Bitcoin-NG
A Scalable Blockchain Protocol

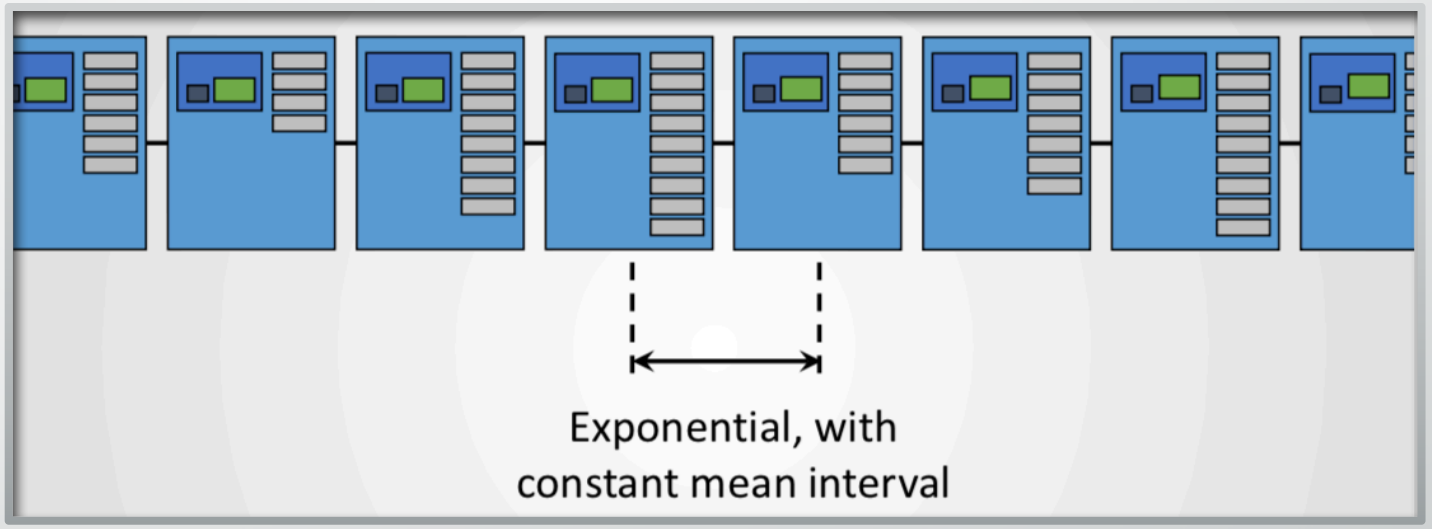
Presented By: Muwei Zheng


1. Review of Bitcoin
2. Problem need to be addressed
3. Bitcoin-NG
4. Performance Evaluation
5. Take-away

Agenda

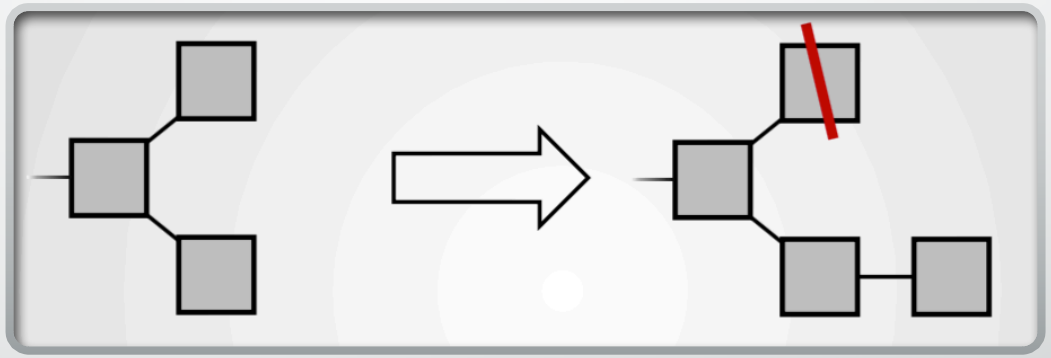
Review of Bitcoin

- Decentralized, P2P
- Hash Block
  - Header
  - Transactions
- Target
  - Hash puzzle
  - Mean interval: 10 mins
- Size
  - 1 MB
  - Debate



Review of Bitcoin

- Fork
  - Longest Chain
  - Pruned

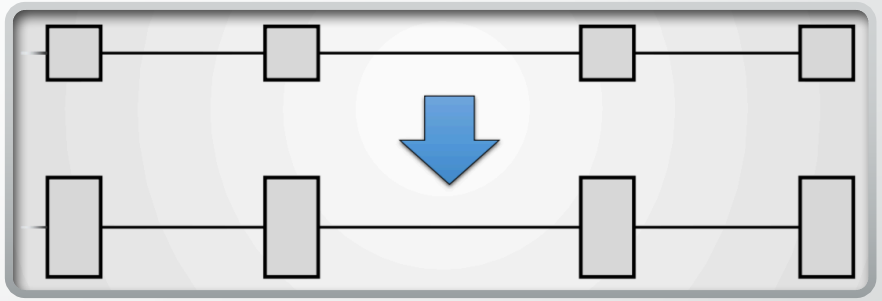


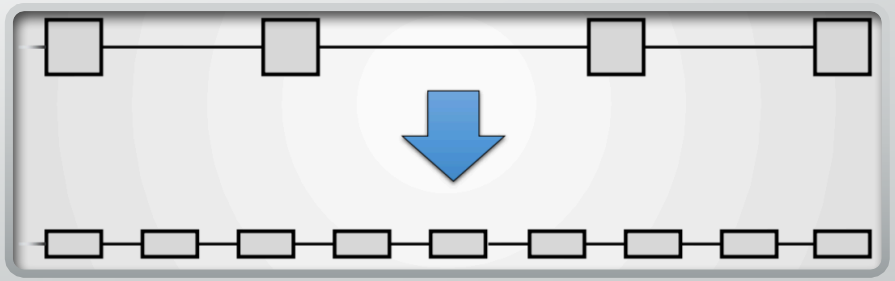
Problem: Scalability

- Max Throughput:
  - Max Transactions per Block
    - $\frac{1{,}000{,}000\ Bytes}{495\ Bytes} = 2020$
  - Max Throughput:
    - $\frac{2020\ transactions}{10\ mins} = 3.37\ tps$

Problem: Potential Solutions

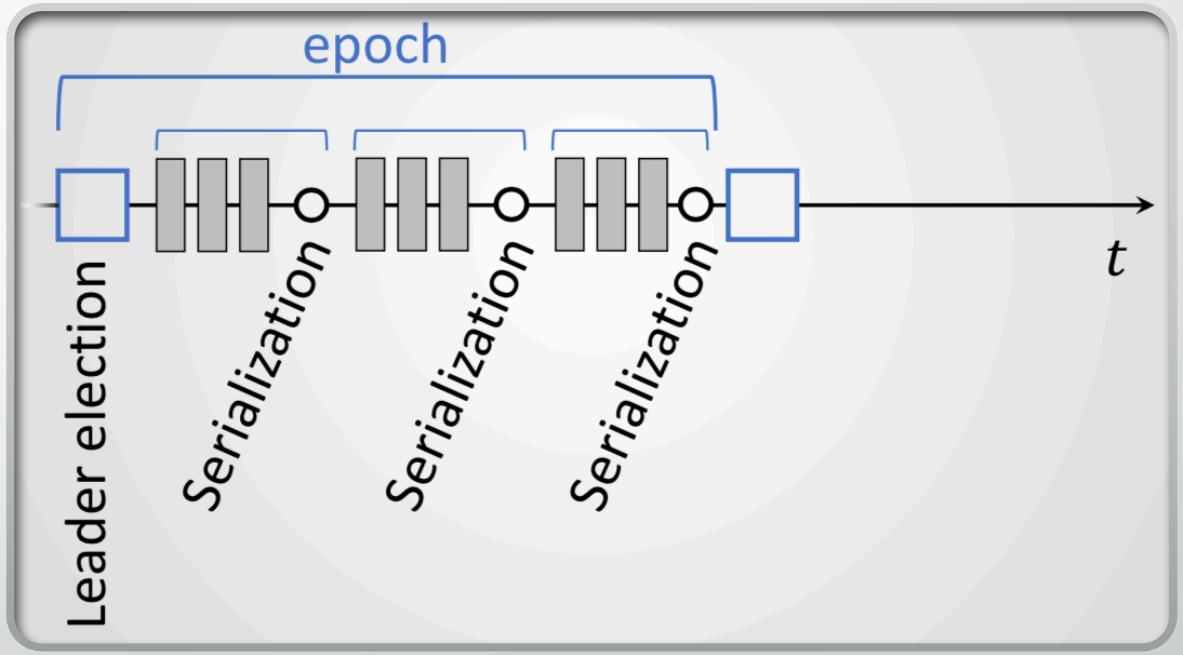
- Increase block size
  - More centralized
- Increase block frequency
  - More forks





Bitcoin-NG

- Leader
  - Puzzle Solver
- Key block
  - Hash values like Bitcoin
  - Proof of Work
  - Pub key of leader
  - No transactions included
- Microblocks
  - Every 10 sec (min)
  - Leader signed header
  - No Proof of Work
  - Contain transactions



Every node reach consensus on header, instead of each transaction

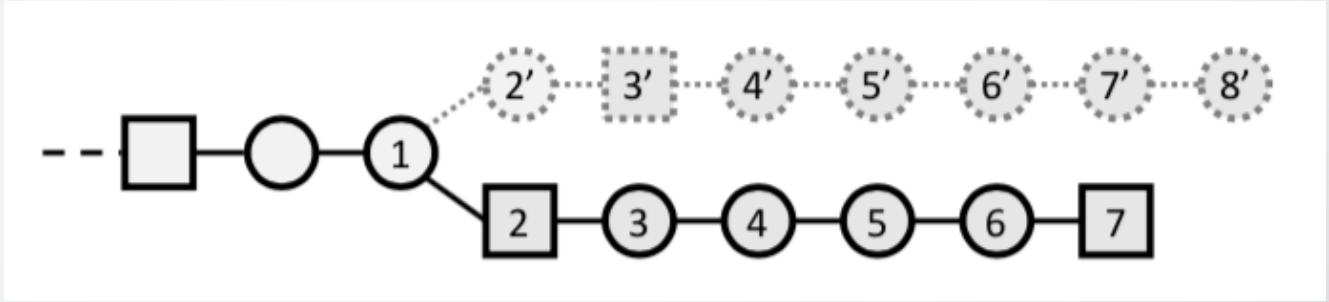
Bitcoin-NG

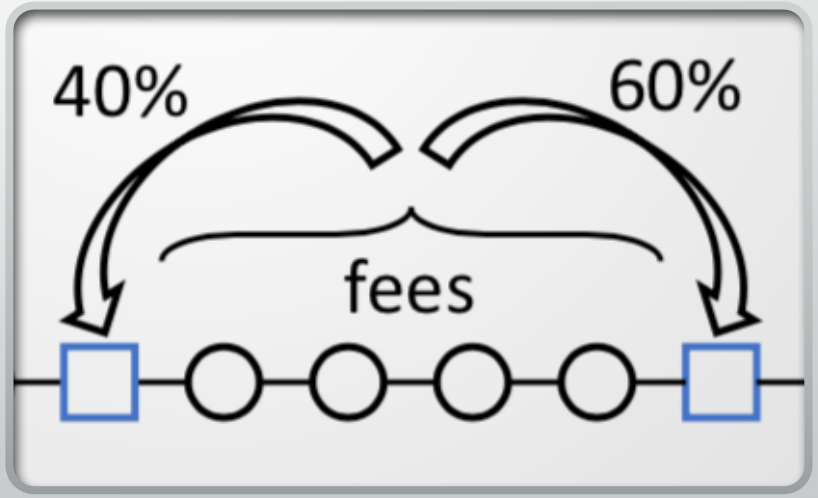
- *WHAT IF* leader maliciously signed invalid transaction?

Every node has the full ability to verify the each transaction as in Bitcoin, therefore if anyone finds out a poisoned leader, they can broadcast this message, and the poisoned leader will lose the position and all its revenue as leader. The node who finds out will get a small portion amount of the revenue as reward.

Bitcoin-NG

- Fork
  - Key block counts
- Fee Distribution
  - 40% - 60%
  - Transaction Inclusion
  - Longest Chain Extension



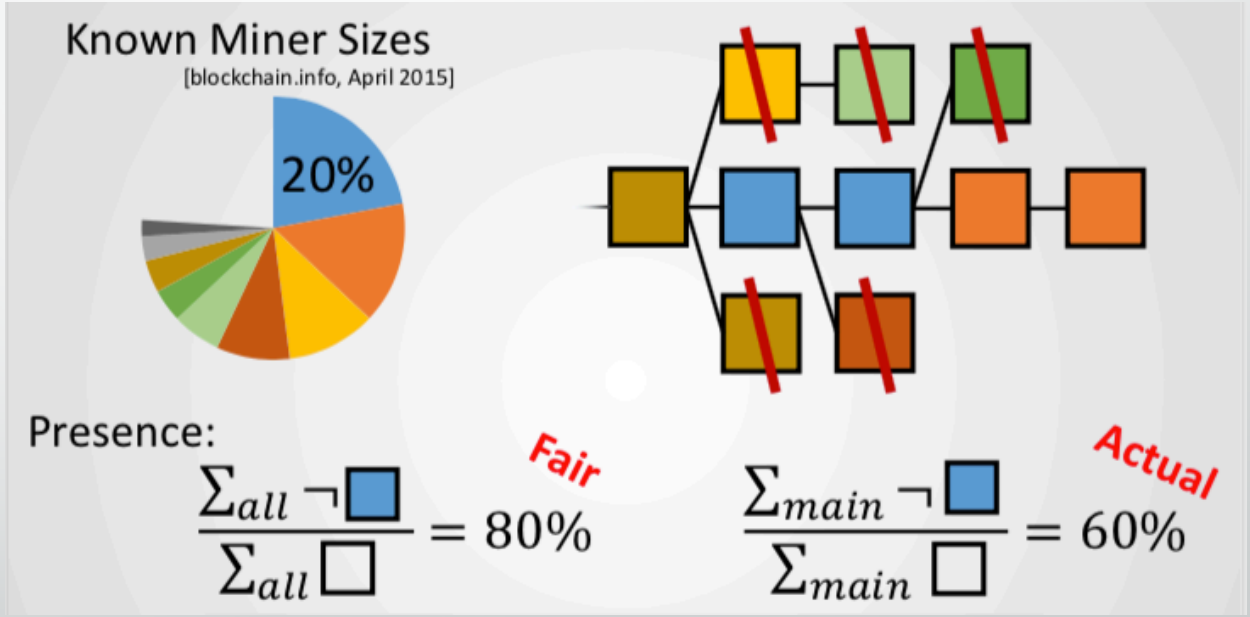


Performance Evaluation – Metrics

- Consensus Delay (Appendix)
- Fairness
- Mining Power Utilization
- Time to Prune & Time to Win (Appendix)

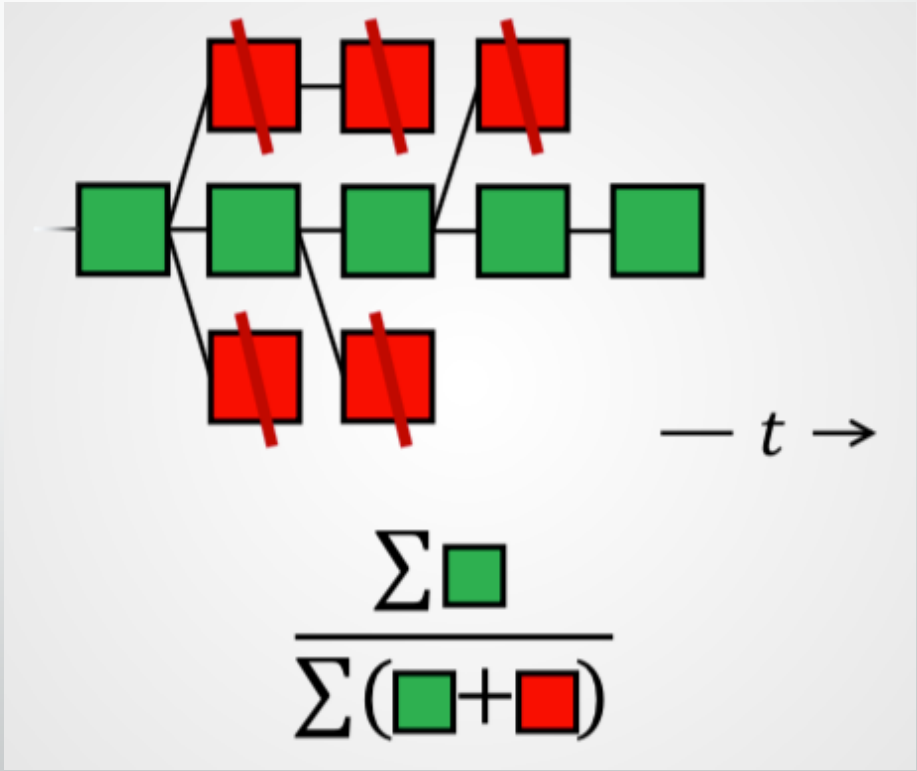
Performance Evaluation - Metrics

- Fairness
  - $\frac{Actual\ Presence}{Fair\ Presence} = \text{Fairness}$
  - Greater the better

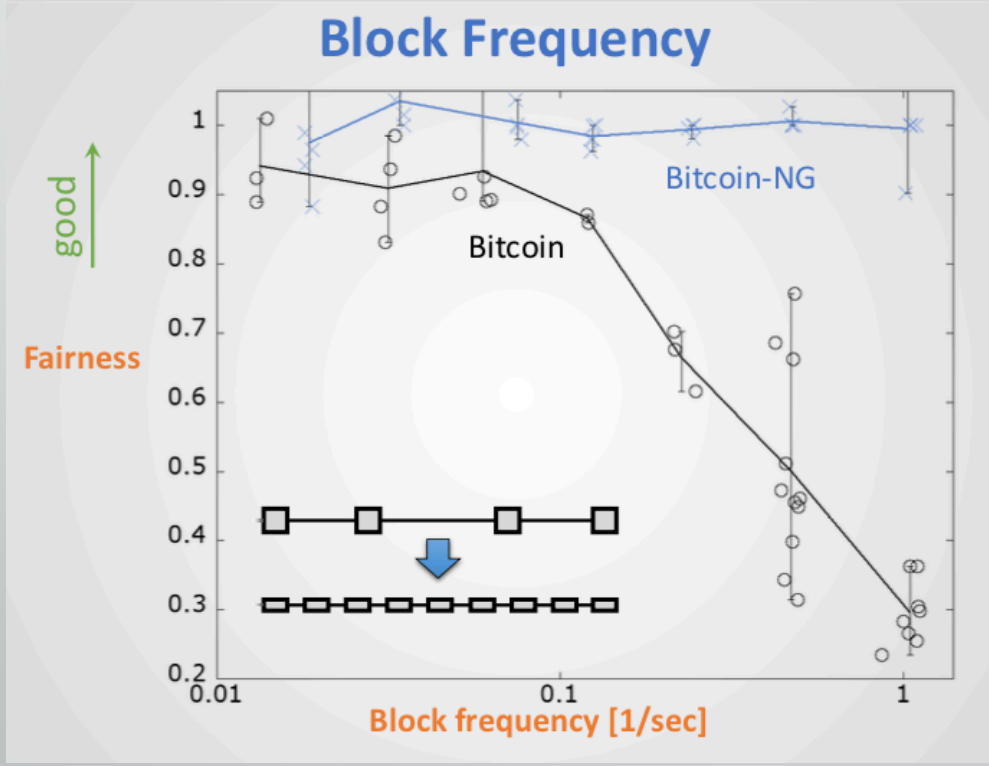


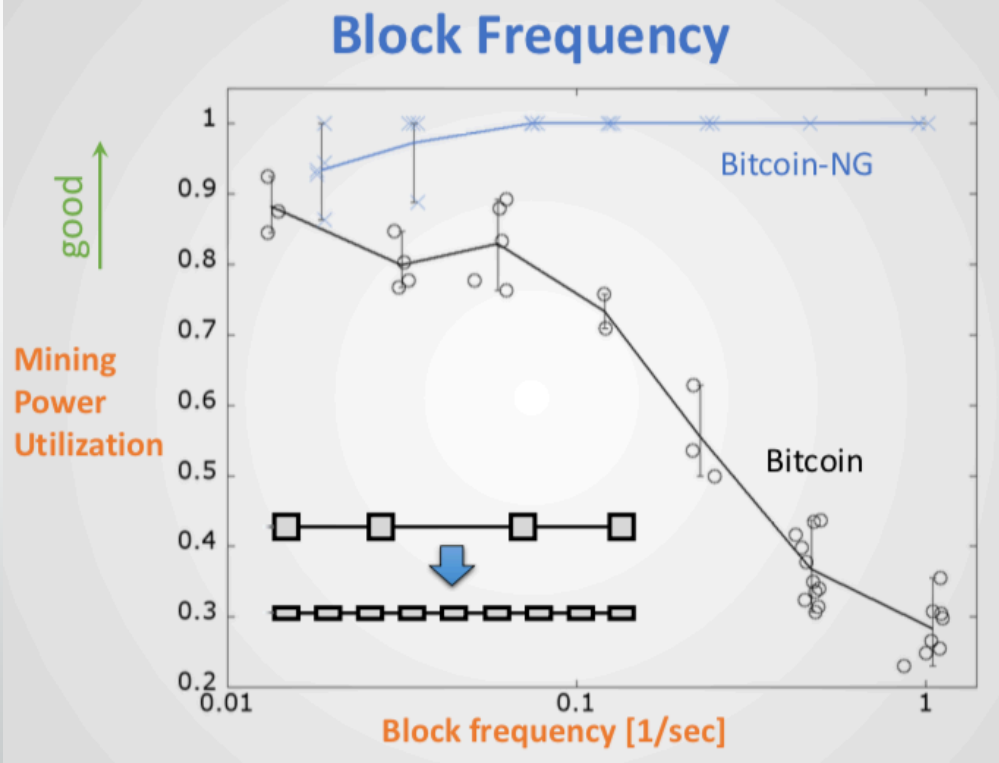
Performance Evaluation - Metrics

- Mining Power Utilization
  - $\frac{Hash\ power\ contributed\ to\ best\ chain}{Total\ hash\ power}$
  - Greater the better

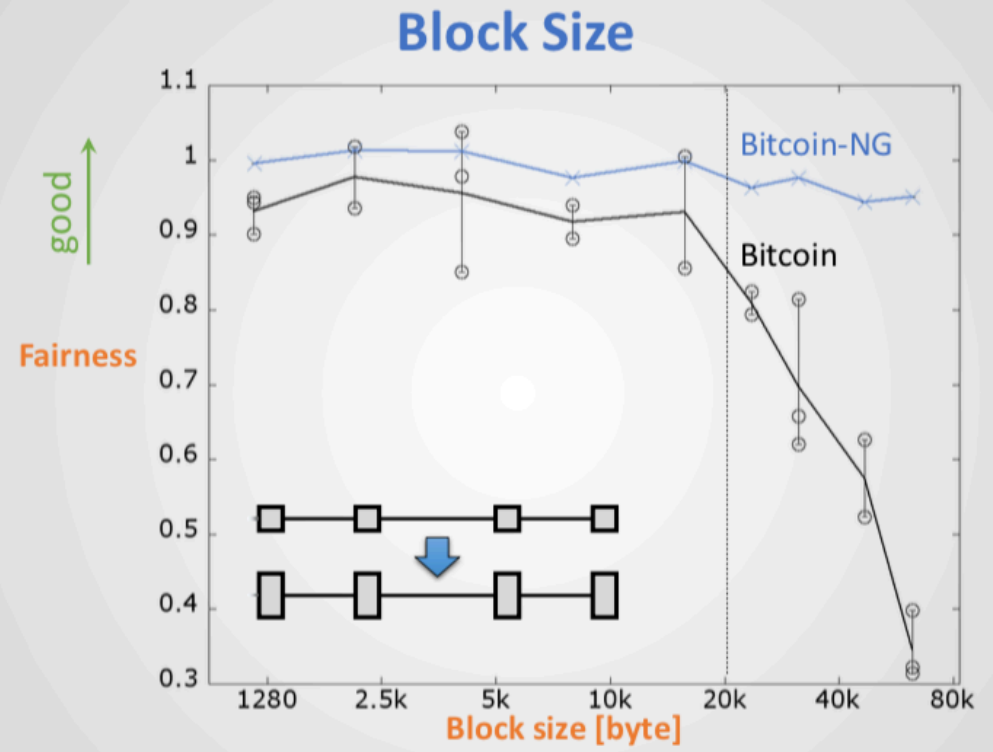


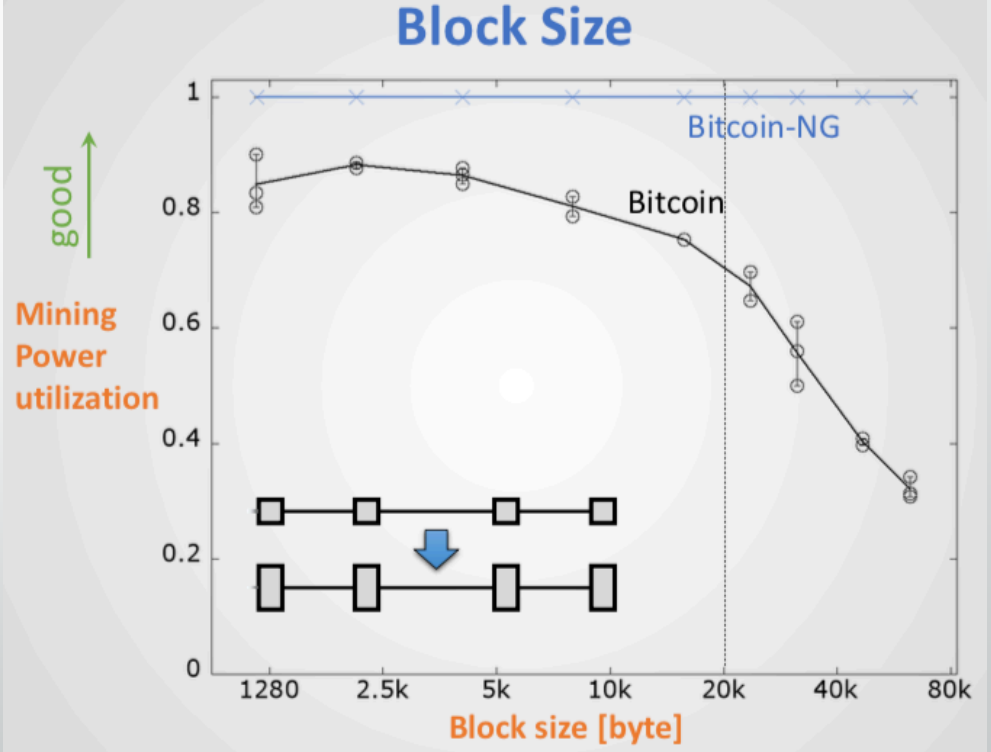
Performance Evaluation – key findout





Performance Evaluation – key findout





Take-away

Bitcoin-NG	Metrics

• PoW key block; leader signed microblock
• Consensus on header
• 40% - 60% fee distribution
  • Transaction Inclusion
  • Longest chain extension
	• Consensus Delay – (% time, % nodes)
	• Fairness
	• Mining Power Utilization
	• Time to Prune & Time to Win

Questions?

References:

- Bitcoin-NG: A Scalable Blockchain Protocol. NSDI'16
- Block size debate: https://en.bitcoin.it/wiki/Block_size_limit_controversy
- Nakamoto consensus: https://blockonomi.com/nakamoto-consensus/
- Majority is not Enough: Bitcoin Mining is Vulnerable. Ittay Eyal, Emin G¨un Sirer 2013

Appendix

- Nakamoto Consensus
  - To achieve BFT in a large scale P2P network.
  - 4 parts:
    - Proof of work
    - Block Selection
    - Scarcity
    - Incentive Structure

Appendix

- Debate on increasing block size:
  - Favor:
    - Need supply meets demand
    - Lower fee -> more appealing to new users
  - Opposed:
    - Higher barrier -> damage to decentralization
    - Possible future damage to censorship-resistant nature

Appendix

- Key Block Header:
  - Ref to the previous block
  - Current Unix time
  - Coinbase transaction reward
  - Target value
  - Nonce field
  - Public key
- Microblocks Header:
  - Ref to the previous block
  - Current Unix time
  - Hash of its ledger entries
  - Signature of the header

Appendix

- Bitcoin-NG fee distribution
  - Transaction Inclusion

$$\overbrace{\alpha \times 100\%}^{\text{Win } 100\%} + \overbrace{(1-\alpha) \times \alpha \times (100\% - r_{\text{leader}})}^{\text{Lose } 100\%\text{, but mine after txn}} < r_{\text{leader}}$$

  - Longest Chain Extension

$$\overbrace{r_{\text{leader}}}^{\text{Place in microblock}} + \overbrace{\alpha(100\% - r_{\text{leader}})}^{\text{Mine next key block}} < \overbrace{100\% - r_{\text{leader}}}^{\text{Mine on existing microblock}}$$

  - α - mining power ratio; $r_{learder}$ – ratio of revenue of the leader; $\alpha < 0.25$

Appendix

- Selfish Mining

Algorithm 1: Selfish-Mine

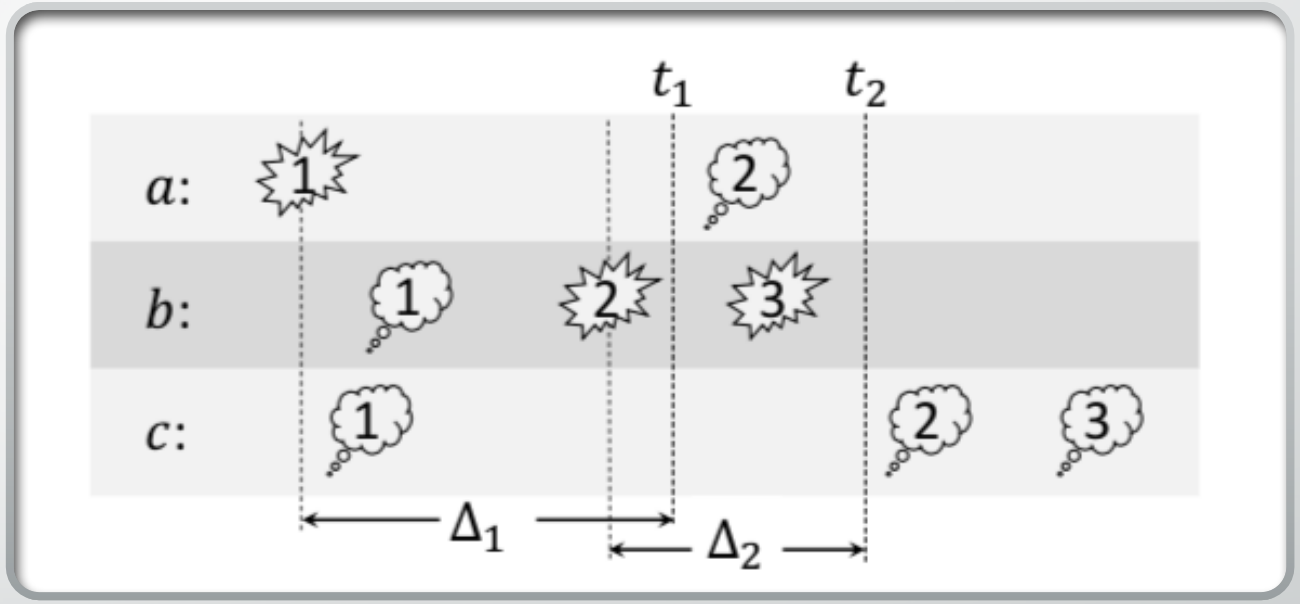
```
1  on Init
2      public chain ← publicly known blocks
3      private chain ← publicly known blocks
4      privateBranchLen ← 0
5      Mine at the head of the private chain.

6  on My pool found a block
7      Δ_prev ← length(private chain) − length(public chain)
8      append new block to private chain
9      privateBranchLen ← privateBranchLen + 1
10     if Δ_prev = 0 and privateBranchLen = 2 then        (Was tie with branch of 1)
11         publish all of the private chain               (Pool wins due to the lead of 1)
12         privateBranchLen ← 0
13     Mine at the new head of the private chain.

14 on Others found a block
15     Δ_prev ← length(private chain) − length(public chain)
16     append new block to public chain
17     if Δ_prev = 0 then
18         private chain ← public chain                   (they win)
19         privateBranchLen ← 0
20     else if Δ_prev = 1 then
21         publish last block of the private chain        (Now same length. Try our luck)
22     else if Δ_prev = 2 then
23         publish all of the private chain               (Pool wins due to the lead of 1)
24         privateBranchLen ← 0
25     else                                               (Δ_prev > 2)
26         publish first unpublished block in private block.
27     Mine at the head of the private chain.
```

Performance Evaluation - Metrics

- Consensus Delay
  - Denoted as (ε, δ)
  - ε – time ratio; δ – node ratio
  - (50%, 90%) = 10 sec; means 90% of the time, 50% of the nodes agree on the state of machine 10 seconds ago.



Performance Evaluation - Metrics

- Time to Prune & Time to Win
  - Time to Prune
    - Learn the 1st branch block -> prune branch
  - Time to Win
    - 1st main branch block -> last side branch block