

The Story of Bitcoin

Eston Schweickart



Slides adapted from Ittay Eyal's System Lunch Talk, April 2014

Motivation

Sending Money is Hard

- Payment between Alice and Bob
- Want to prevent:
 - Double spending
 - Stealing
 - Invalid transactions
- Not a local solution!



Why Not Use Banks?

- Banks can...
 - Monitor transactions, detect fraud
 - Prevent double spending
- But...
 - Puts trust in a single third party
 - Results in higher transaction fees

Bitcoin to the Rescue!

- All transactions public, verifiable
- Majority decides right, not a single party
- Distributed and peer-to-peer, no single point of failure





- Last year, nearly \$14 billion worth of Bitcoin in circulation!
- Only a measly \$5 billion now (~\$400 / 1 BTC)

Bitcoin System Structure

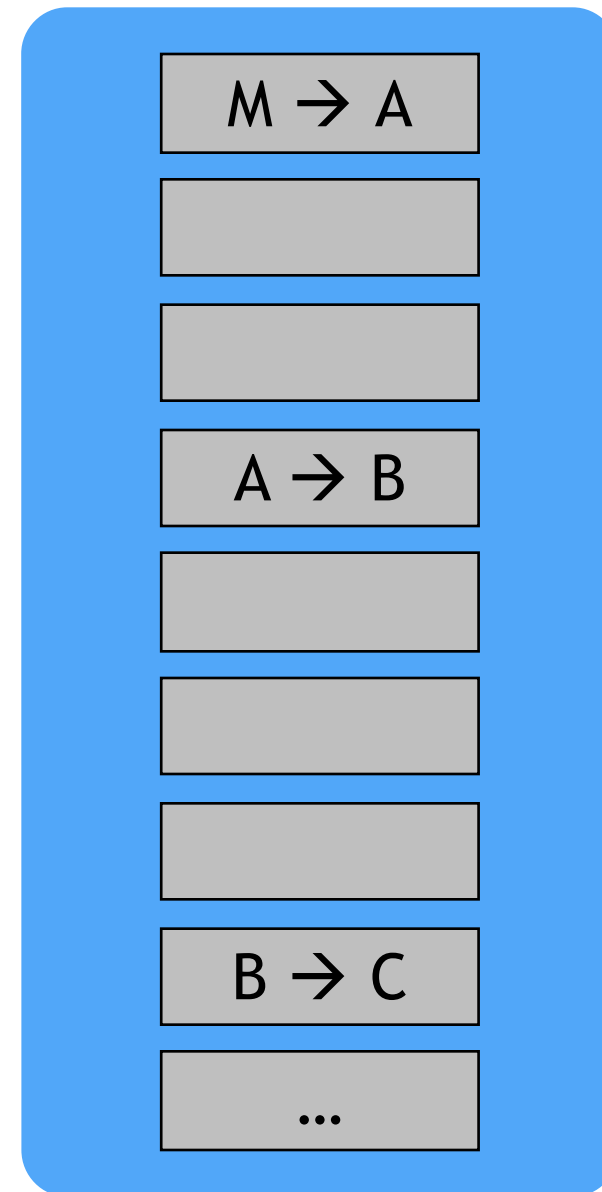
(Nakamoto '08)

Origin

- Developed by Satoshi Nakamoto
 - This is a pseudonym. No one knows the true identity of the original developer(s).
- Whitepaper released in 2008
- Development started soon afterwards

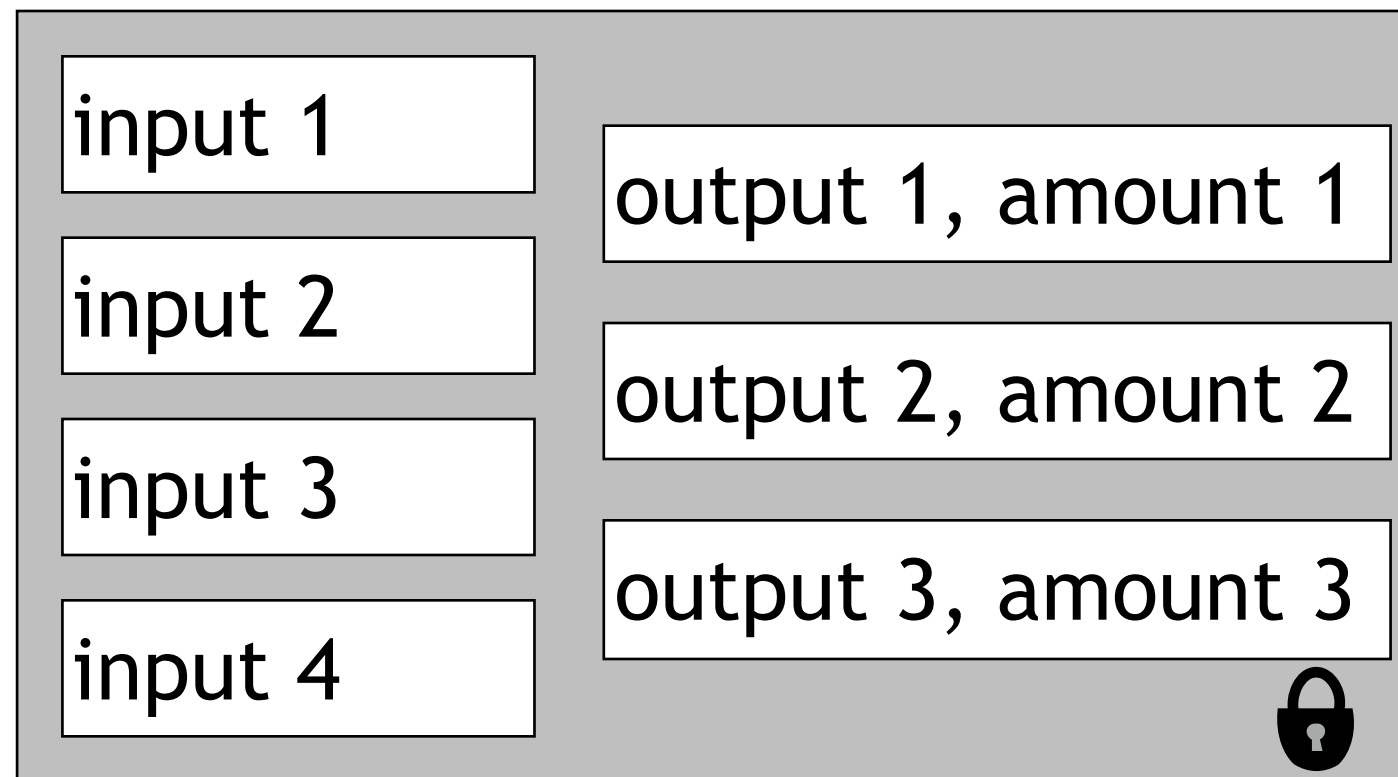
Global Ledger

- All transactions since beginning recorded
- Transactions store origin, recipient, amount
- Special transactions make Bitcoin out of thin air

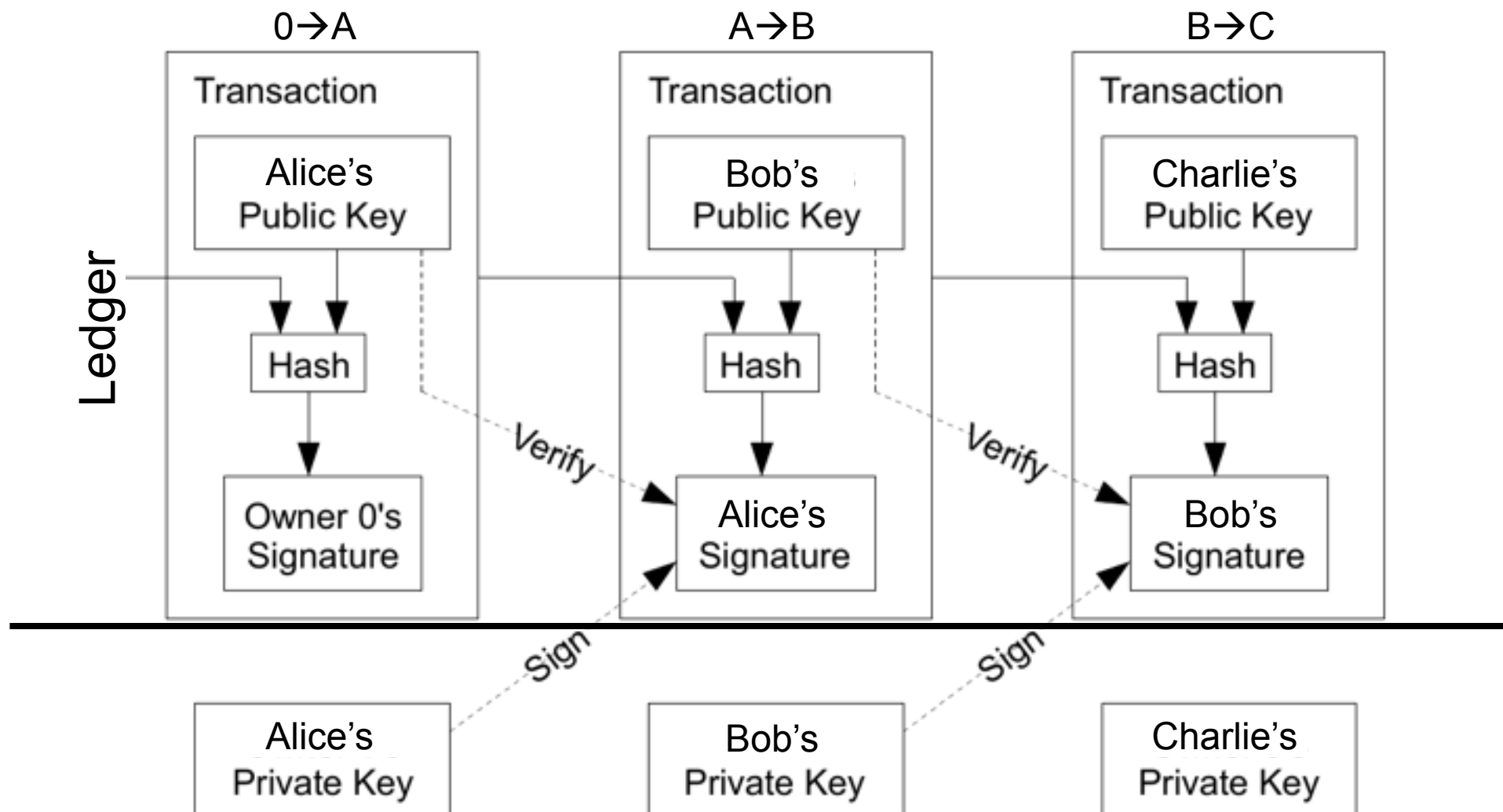


Addresses and Transactions

Transaction structure (roughly):

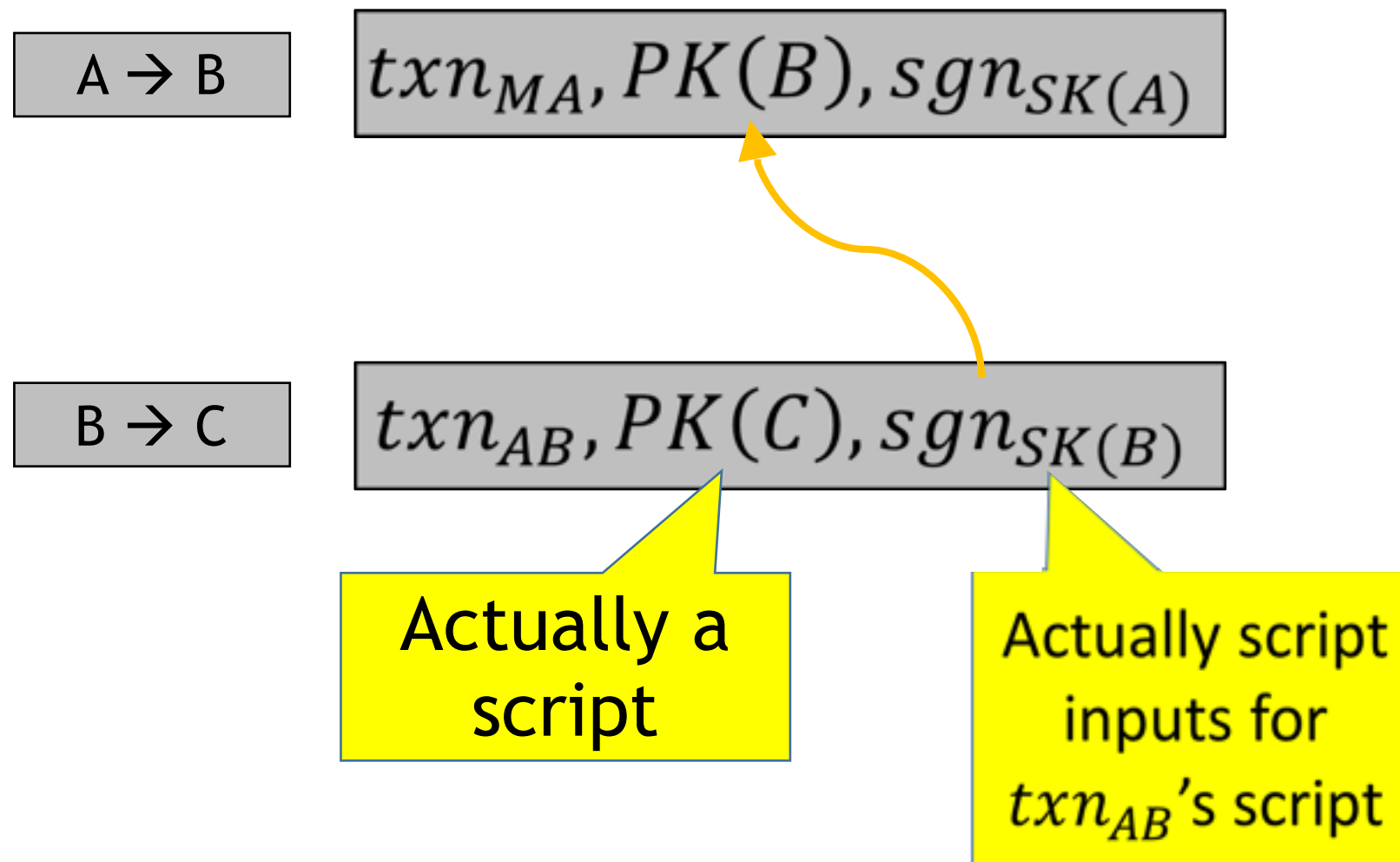


Addresses and Transactions



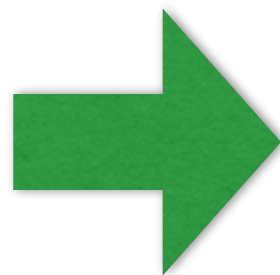
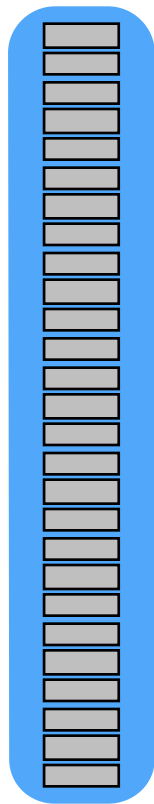
[Nakamoto'08]

Addresses and Transactions

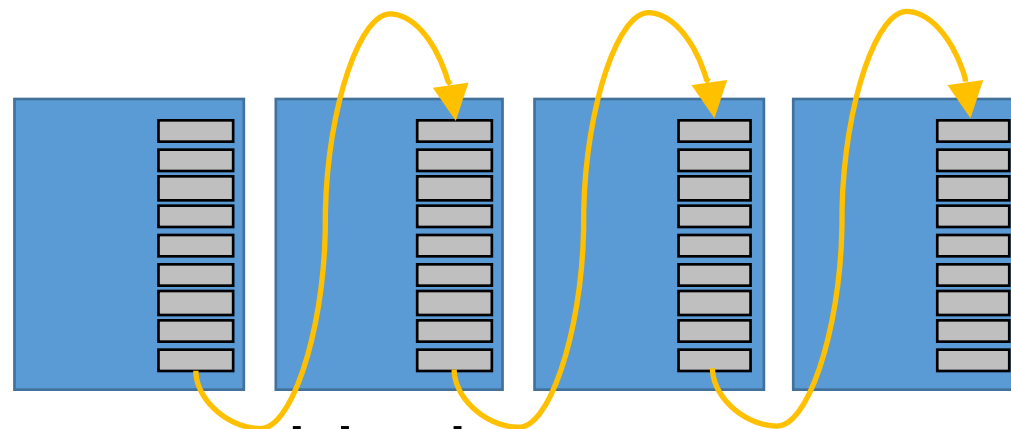


The Blockchain

Ledger



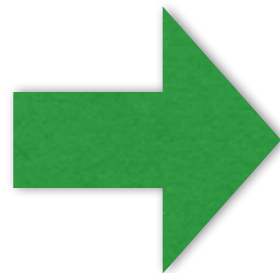
Blockchain



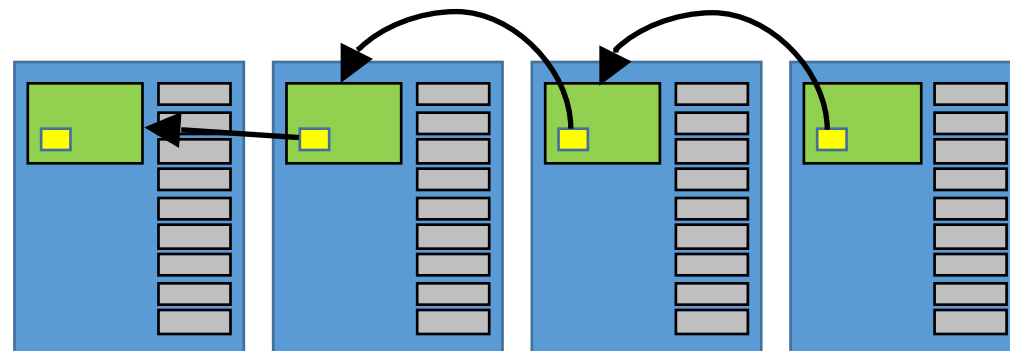
block

The Blockchain

Ledger

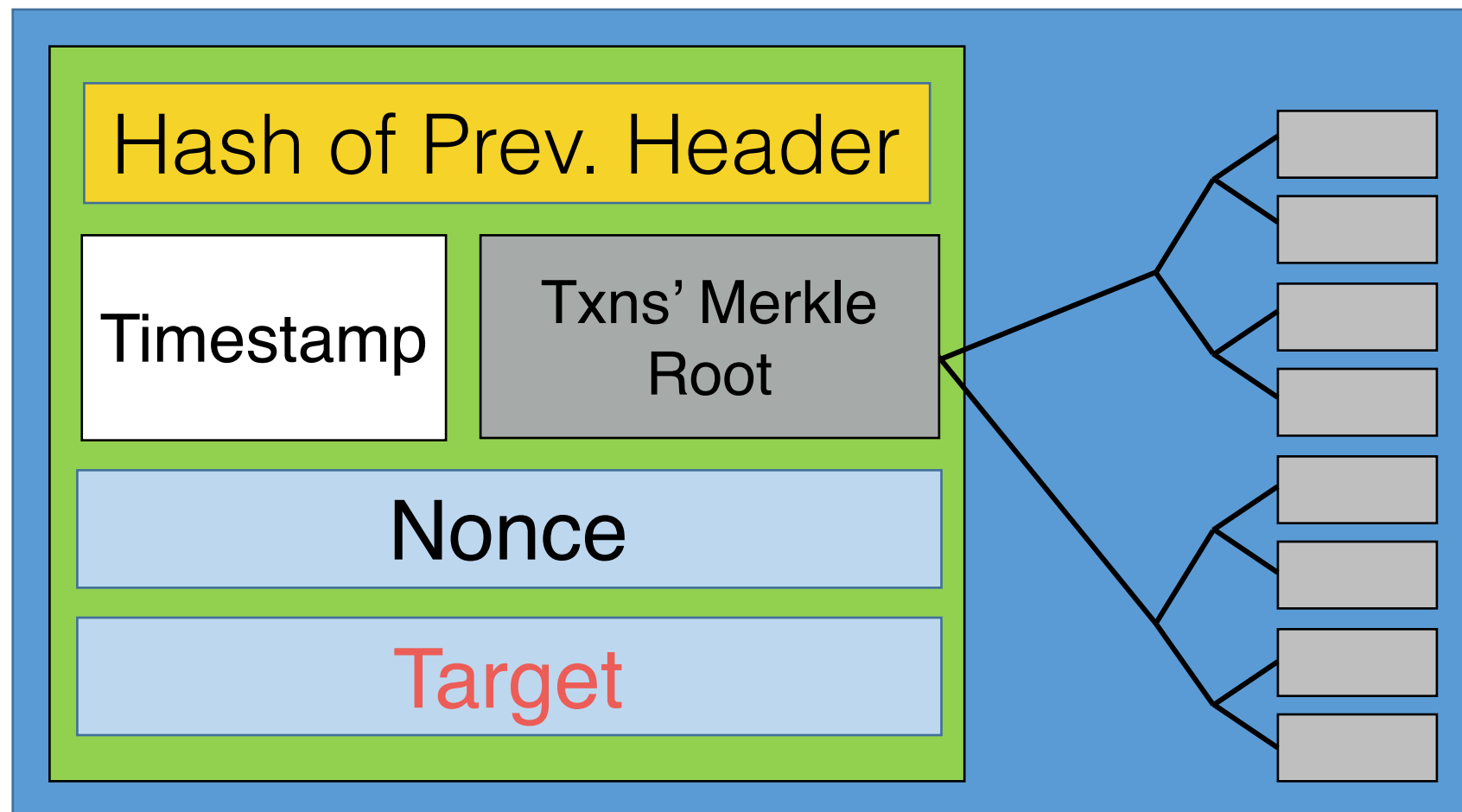


Blockchain



block

Block Header Structure



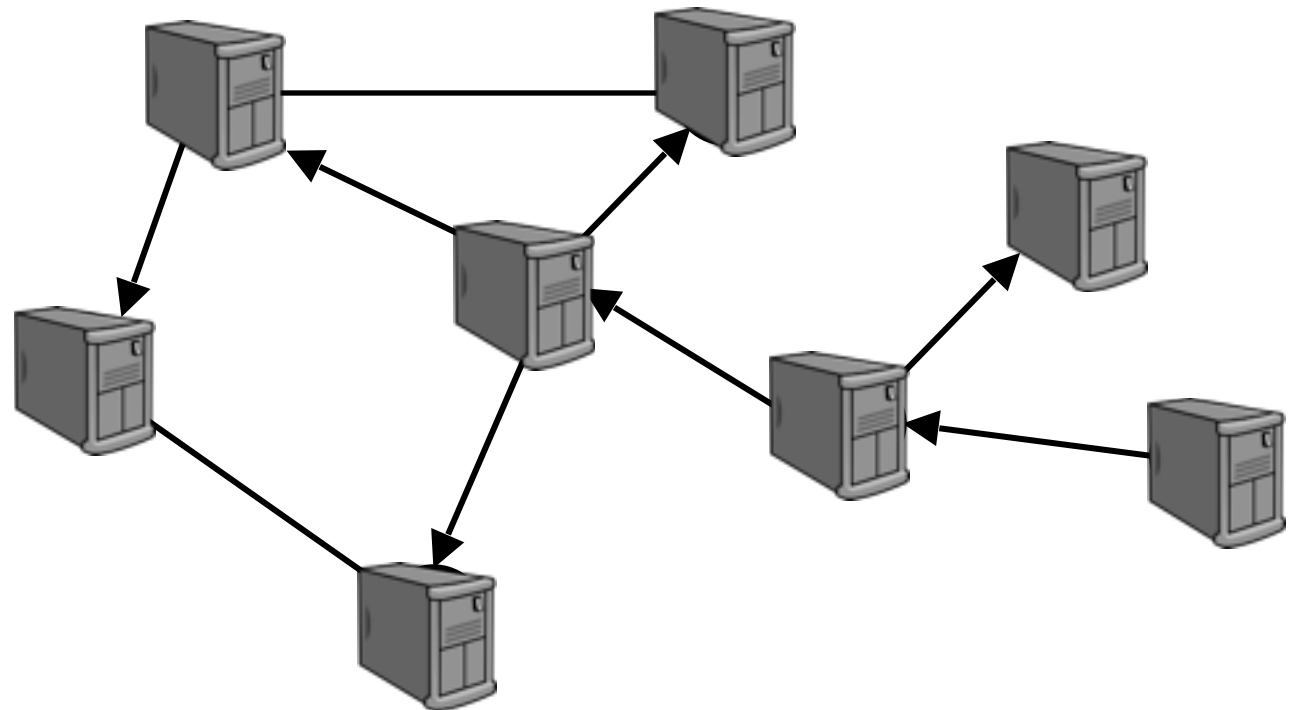
$$\text{SHA256}(\text{SHA256}(\text{Block Header})) < \text{Target}$$

Block Mining

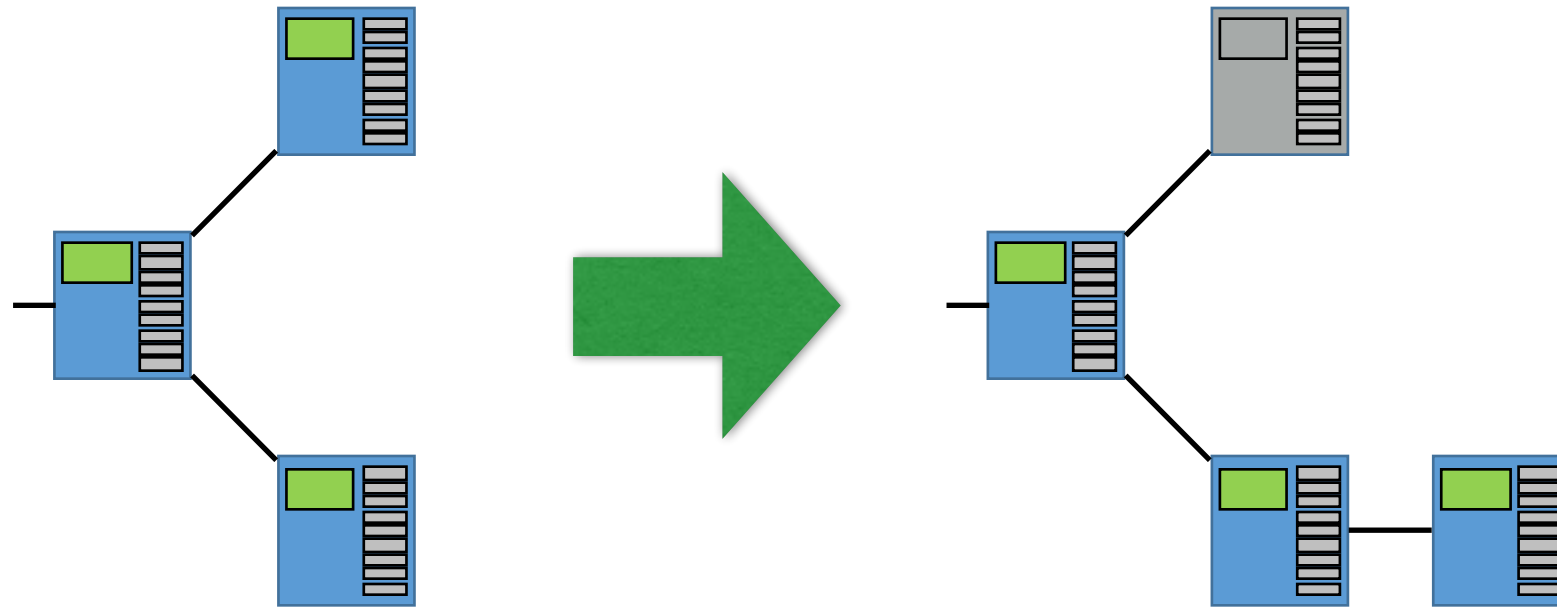
- Block contains nonce => hash contains number of preceding 0s
 - Block mining => inverting a hash function
 - Scalable difficulty in number of 0s (avg. 1/10 minutes by design)
 - Easy to verify
 - Miner gets bitcoin reward in special transaction, plus payer-defined transaction fees
- If blocks accepted to blockchain, transactions committed
 - ...but maybe not forever?

Blockchain Distribution

- Once a block is found, it is distributed to neighbors
- Others verify, then propagate
- Network delays, dropped messages, firewalls, etc. create forks in blockchain

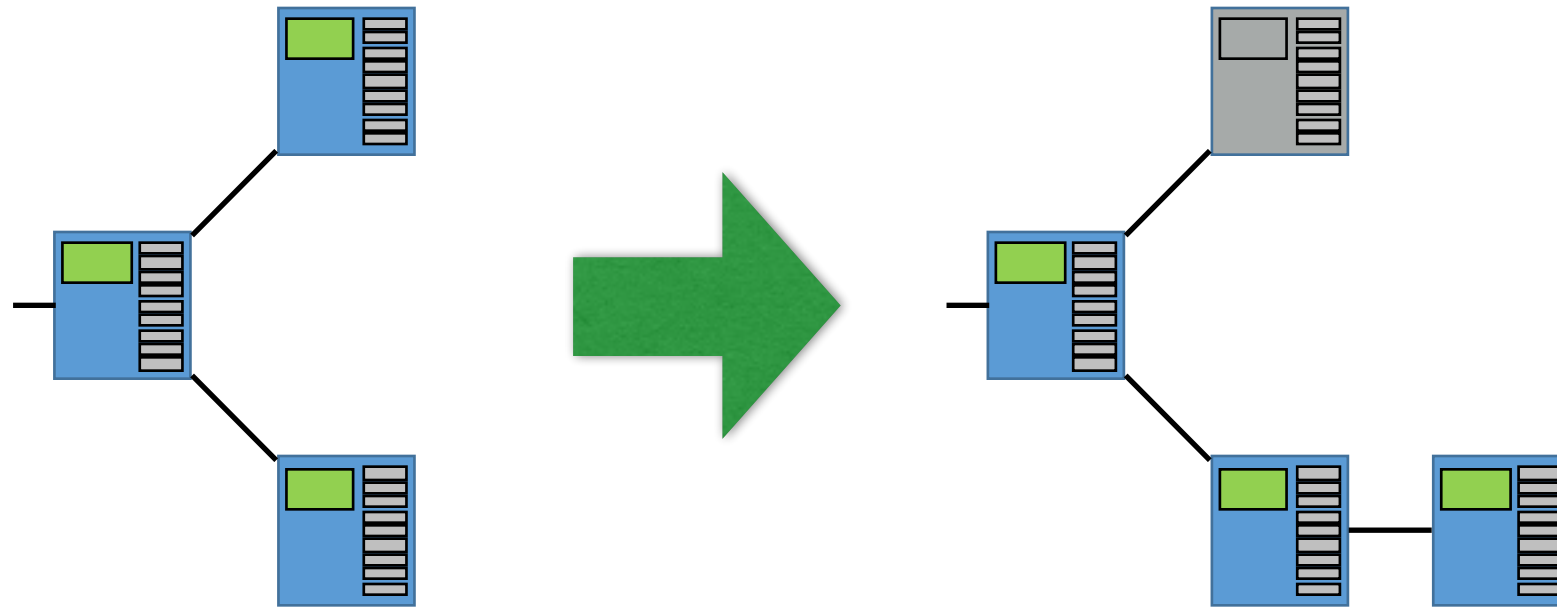


Blockchain Forks



- Conflicting blocks stored
- Longest chain first wins
- Other blocks discarded (!!!)

Blockchain Forks



- If majority of CPU power is honest, no invalid transactions
- Vanishing likelihood of beating majority's chain

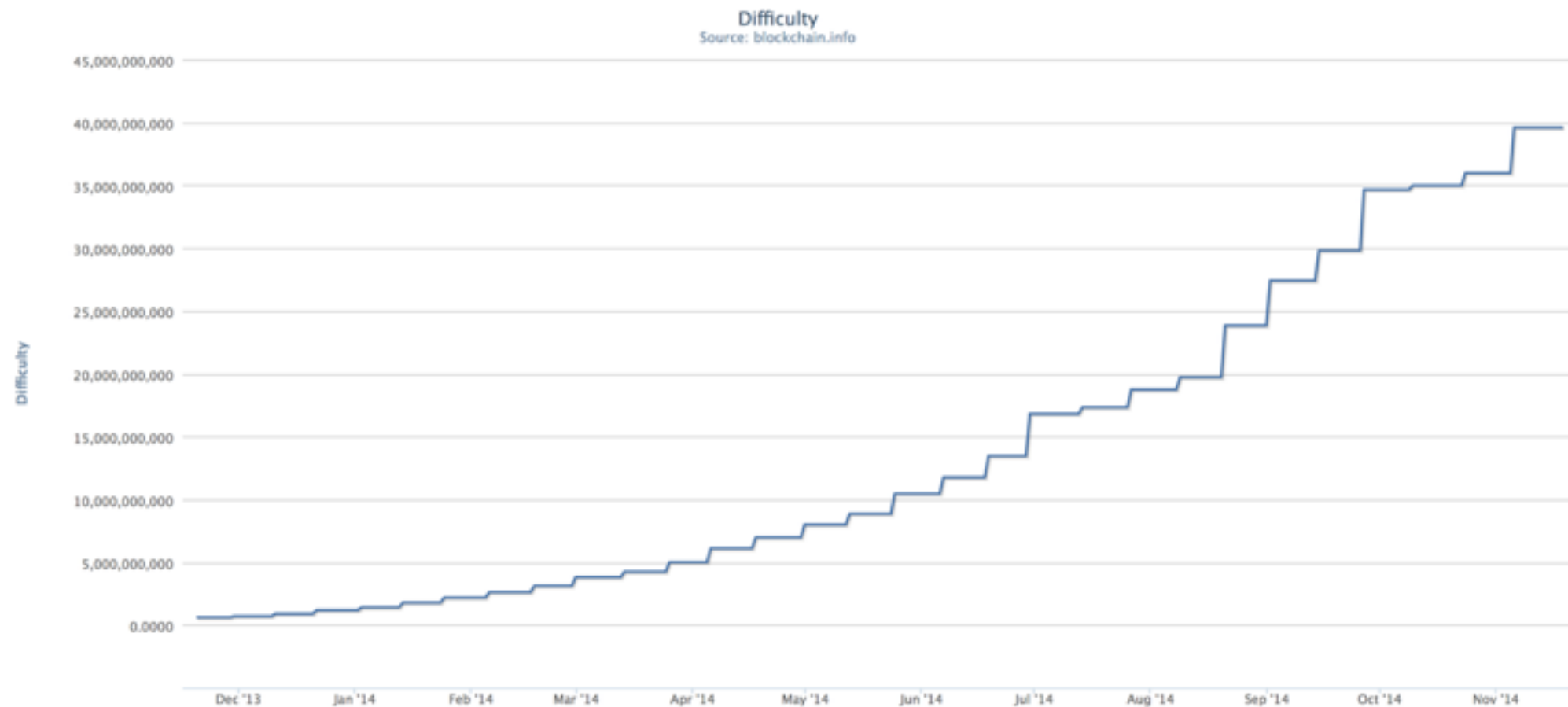
One Way Things Can Go Wrong

(Eyal and Sirer '13)

Enormous Amount of Mining Computation Power



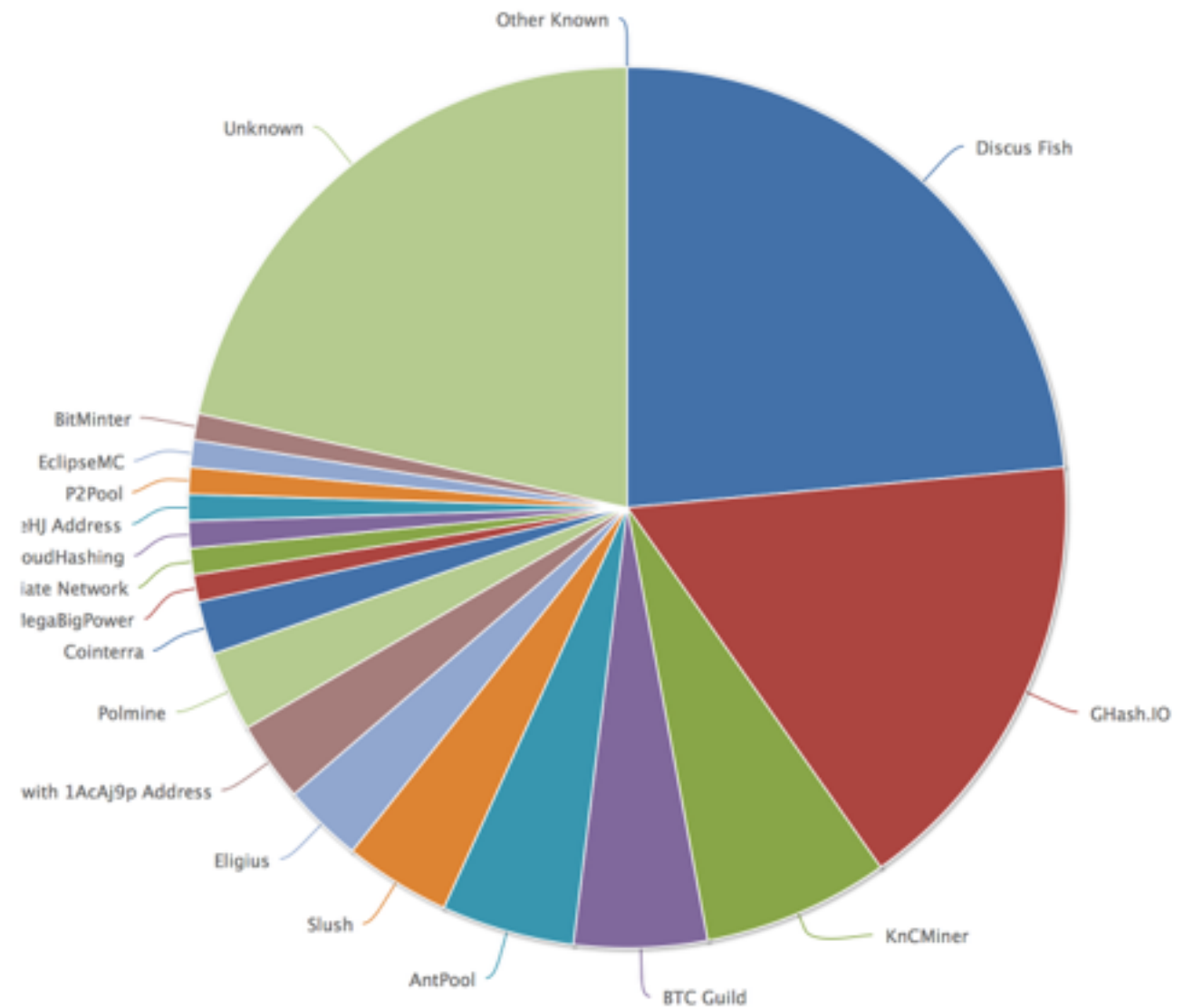
Mining Difficulty



- Roughly 40,000,000,000x more difficult to mine blocks than in 2009
- Mining a single block could take years!

Mining Pools

- Form groups to consolidate CPU power
- Lower variance in payoff

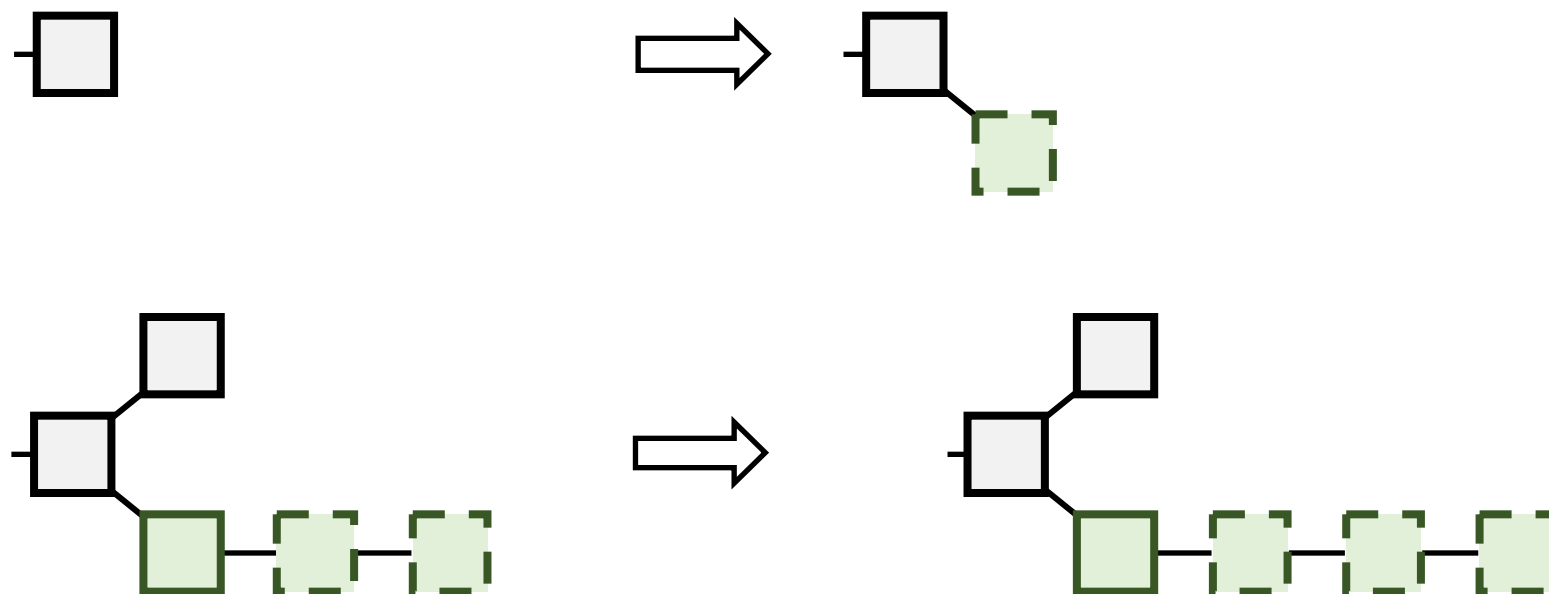


How to Beat the System

- Nakamoto model assumes honest majority, no hidden information
 - Best payoff: being honest
- But what if we hide information?
- Idea: don't publish blocks we find
 - Keep track of public chain and secret chain
 - Try to get the secret chain ahead of public, then reveal

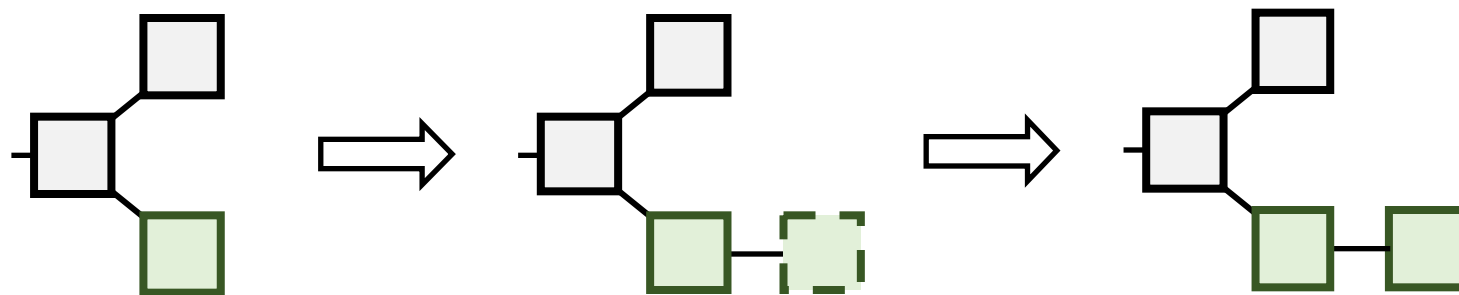
Selfish Mining

- Case (a): Any state but two branches of length 1. We find a block.
- Result: Keep it secret. No revenue yet.



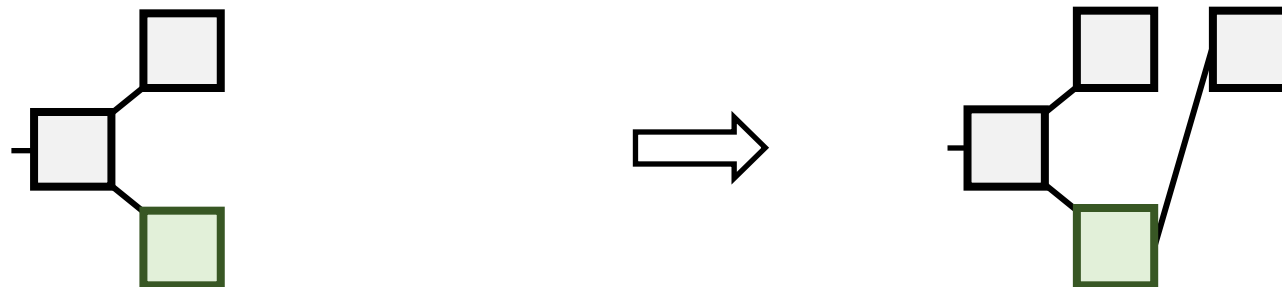
Selfish Mining

- Case (b): Two branches of length 1. We find a block.
- Result: Publish it! +2 for us!



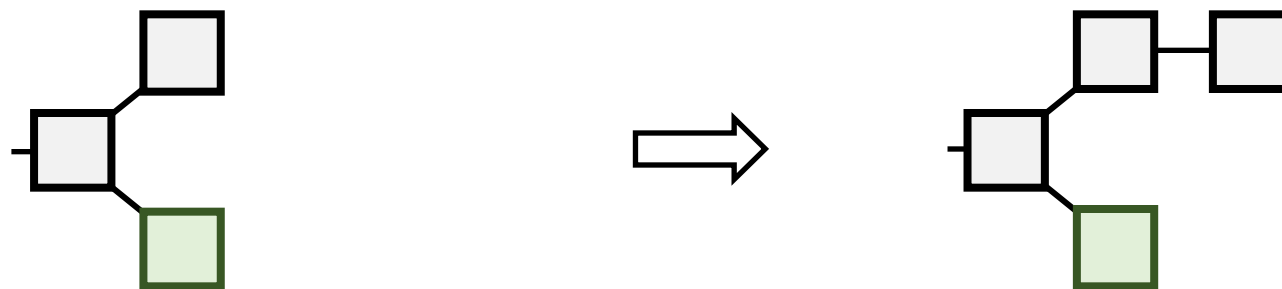
Selfish Mining

- Case (c): Two branches of length 1. Public finds a block on top of ours.
- Result: Not bad, +1 to either side.



Selfish Mining

- Case (d): Two branches of length 1. Public finds a block on top of theirs.
- Result: +2 for them. At this point, we should abandon our chain.



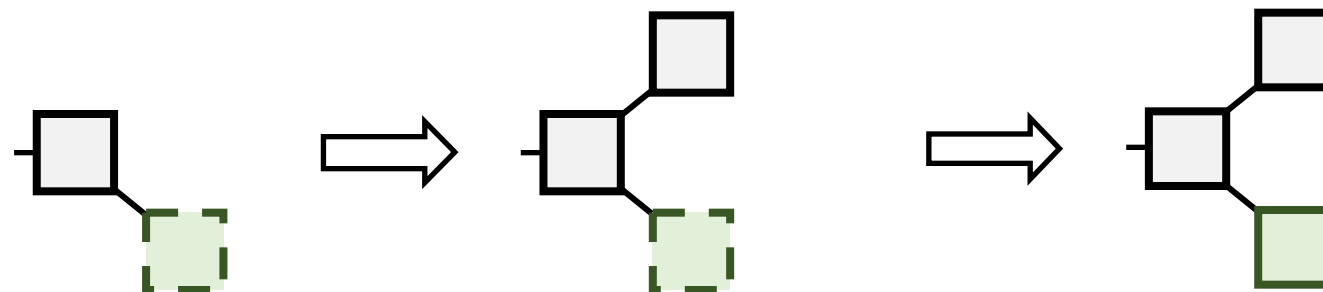
Selfish Mining

- Case (e): No gain over public branch. Public finds a block.
- Result: +1 for them. At this point, we should mine on the new block.



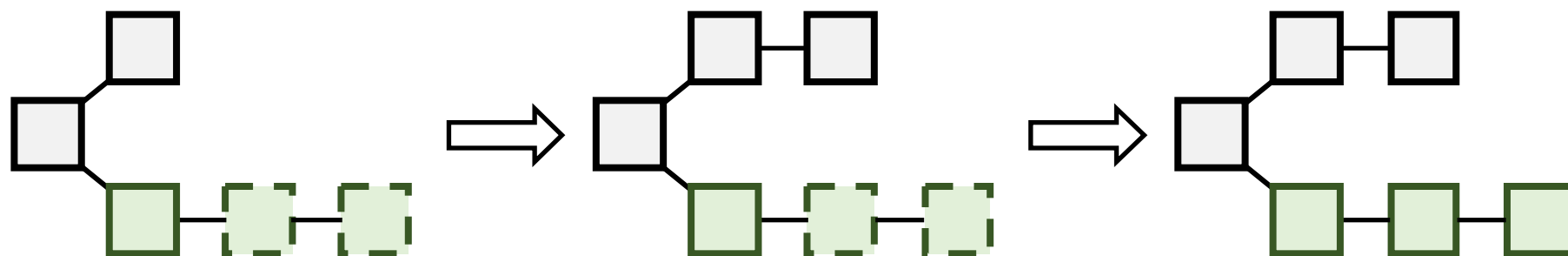
Selfish Mining

- Case (f): Our chain is 1 longer. Public finds a block.
- Result: Publish ours, intentionally creating a fork. No profit yet.



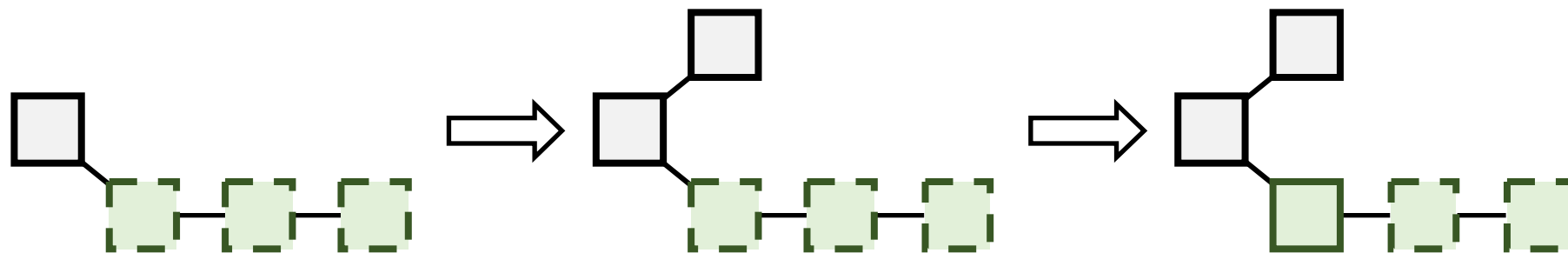
Selfish Mining

- Case (g): Our chain is 2 longer. Public finds a block.
- Result: Publish! +2 (or more) for us!

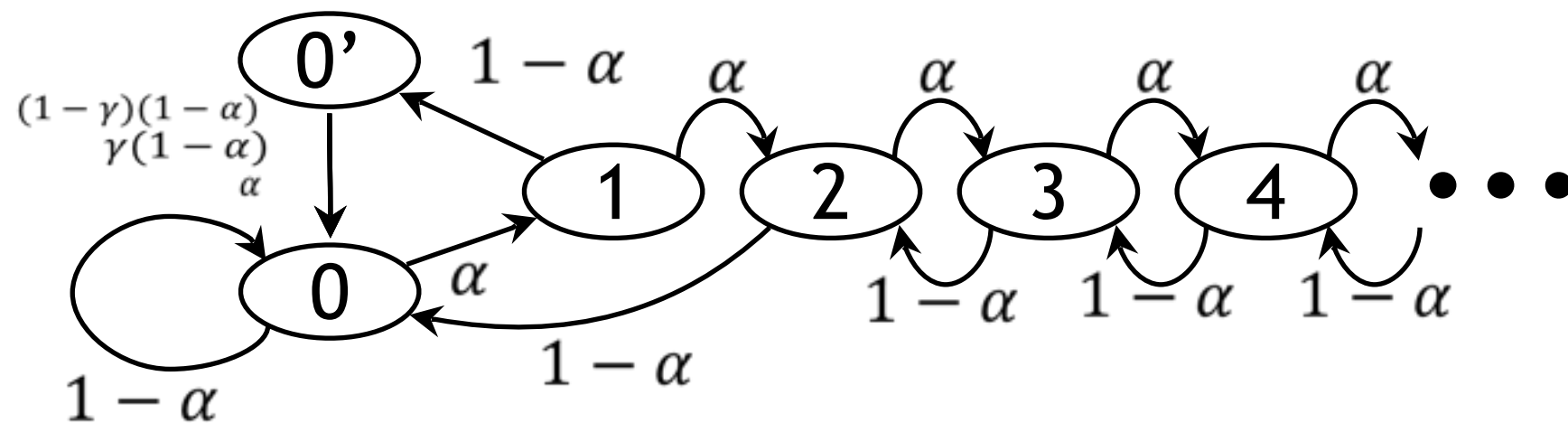


Selfish Mining

- Case (h): We lead by more than 2. Public finds a block.
- Result: Publish one, intentionally creating a fork (if one didn't exist already)



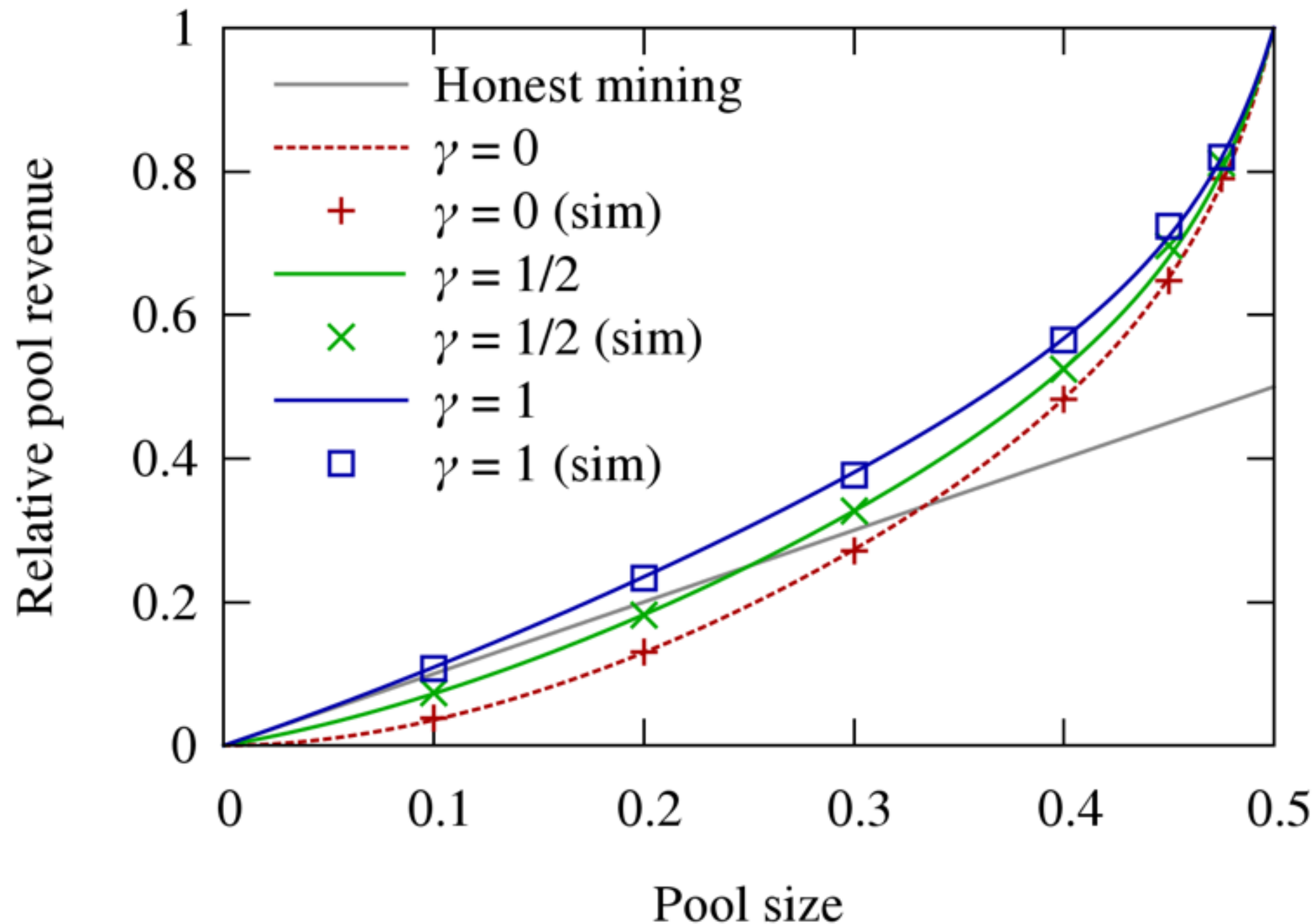
Selfish Mining Analysis



$$\begin{aligned}
 r_{\text{others}} &= \overbrace{p_{0'} \cdot \gamma(1-\alpha) \cdot 1}^{\text{Case (c)}} + \overbrace{p_{0'} \cdot (1-\gamma)(1-\alpha) \cdot 2}^{\text{Case (d)}} + \overbrace{p_0 \cdot (1-\alpha) \cdot 1}^{\text{Case (e)}} \\
 r_{\text{pool}} &= \overbrace{p_{0'} \cdot \alpha \cdot 2}^{\text{Case (b)}} + \overbrace{p_{0'} \cdot \gamma(1-\alpha) \cdot 1}^{\text{Case (c)}} + \overbrace{p_2 \cdot (1-\alpha) \cdot 2}^{\text{Case (g)}} + \overbrace{P[i > 2](1-\alpha) \cdot 1}^{\text{Case (h)}}
 \end{aligned}$$

α is our relative computational power, γ is the portion of the public that mines on our published blocks

Selfish Mining Analysis



Selfish Mining Analysis

- With $\gamma=1$, there is incentive to mine selfishly *even with very limited computational power*
 - In the Nakamoto model, feasible with selfish scout nodes
- Even with $\gamma=0$, incentive exists at 33%
- With $\gamma=1/2$, incentive exists at 25%
 - If nodes randomly mine on one of fork heads, this is the case — but no patch released to my knowledge

Consequences

- Before: no real benefit to joining large pool
- Now: if a selfish pool is more profitable than others, new miners will prefer it
- Incentive only increases as users join
- Once 51% is reached, pool has unlimited control over transactions! Bitcoin would die!

Reactions

SECURITY

Bitcopocalypse! Top crypto-currency can be HIJACKED, warn boffins

Selfish miners could derail Bitcoin's decentralized design to new study

by Jack Clark, 5 Nov 2013

Basically the picture you are trying to paint is:

1. All us Bitcoin fanatics zealously believed Bitcoin is perfect and has no vulnerabilities / open problems at all.
2. You heroically showed us that Bitcoin is totally broken and we should all abandon it immediately.

I disagree with both parts of this picture.

As for the accuracy of the research results themselves, they are still to be verified.

Another day, another Bitcoin burglary as Bitcash.cz goes titsup


The foundation on which Bitcoin is built is a sequential list of blocks that contain a small set of Bitcoin transactions, stored securely and permanently.

Hacking, Distributed

Bitcoin Is Broken

Ittay Eyal, and Emin Gün Sirer

Bitcoin is broken. And not just superficially so, but fundamentally, at the core protocol level. We're not talking about a simple buffer overflow here, or even a badly designed API that can be easily patched; instead the problem is intrinsic to the entire way Bitcoin works.

 **Shana Pippet** → Emin Gun Sirer · a year ago

Your arrogance is incredible

19 ^ | v · Reply · Share ›

And Then, The Unthinkable

- June 2014: Mining pool GHash, operated by CEX.io, passes 55% of the total mining power
- ...for about 24 hours
- “Is this really armageddon? Yes, it is.” -IE & EGS
- Now things appear to be stable (no one pool over 25%)

Notes on Network Topology and Message Propagation

(Decker and Wattenhofer '13)

Network Topology

- When nodes join the network, they first query several DNS nodes, which return bootstrap nodes
- Neighbors advertise themselves; random path is followed through the network
- If you accept incoming connections, you have more neighbors
- Warning: DNS impersonation could give control of network topology!

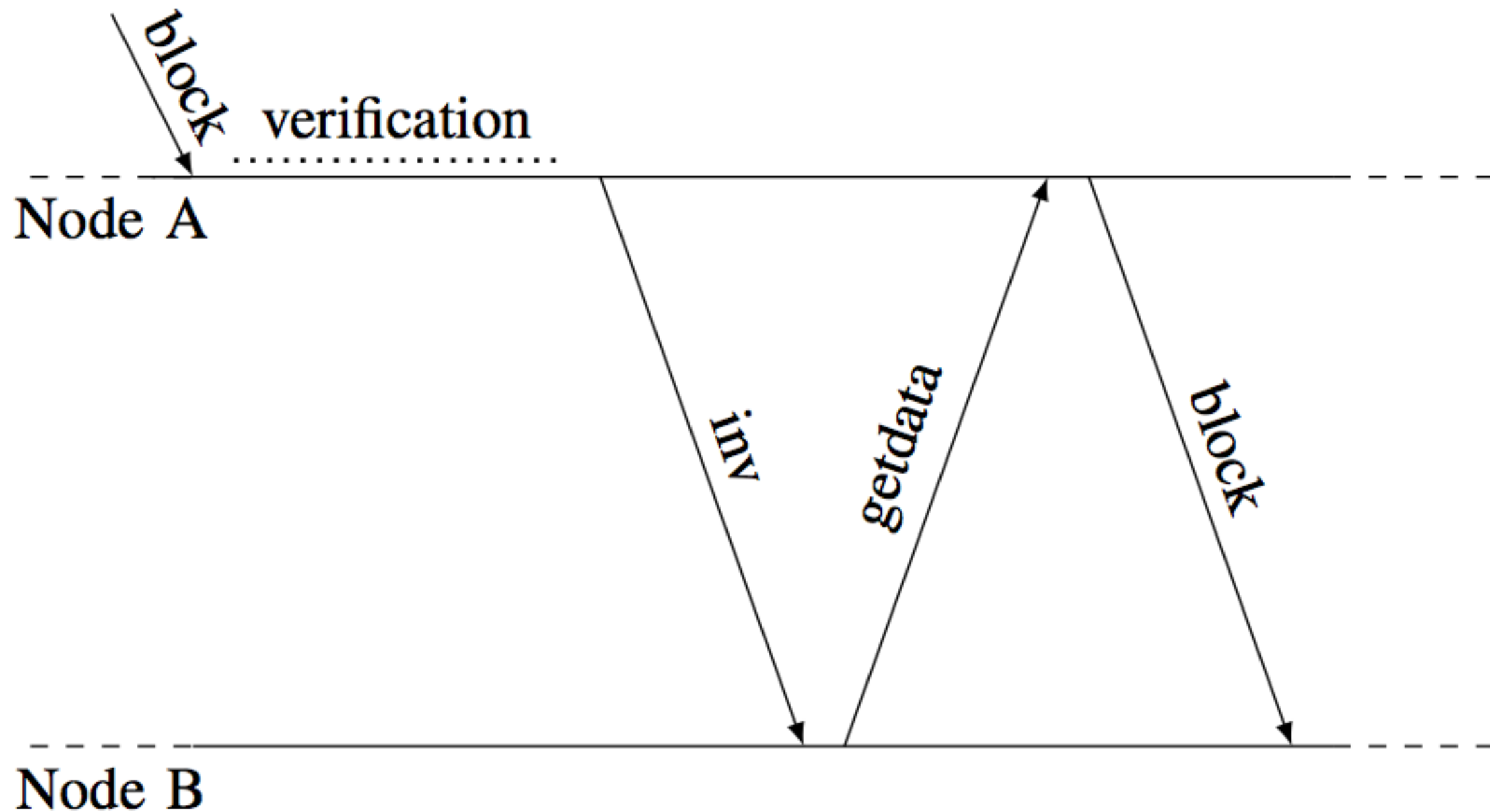
Network Topology

- What happens in a network partition scenario?
- Both halves continue on independently
- Warning: this creates a large fork that will invalidate many transactions when partition is healed!
- Solution: use block discovery rate to detect partitions

Transactions

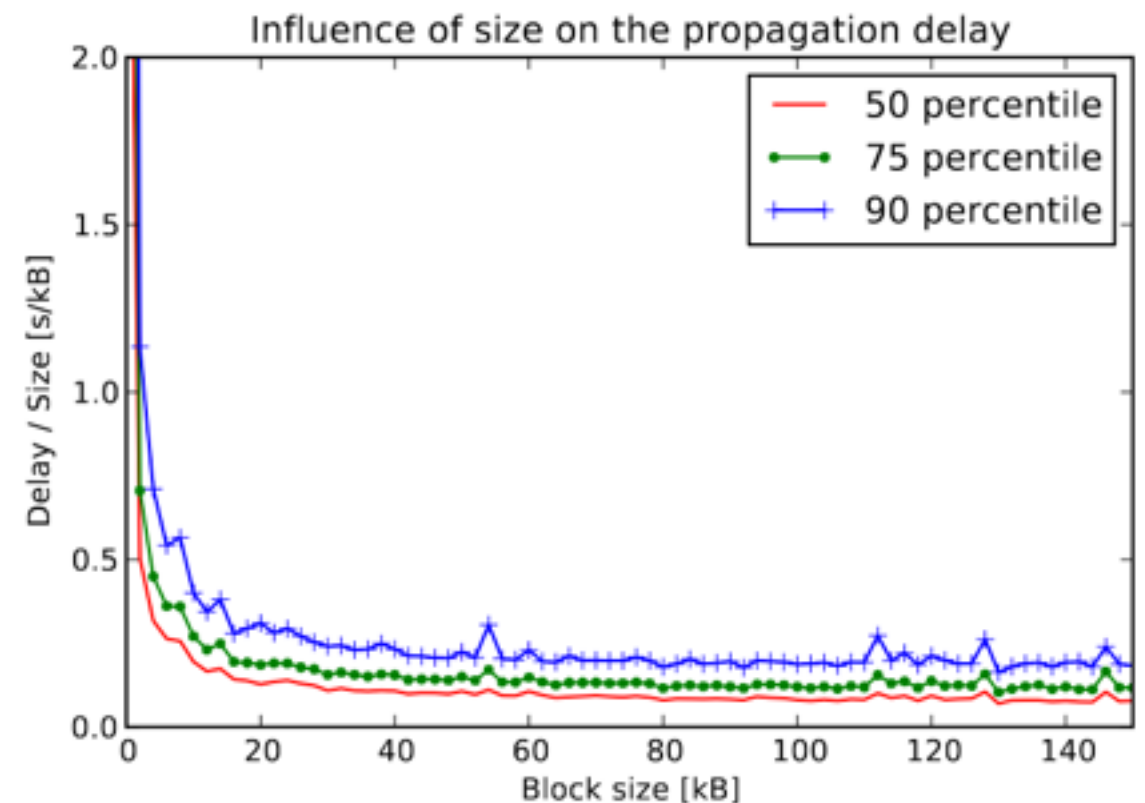
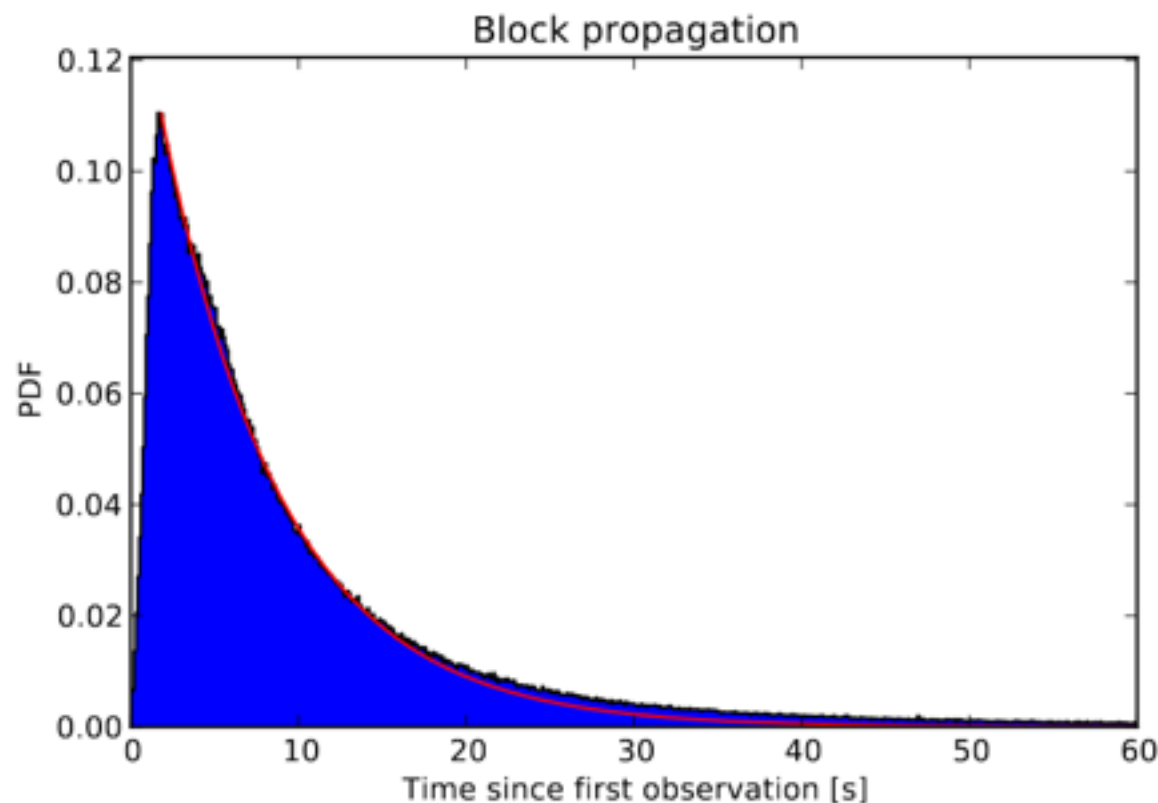
- Propagated through the network from a single node or a few seeds
- Warning: transaction fees mean less incentive to propagate to neighbors! (Babaioff et al '11)
 - Could result in large wait times at the coffee shop
- Solution:
 - Distribute transaction fees among miners in the information chain
 - Distribute to a few seeds rather than just one

Block Propagation



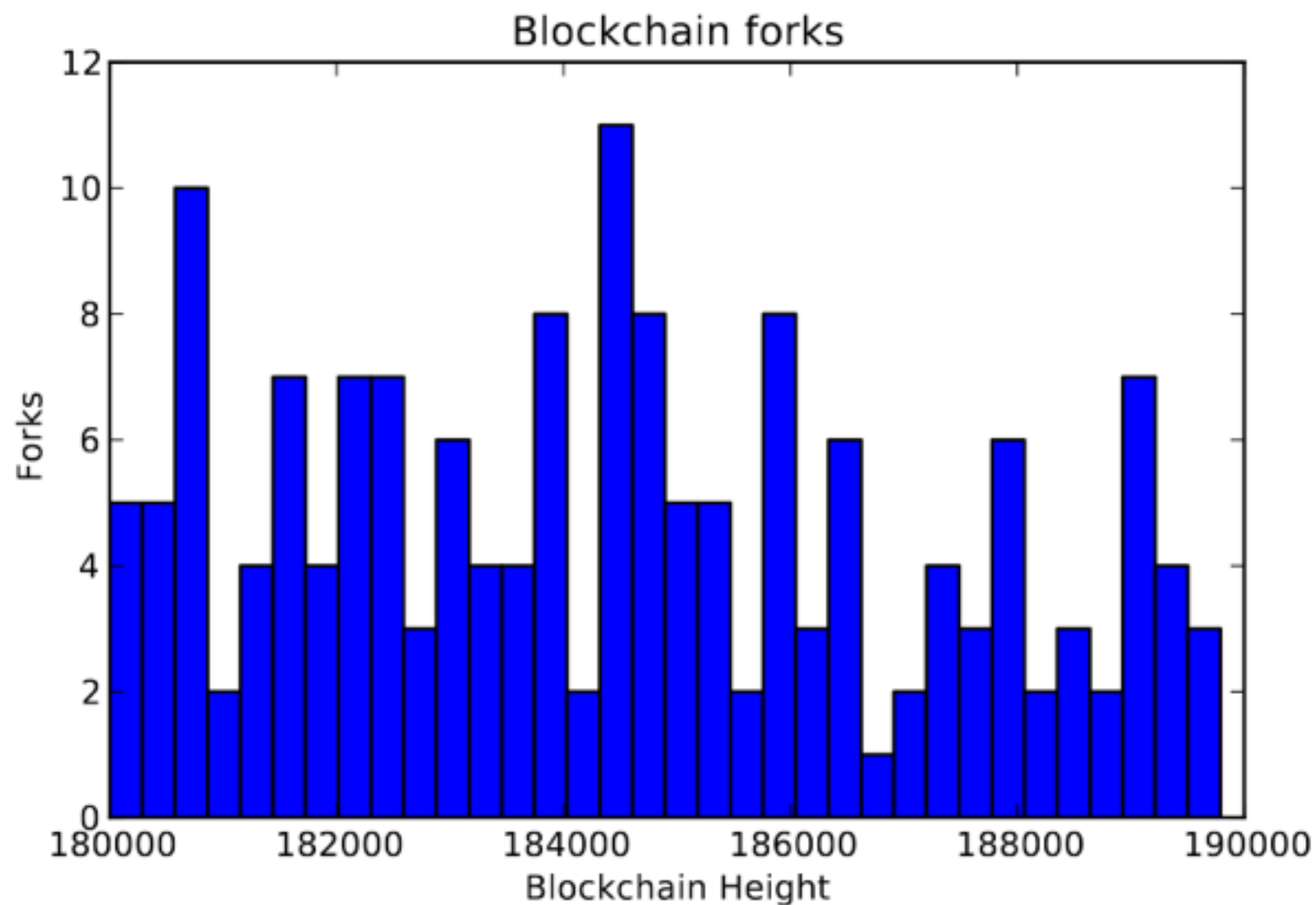
Block Propagation

- Size of block important factor in propagation delay
- Like gossip, long tail on distribution

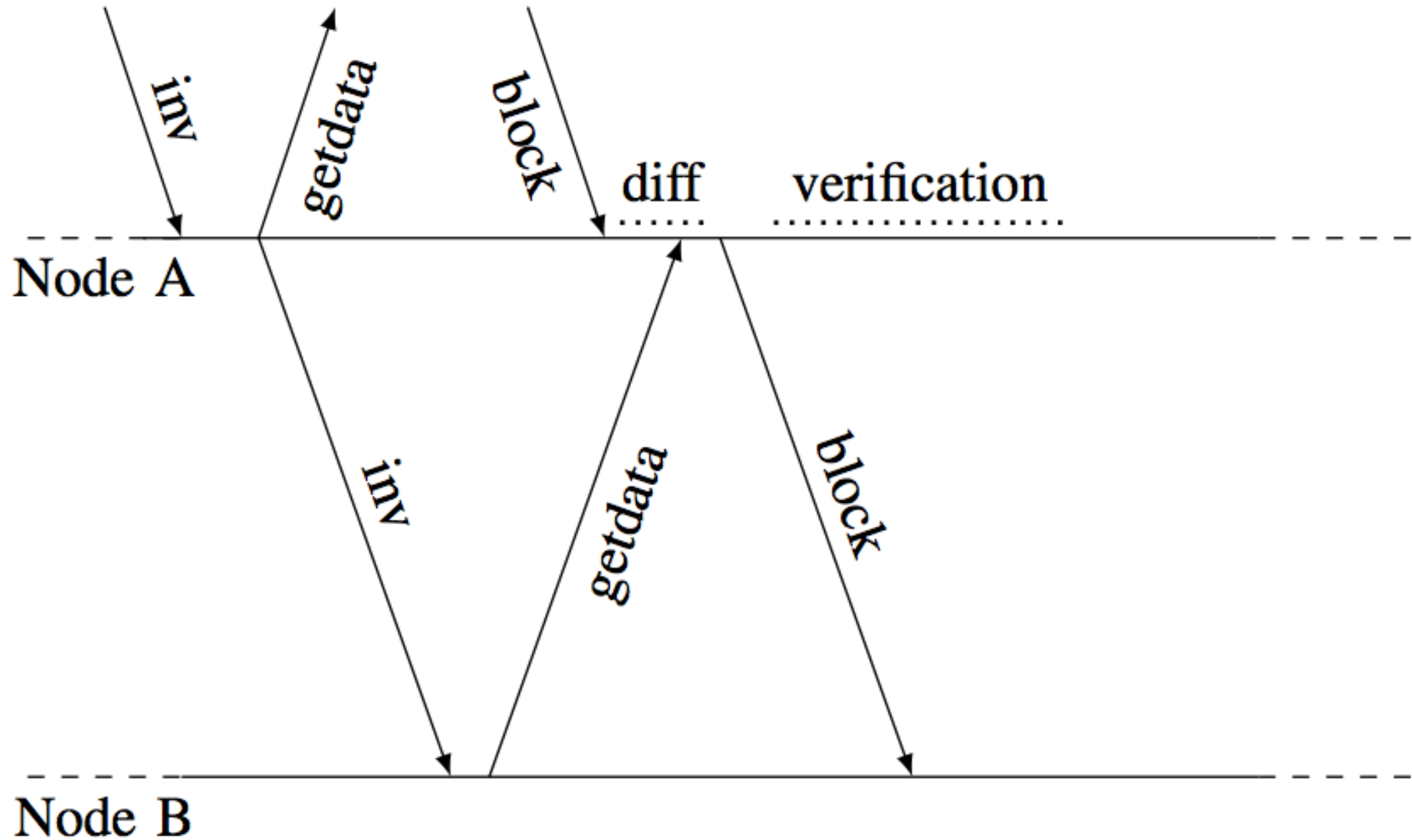


Block Distribution

- Warning: long tails cause blockchain forks! (~1.69%)



Solutions

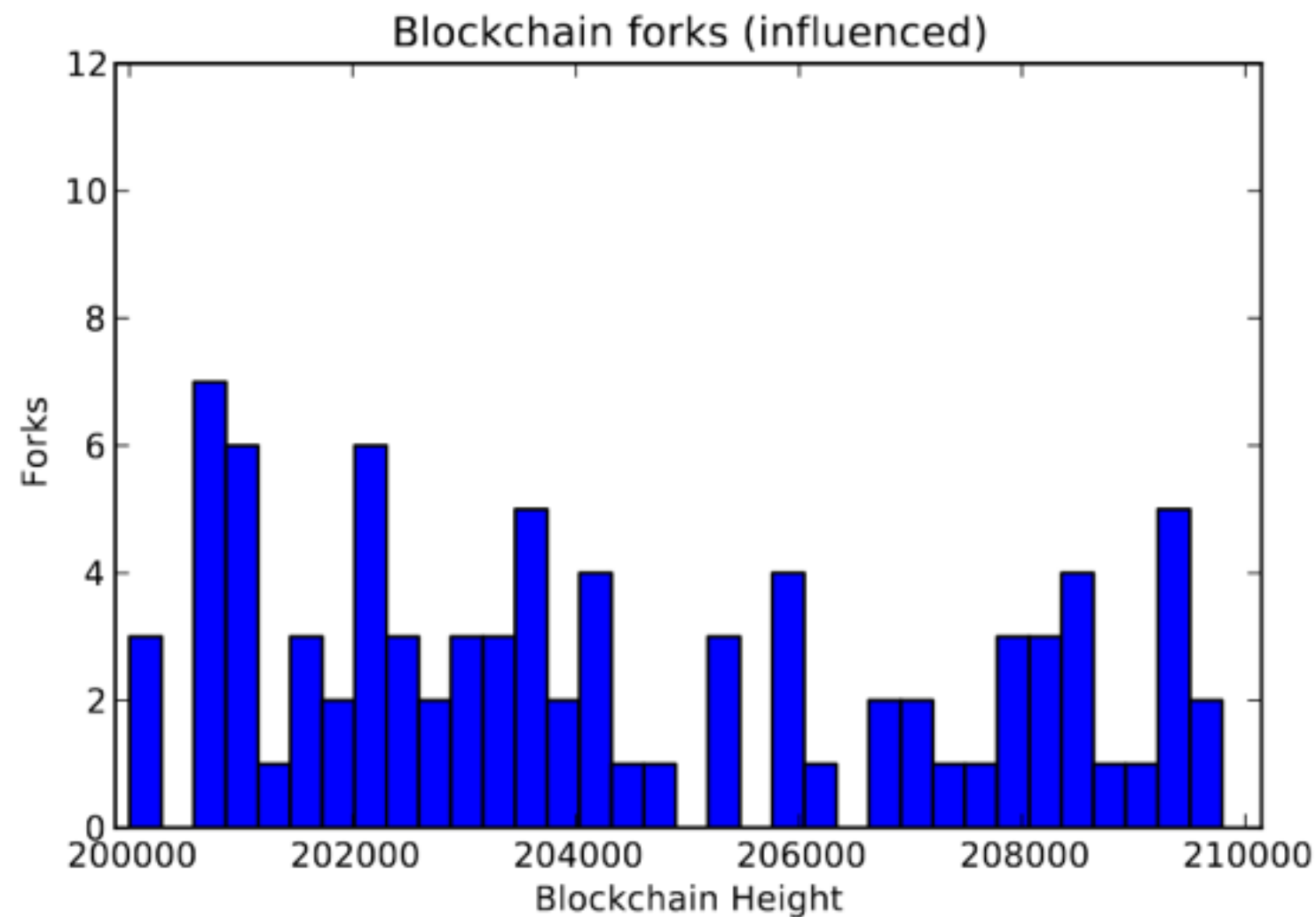


Solutions

- Pro: blocks distributed with less delay
- Con: invalid blocks are propagated
 - Could result in DoS attacks => more forks!
 - Hybrid model might work
- Other solutions: increase network connectivity
 - Greatly helps propagation speed if honest

Solutions

- Reduces block chain forks by about 1/2 (0.78%)



Discussion

- Is a non-permanent commit model the right choice?
- Is selfish mining detectable? Fixable?
- What about the other issues and potential attacks?
- Is cryptocurrency a feasible alternative to modern day currencies?