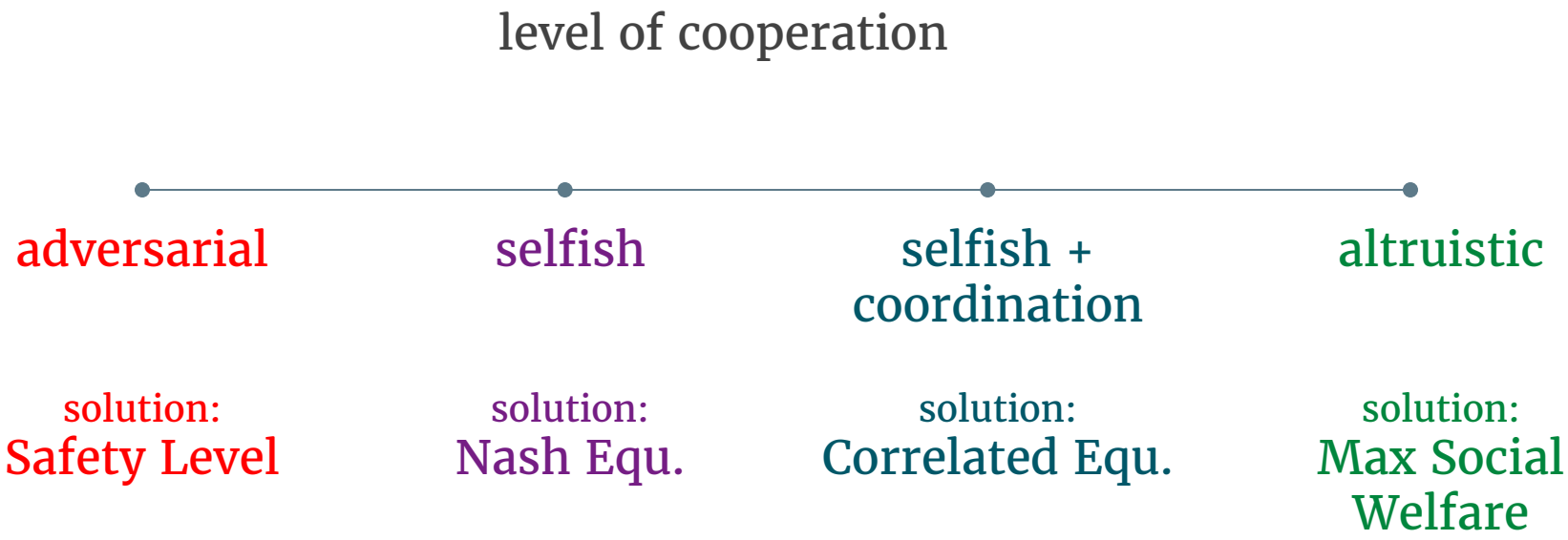
	miner2 chose tx1	**miner2 chose tx2**
miner1 chose tx1	(0.5*tx1, 0.5*tx1)	(tx1, tx2)
miner1 chose tx2	(tx2, tx1)	(0.5*tx2, 0.5*tx2)

collision on tx1

collision on tx2

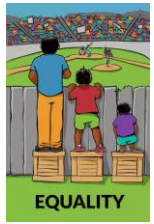
How to “solve” the game

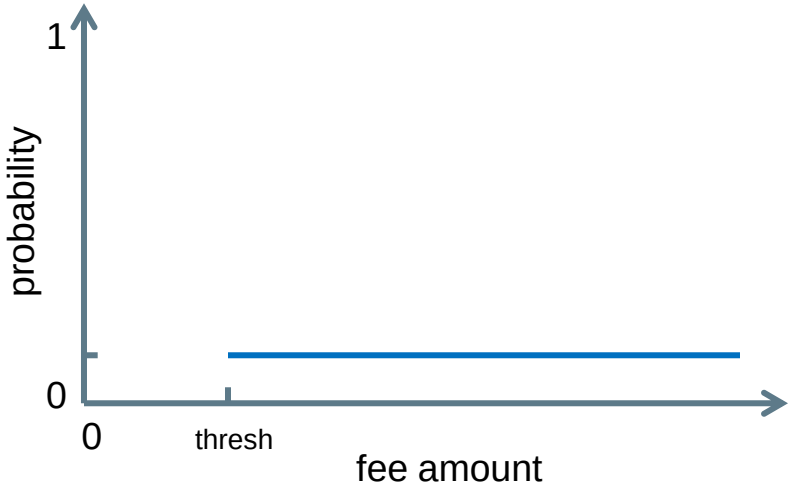
– – –



Max Social Welfare

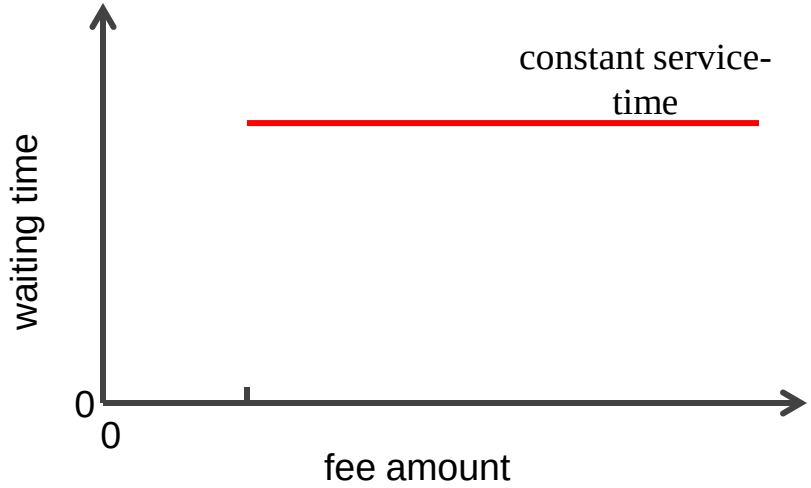
- Solution: select txns uniformly [above capacity threshold]
- No collisions, full utilization
- But there's a catch…





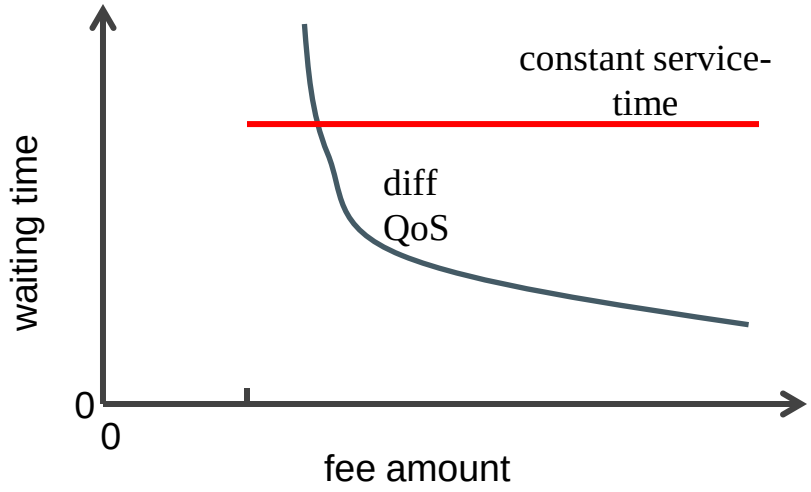
High throughput is not enough

1. Strategically unstable
2. Forces egalitarian waiting times, no QoS levels and preferential treatment



High throughput is not enough

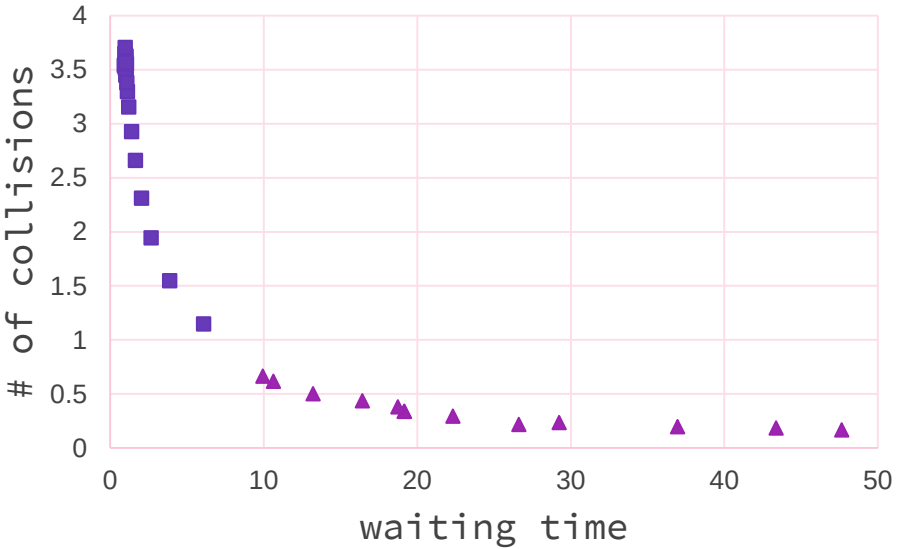
1. Strategically unstable
2. Forces egalitarian waiting times, no QoS levels and preferential treatment

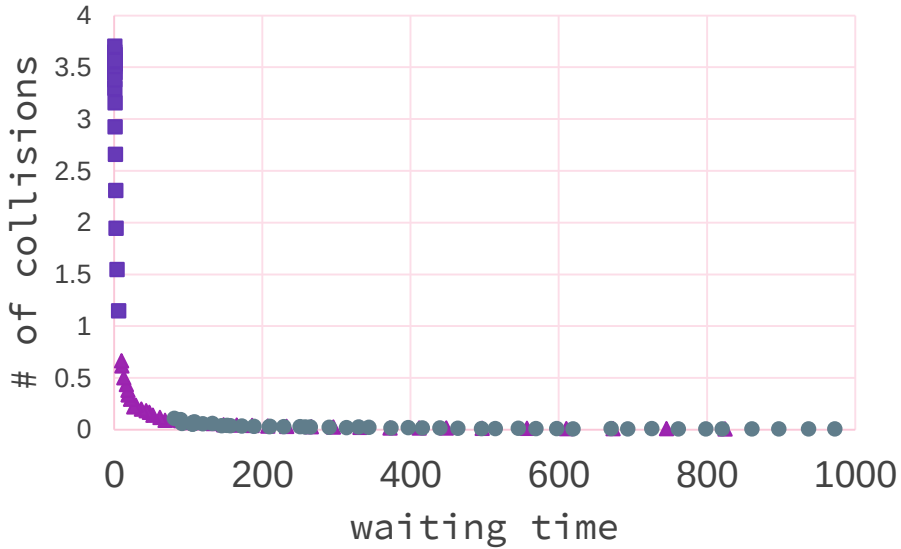


Trade-off: high utilization vs fast conf. times

---

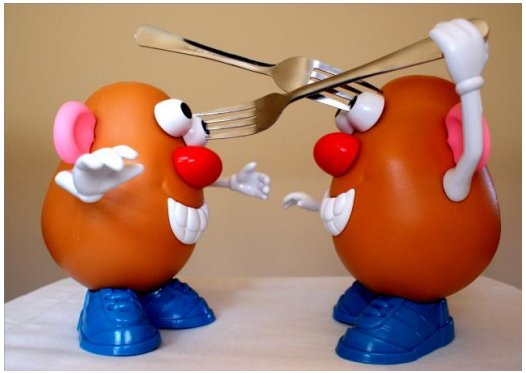
shorter waiting times → more collisions → lower utilization





Nash Equilibrium

- Finding Nash usually hard
- Tit-for-tat strategies
- Greedy pools/miners will suffer retaliation (?)



round #1

miner1\miner2	tx1	tx2
tx1	(0.5*tx1, 0.5*tx1)	(tx1, tx2)
tx2	(tx2, tx1)	(0.5*tx2, 0.5*tx2)

round #2

miner1\miner2	tx1	tx2
tx1	(0.5*tx1, 0.5*tx1)	(tx1, tx2)
tx2	(tx2, tx1)	(0.5*tx2, 0.5*tx2)

round #3

miner1\miner2	tx1	tx2
tx1	(0.5*tx1, 0.5*tx1)	(tx1, tx2)
tx2	(tx2, tx1)	(0.5*tx2, 0.5*tx2)

Nash Equilibrium (myopic)

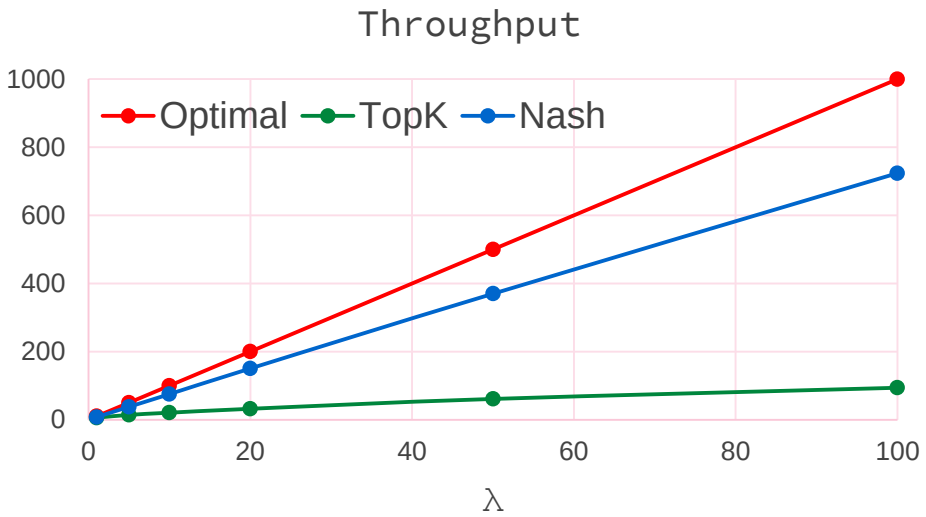
- Assigns high probability to high paying txns
- Not too greedy: top txns not necessarily selected

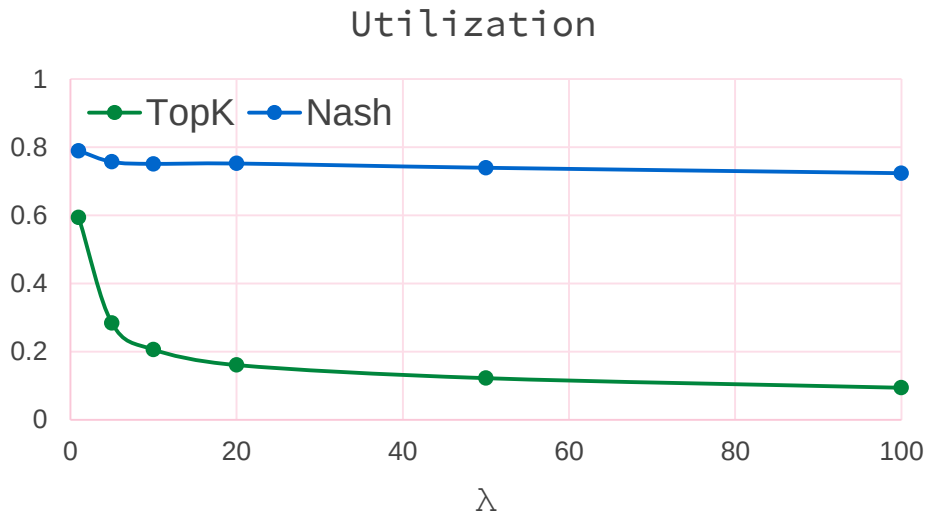
Theorem 6. *Suppose the memory buffer consists of k_l transactions with fee v_l $(1 \le l \le n)$. Denote the individual transactions by $w_1, \ldots, w_m$, which are sorted in descending order of their fees. Denote the index of $v(w_i)$ by $l(w_i)$. The marginal probability $p_i := \frac{q_{l(w_i)}}{k_{l(w_i)}}$ $(1 \le i \le m)$ defines a symmetric equilibrium in the single-shot inclusive-F game, where:*

- $q_l = \begin{cases} k_l \cdot \min\left(f^{-1}\left(\frac{c_{k_{max}}}{v_l}\right), 1\right) & 1 \le l \le k_{max} \\ 0 & k_{max} < l \le n \end{cases}$
- $\forall\, 1 \le l \le n$: $G_l(z) := \sum_{h=1}^{l} k_h \cdot \min\left(f^{-1}\left(\frac{z}{v_h}\right), 1\right) - b$
- $k_{max} := \max\{k \le n \mid \forall l \le k : G_l(v_l) \le 0\}$
- $c_{k_{max}}$ *is the root of* $G_{k_{max}}$.

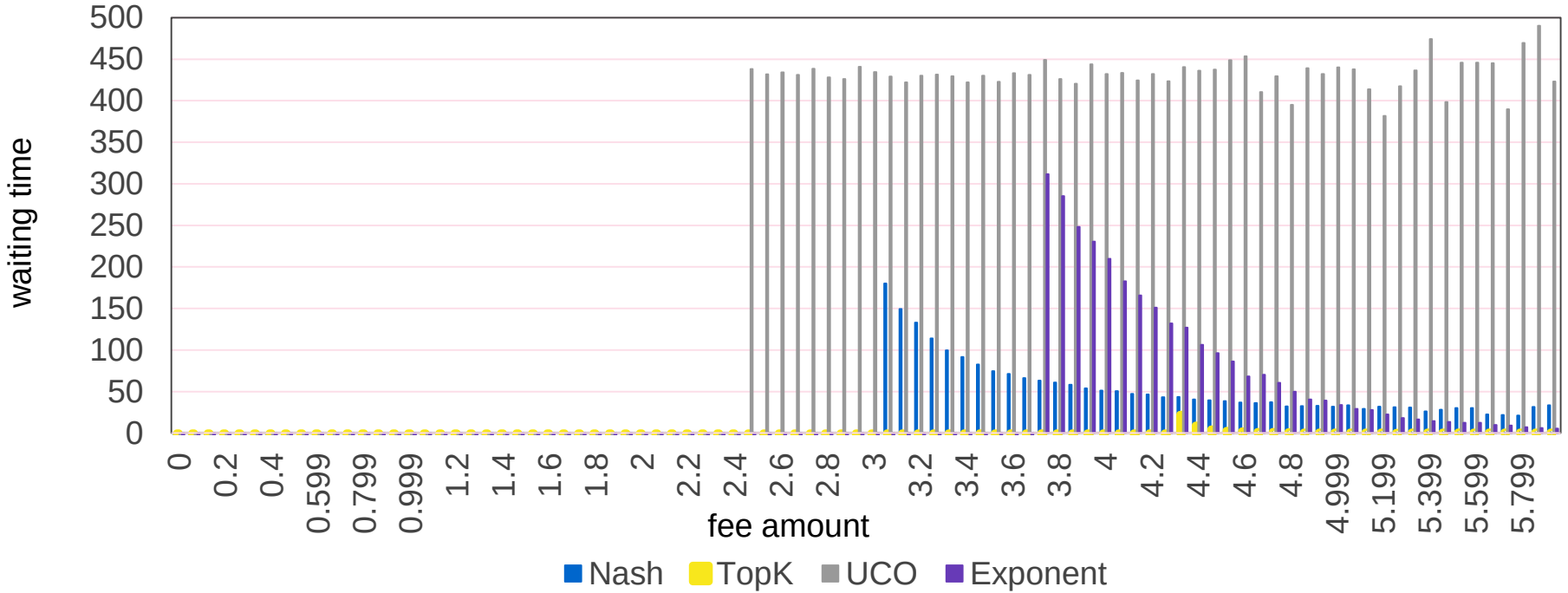
Throughput under Nash

Throughput of: DAG + greedy mining (green)
DAG + Nash Equ. (blue)
DAG + optimal utilization (red)





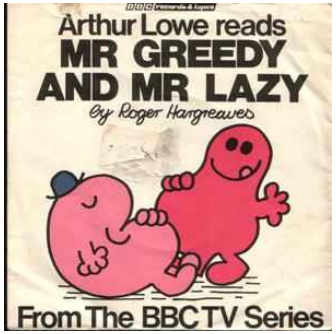
QoS levels


waiting time
500
450
400
350
300
250
200
150
100
50
0
0
0.2
0.4
0.599
0.799
0.999
1.2
1.4
1.6
1.8
2
2.2
2.4
2.6
2.8
3
3.2
3.4
3.6
3.8
4
4.2
4.4
4.6
4.8
4.999
5.199
5.399
5.599
5.799
fee amount
Nash
TopK
UCO
Exponent

Correlated Equilibrium / asymmetric Nash

- Can we do better by somehow coordinating between miners?
- Preliminary results: yes, higher throughput
- Coordination mechanism: using prvs blocks’ randomness
- Future work

Scaling and incentives



- Strategic mining in Bitcoin -- sophisticated, risky
  in DAG -- easy (but also marginal)
- Decisions more granular: *which txns to select?*
  how fast to release blocks?
- “Lazy” selfish mining -- miner is lazy in information sharing, does not contribute reasonable bandwidth

When implementing BlockDAG protocols -- incentives *really* matter

“Bitcoiners of the world, unite!
You have nothing to lose but your chains!“