PoW Alternatives and scaling PoW

PoW problem?

- Wastes LOTS of energy !
- ASICs dominate mining
  from single manufacturer

A better PoW

- Asics resistance:
  - ASIC processor optimised for sha256.
  - Idea: Use more complex function.
    - e.g. complex computation, similar to cpu workload.
    - e.g. I.O. (memory) bound function

premising

- Usefull PoW:
  - use computation to do something useful eg.
    - find prime number

- protein folding

Problem:

Properties a PoW must have

- Adjustable difficulty
- Fast verification
- Progress freedom
  - Not possible to make progress towards solution.
    i.e. random trials

Exp:

PoW alternatives

Problem: Distribute "voting" power in an

anonymous system with sybils.

PoW: Voting power based on computing power.

Satoshi: „one CPU one vote"

Proof of Storage

Voting power based on storage capacity

one disc one vote

Proof: Similar to merkle proof for randomly chosen segment.

Problem: What to store?

- same data on every disc
  - → how to distinguish discs?
  - → avoid on demand download
- different data
  - → management overhead.

what is usefull Proof of Storage?

Less energy?

Proof of stake (proof of investment)

Use crypto currency as scarce resource

one dollar one vote

~> separate lecture

Scaling throughput

(1 Th ≈ 86000 sec)

visa / mastercard: several x 100 000 tx/sec

bitcoin 7 tx/sec

Scaling trench reparametrization

Parameters

- Block size
- Block delay

Small delay; $\rightsquigarrow$ larger p $\rightsquigarrow$ more forks

$$P[\text{Fork}] = 1-(1-p)^{\delta}$$

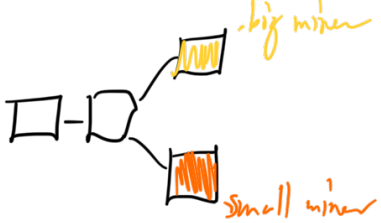
Big size $\rightsquigarrow$ slow sending $\rightsquigarrow$ larger δ

$\rightsquigarrow$ more forks

Increasing throughput gives more forks.

a) o bad for security

b) o bad for small miners (unfair)

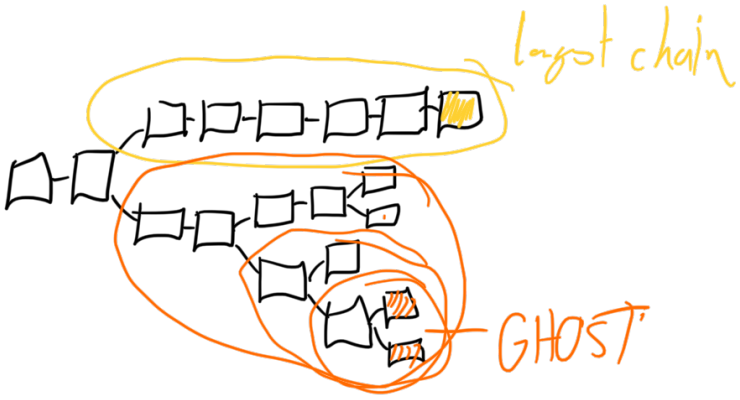


GHOST

Greedy heaviest-observed subtree

Avoid a) by changing longest chain rule.

GHOST rule: select heaviest subtree
(most blocks)

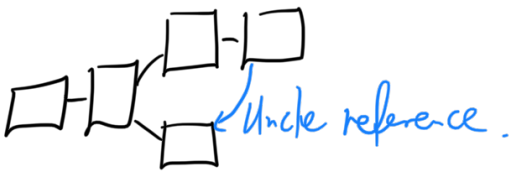


~> used in ethereum

Inclusive blockchain

Uncle blocks

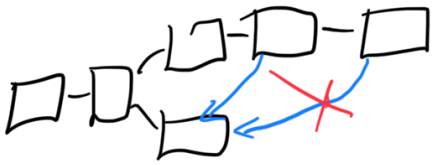
Avoid b) by giving some reward to lost blocks



Uncles nodes can reference uncles.

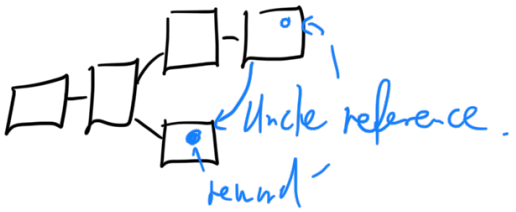
- Uncle has smaller height.

- Uncle is new



Uncle gets fraction of block reward.

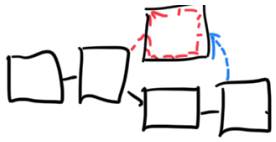
Including miner (nephew) gets a small reward



Uncle rewards create more money.

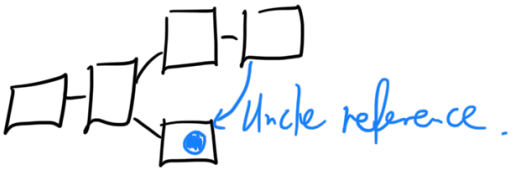
→ Need to adjust difficulty.

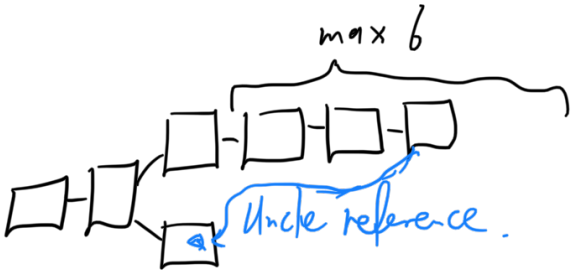
Uncle rewards may reduce risk in selfish mining.



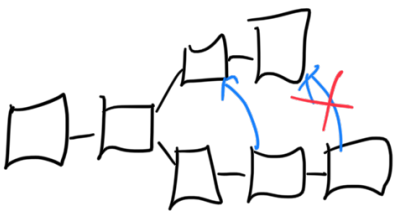
Ethereum uncles

- Smaller reward if nephew is far away



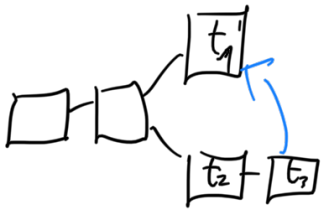


Only children of ancestors



Uncles for throughput

- Can execute transactions included only in uncles. (If not conflicting)



Execution order t_2 t_1 t_3

Not done in Ethereum.

DAG