Hybrid Consensus Algorithm

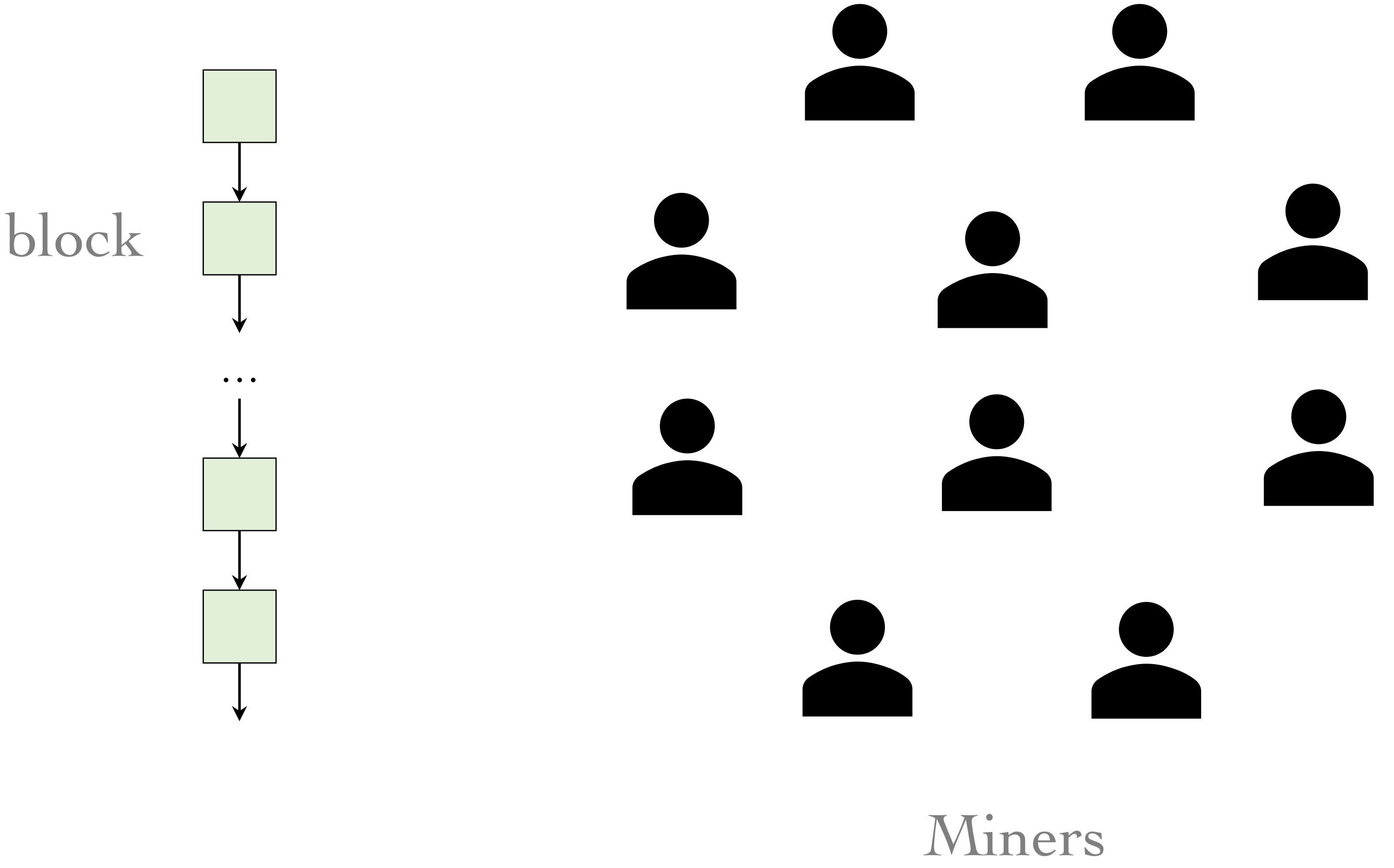
Rujia Li

September 16, 2022

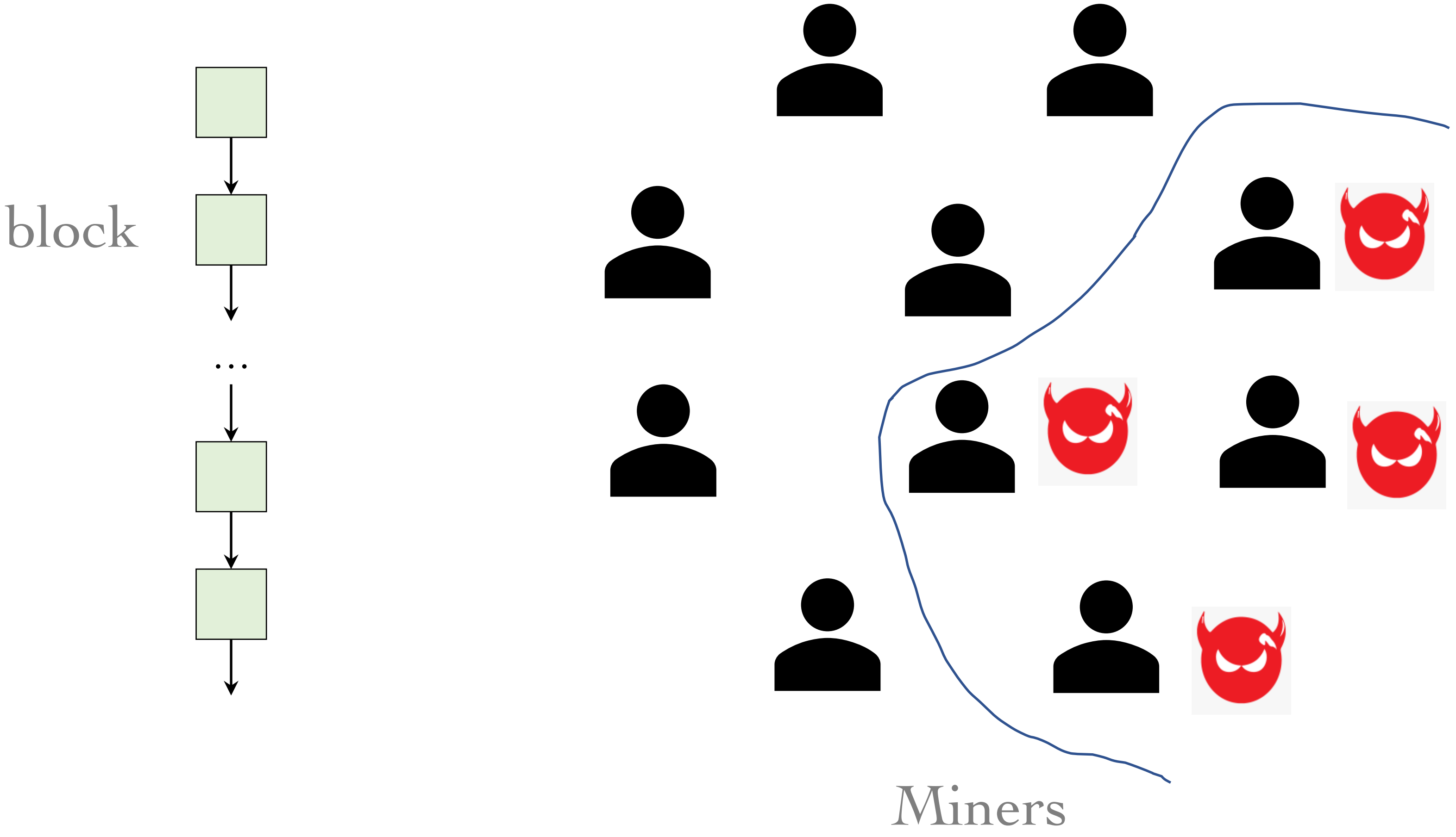
rujia.uk@gmail.com

Some materials are provided by Xinrui Zhang and Qin Wang

Distributed Systems

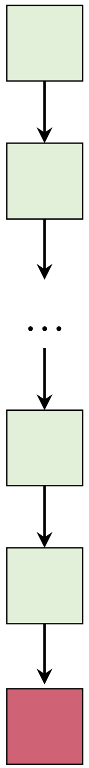


Network Partition



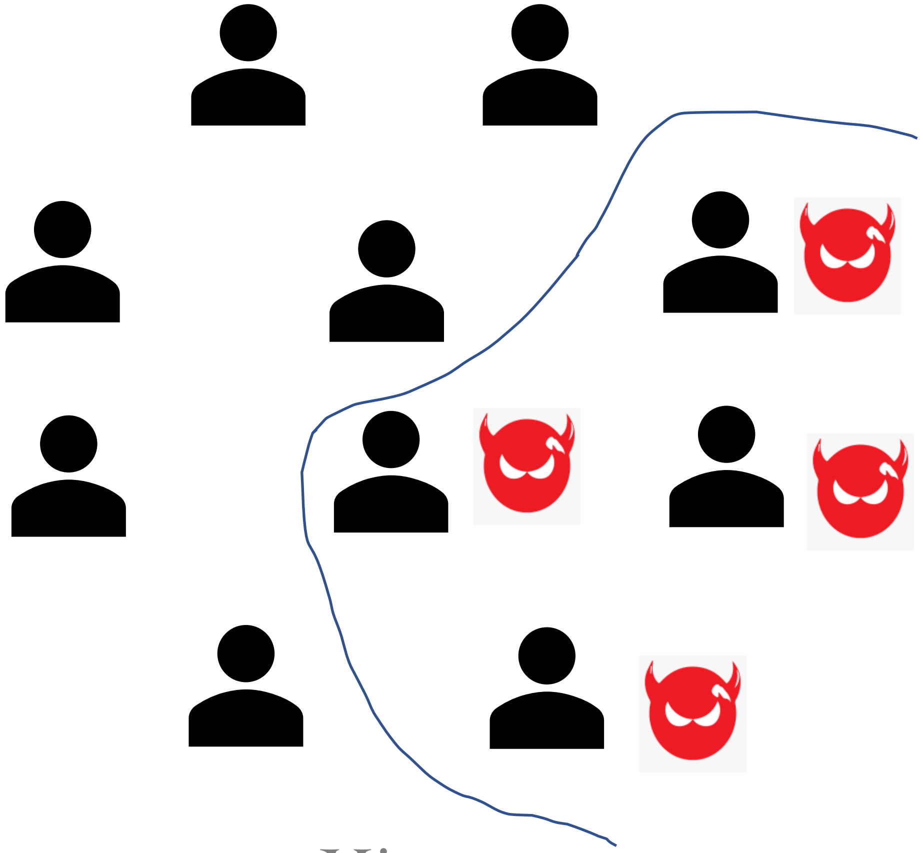
Ideal Consensus Algorithm

block



Keep working !

Available under dynamic participations



Miners

Security Properties

- Liveness

  Valid transactions eventually be accepted

- Safety

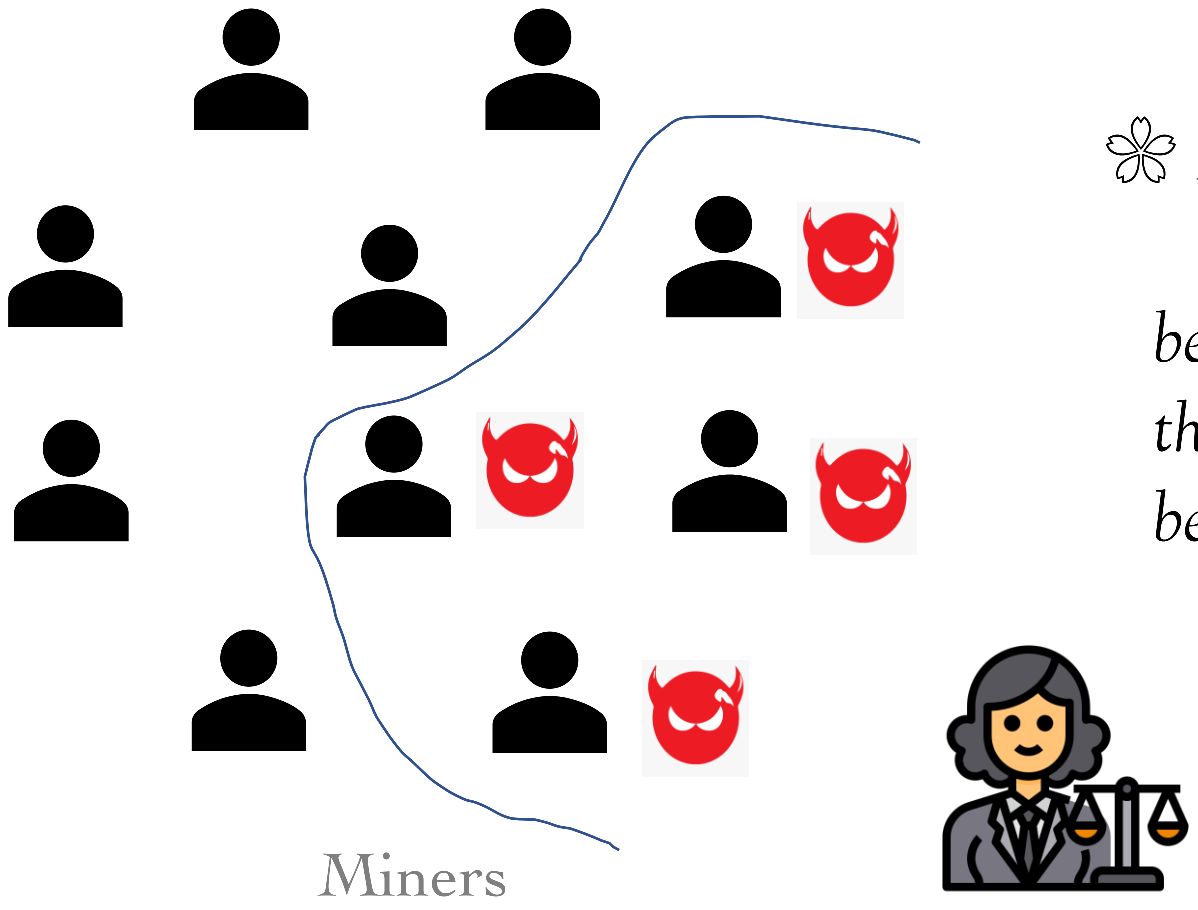
  Honest miners will agree on the same sequence of values

Security Properties

- Liveness
  Valid transactions eventually be accepted
- Safety
  Honest miners will agree on the same sequence of values

- Availability
  Still <u>live</u> even if a fraction of miners leave
- Finality (Consistency)
  Still <u>safe</u> even if a fraction of miners leave

Accountable Safety



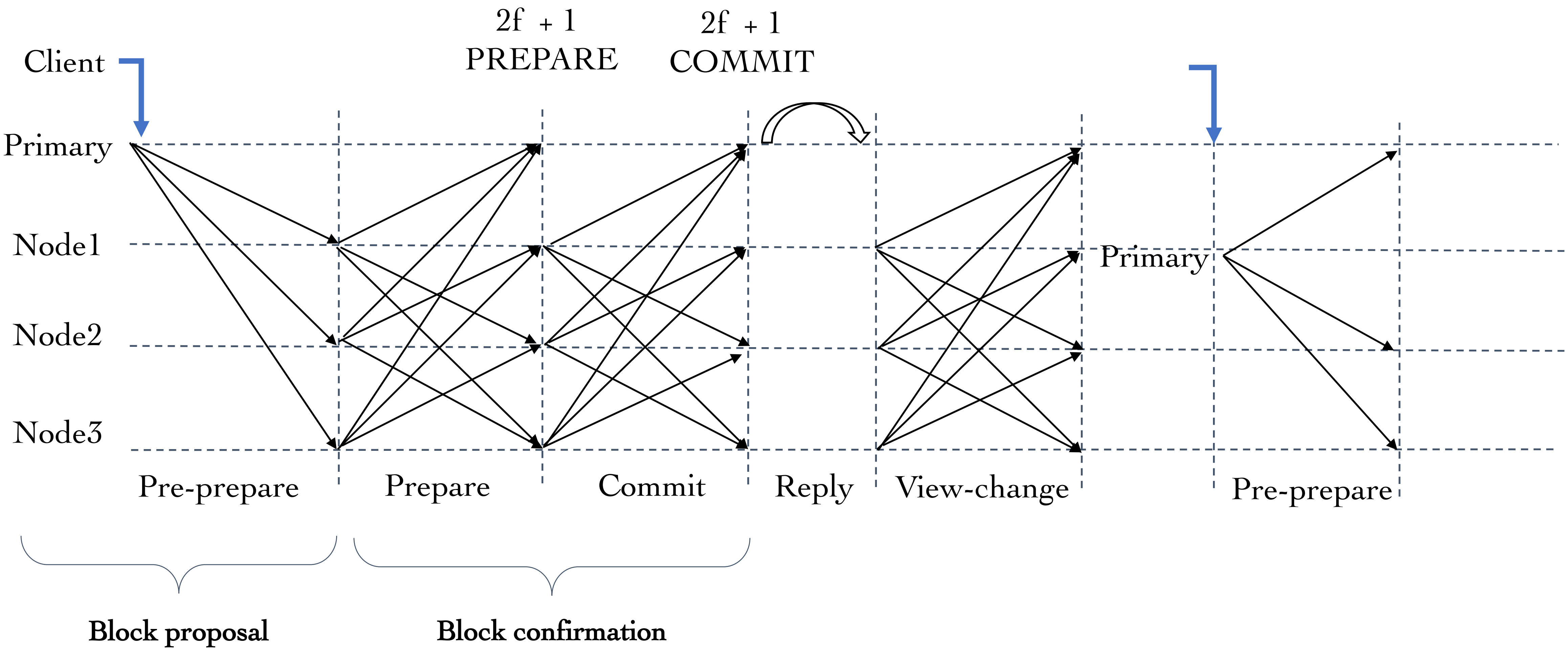
❀ Accountable Safety

Two conflicting value cannot both be finalized. If a safe violation occurs, then the malicious participants can be identified.

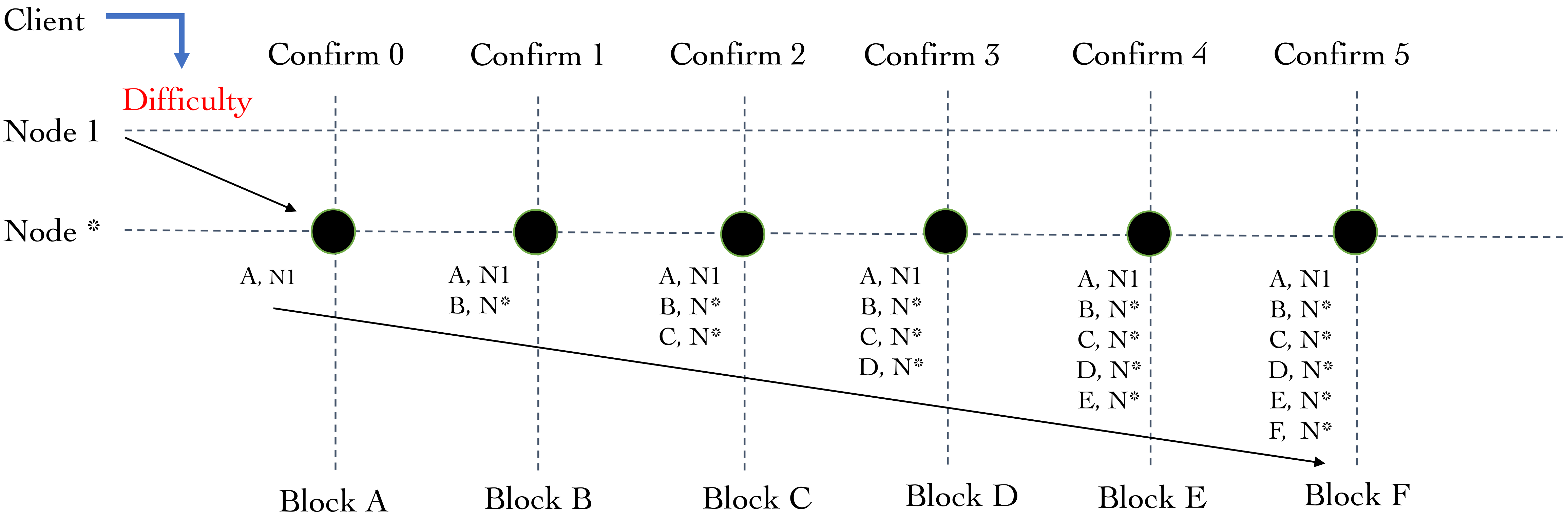
Joachim Neu, Ertem Nusret Tas, and David Tse. The availability-accountability dilemma and its resolution via accountability gadgets. International Conference on Financial Cryptography and Data Security (FC), 2022.

PBFT and PoW

Practical Byzantine Fault Tolerance (PBFT)



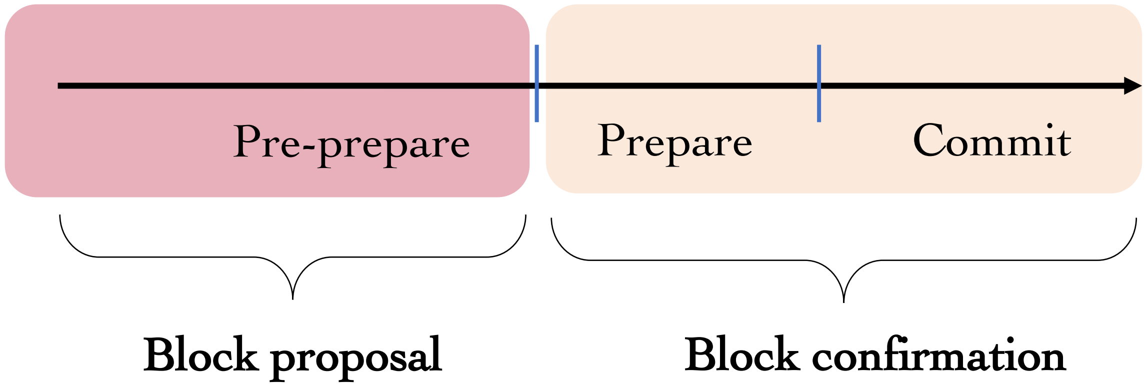
Proof of Work (POW)



Comparison

BFT and Its Variants

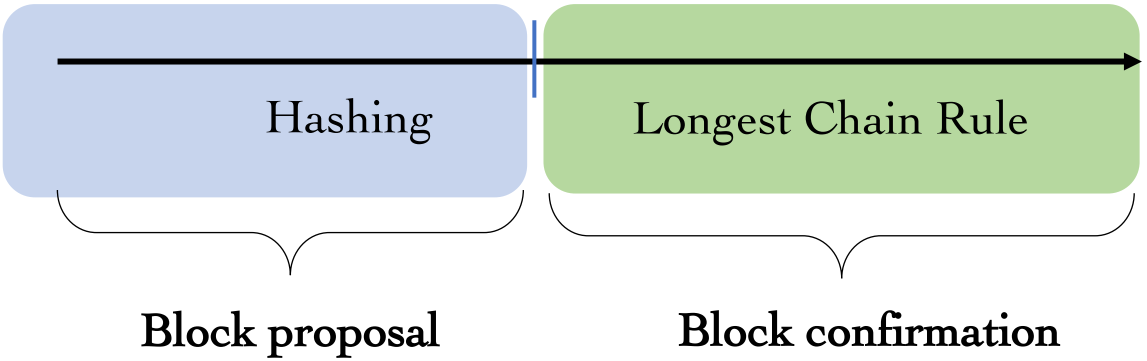
- Permissioned
- Leader-based
- Communication-based
- Safety over liveness



Deterministic finality

POW and Its Variants

- Permissionless
- Leaderless
- Computation-based
- Liveness over safety

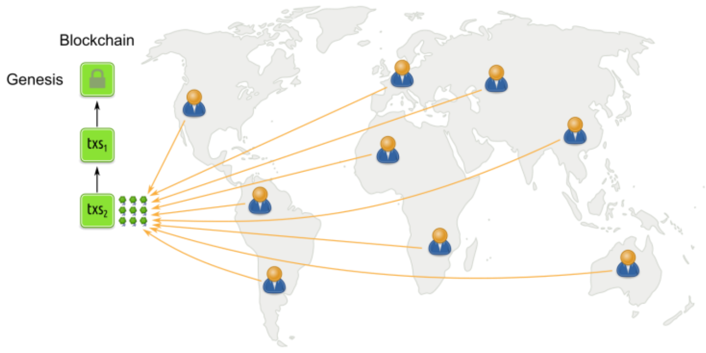


Probabilistic finality

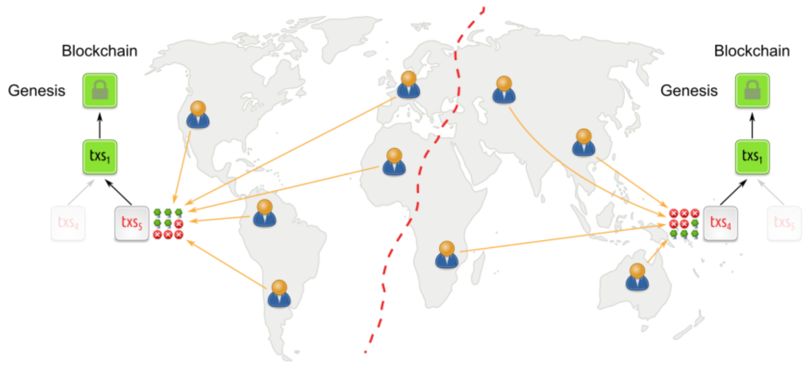
Safety over Liveness

BFT and Its Variants

Image source, https://decentralizedthoughts.github.io/2020-11-01-ebb-and-flow-protocols-a-resolution-of-the-availability-finality-dilemma/



under normal conditions



under network partition

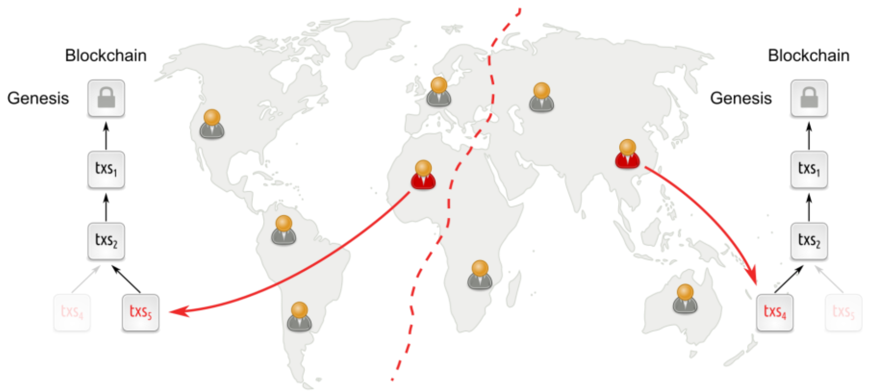
under low participation

Liveness over Safety

POW and Its Variants

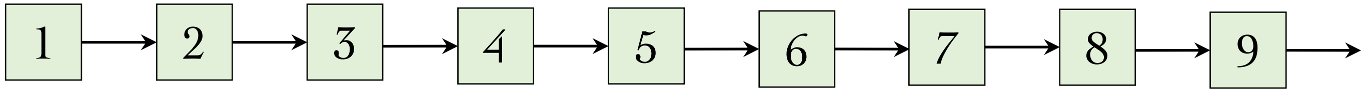
under normal conditions

under network partition



under low participation

“Perfect” Public Ledger



Impossible !!!

- Availability

  Still live even if a fraction of miners leave

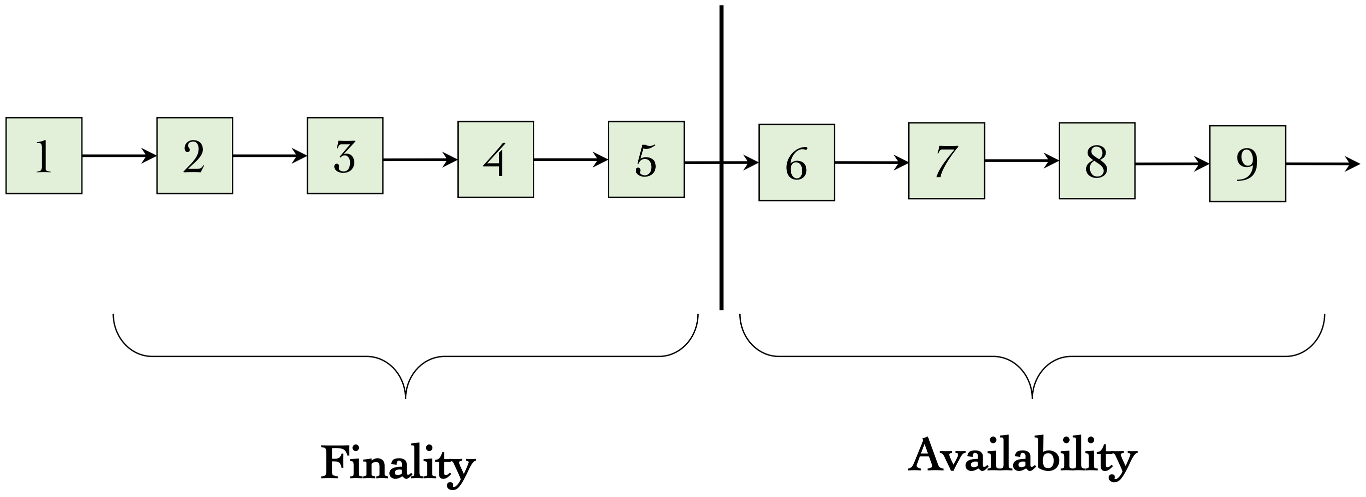
- Finality

  Still safe even if a fraction of miners leave

CAP theorem states that during a network partition, a distributed system must make a choice between availability (liveness) and finality (safety); it cannot offer both.

Gilbert, Seth, and Nancy Lynch. "Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services." *Acm Sigact News* 33, no. 2 (2002): 51-59.

Nested Public Ledger



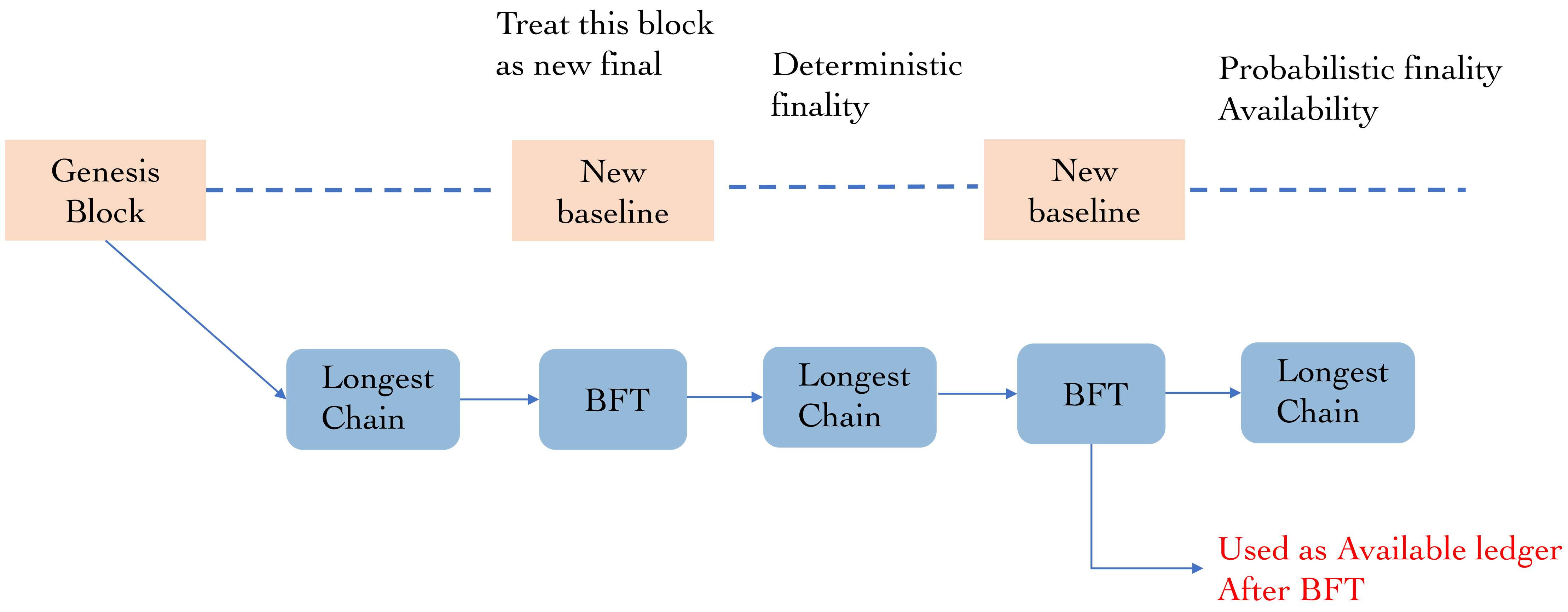
Tradeoff

Accountable safe
Live if network is not partitioned
Enough nodes joining

Safe + live
Under dynamic miner
If network is not partitioned

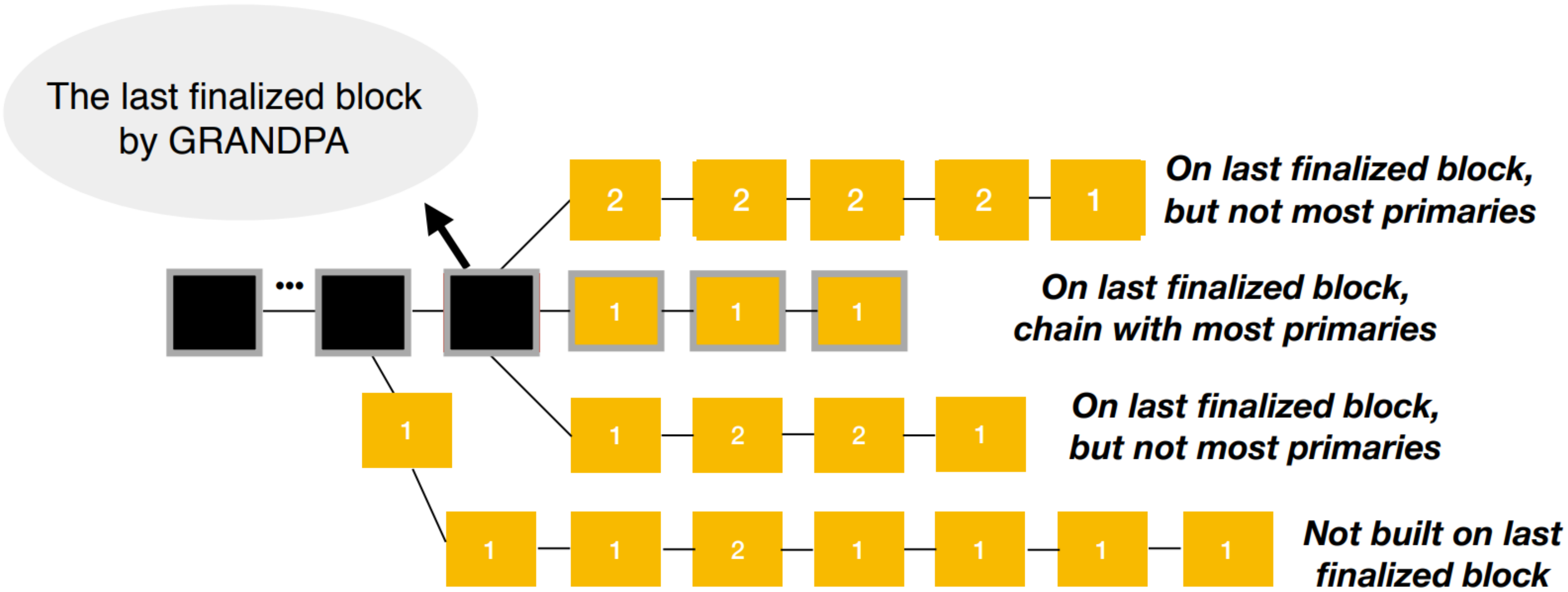
Neu, Joachim, Ertem Nusret Tas, and David Tse. "Ebb-and-flow protocols: A resolution of the availability-finality dilemma." In *2021 IEEE Symposium on Security and Privacy (SP)*, pp. 446-465. IEEE, 2021.

Design Philosophy (1)



GRANDPA

GHOST-based Recursive ANcestor Deriving Prefix Agreement (GRANDPA)

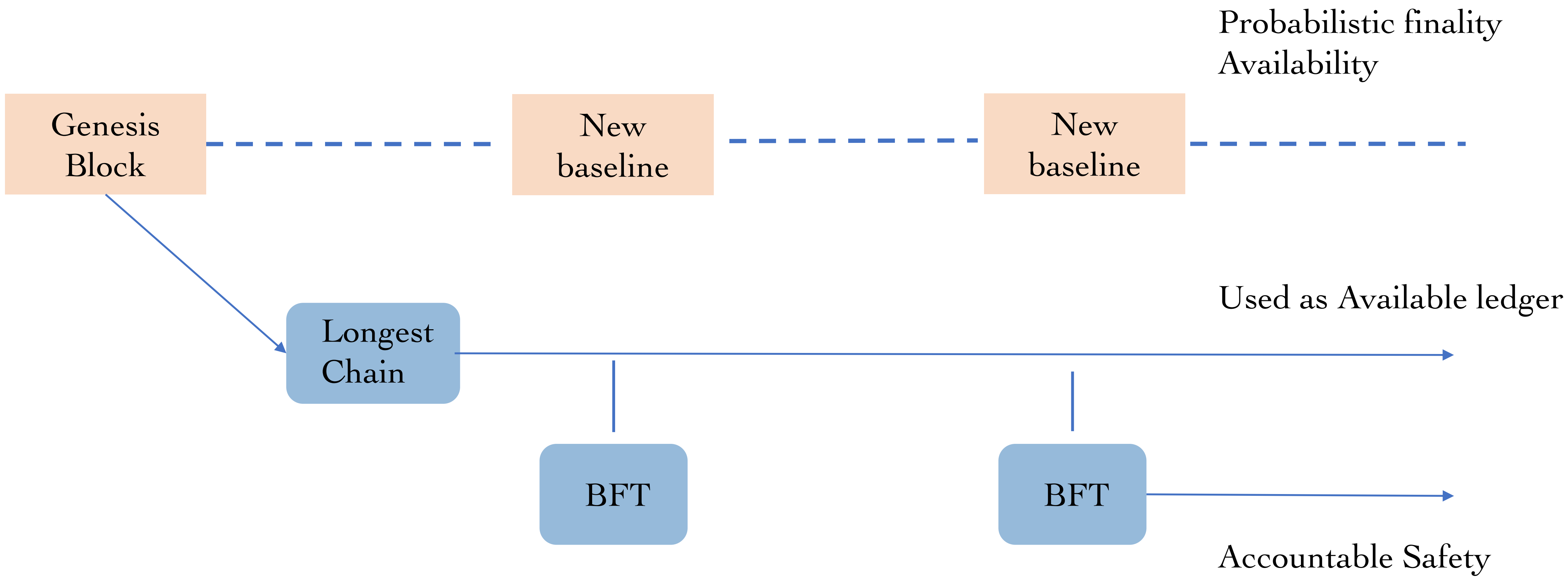


Longest chain with most primaries on last finalized GRANDPA block

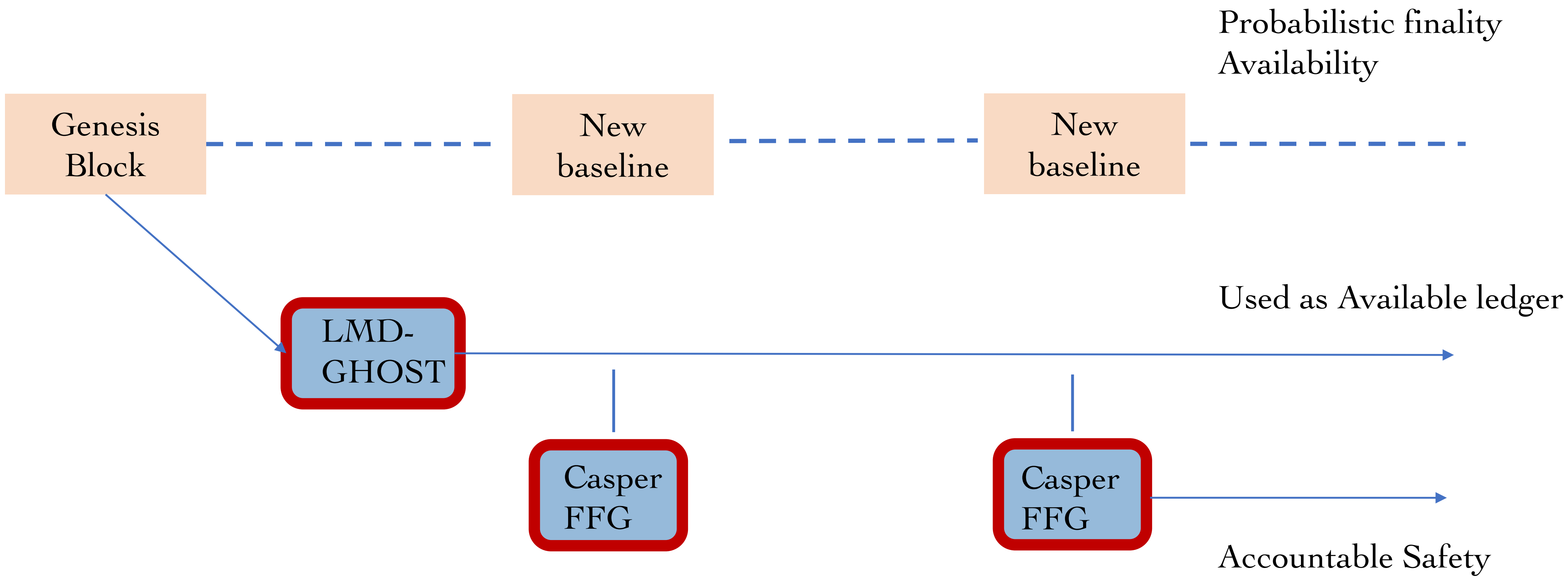
Stewart, Alistair, and Eleftherios Kokoris-Kogia. "GRANDPA: a Byzantine finality gadget." arXiv preprint arXiv:2007.01560 (2020).

Ethereum Proof of Stake

Design Philosophy (2)



Ethereum Proof of Stake

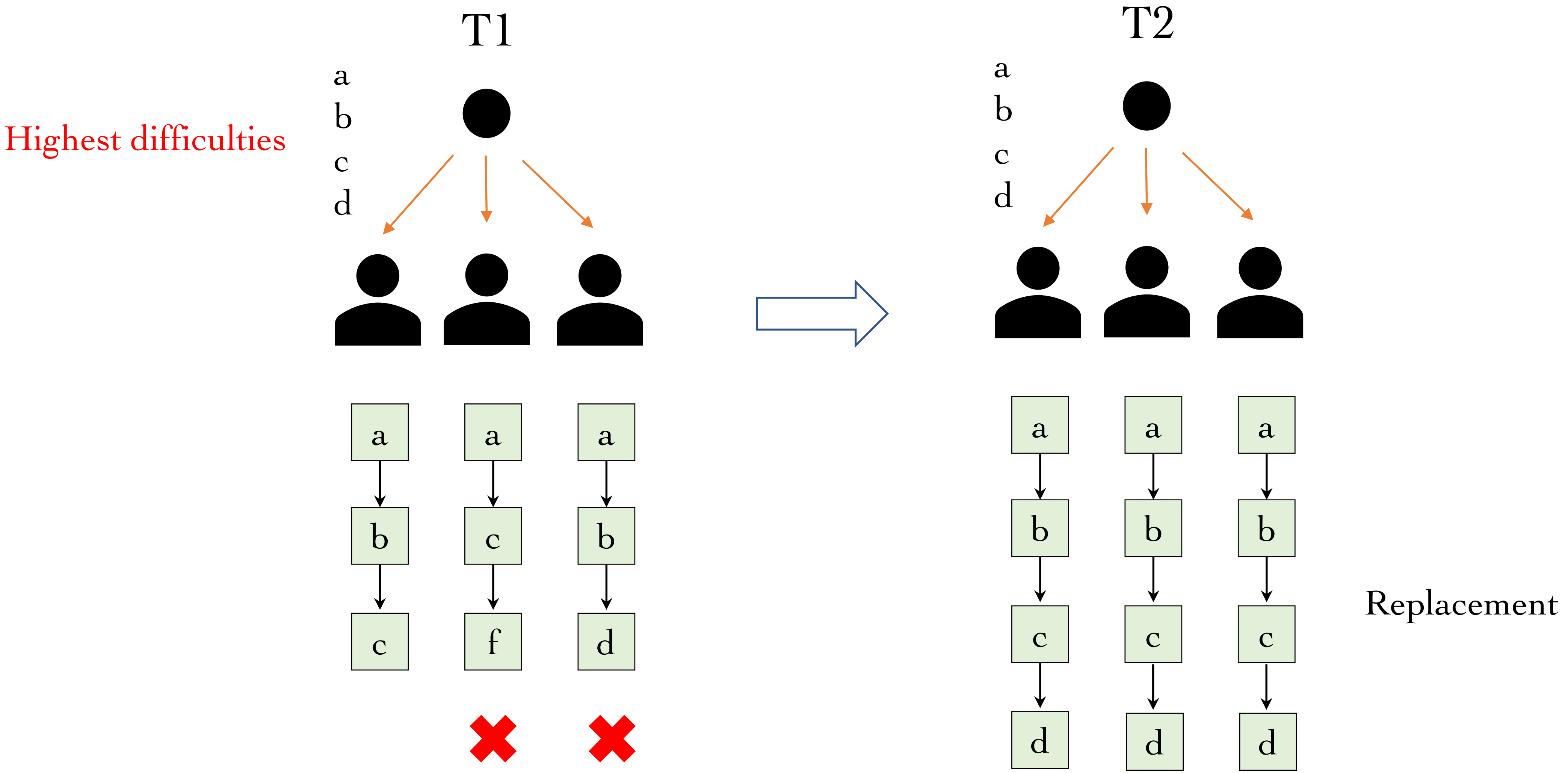


LMD-GHOST

2008	2015	2020	2022
Longest Chain	GHOST	LMD-GHOST	Goldfish
Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." Decentralized Business Review (2008): 21260.	Sompolinsky, Yonatan, and Aviv Zohar. "Secure high-rate transaction processing in bitcoin." In *International conference on financial cryptography and data security*, pp. 507-527. Springer, Berlin, Heidelberg, 2015.	Buterin, Vitalik, Diego Hernandez, Thor Kamphefner, Khiem Pham, Zhi Qiao, Danny Ryan, Juhyeok Sin, Ying Wang, and Yan X. Zhang. "Combining GHOST and casper." arXiv preprint arXiv:2003.03052 (2020).	D'Amato, Francesco, Joachim Neu, Ertem Nusret Tas, and David Tse. "No More Attacks on Proof-of-Stake Ethereum?." *arXiv preprint arXiv:2209.03255* (2022).

Latest Message Driven GHOST

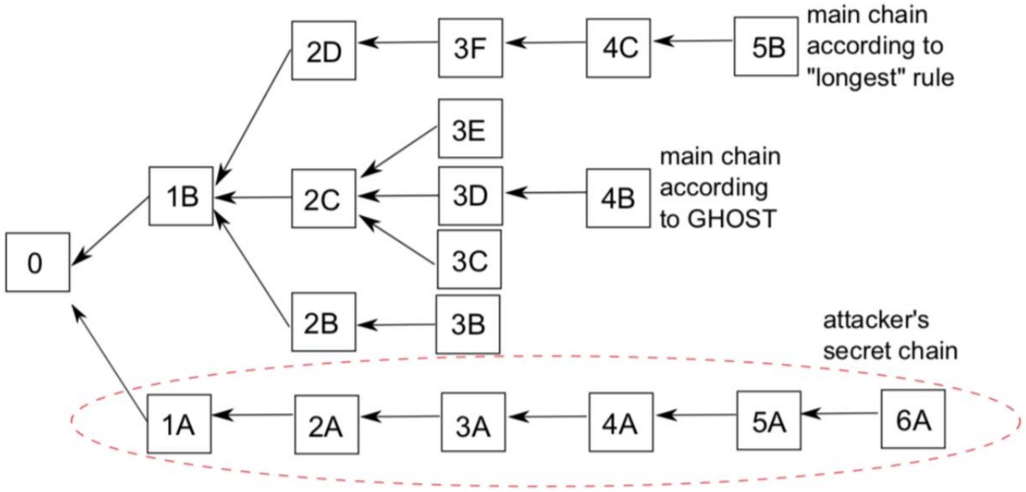
Longest Chain Rule



Greedy Heaviest Observer SubTree (GHOST)

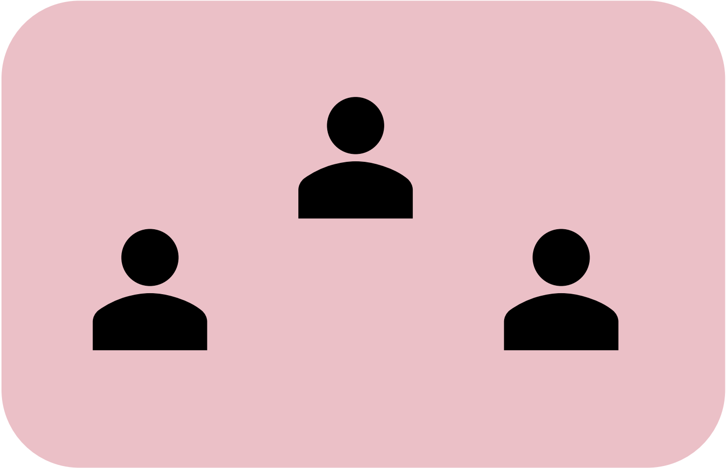
```
// If the total difficulty is higher than our known, add it to the canonical chain
// Second clause in the if statement reduces the vulnerability to selfish mining.
// Please refer to http://www.cs.cornell.edu/~ie53/publications/btcProcFC.pdf
if externTd.Cmp(localTd) > 0 || (externTd.Cmp(localTd) == 0 && mrand.Float64() < 0.5) {
        // Delete any canonical number assignments above the new head
        for i := number + 1; ; i++ {
                hash := GetCanonicalHash(hc.chainDb, i)
                if hash == (common.Hash{}) {
                        break
                }
                DeleteCanonicalHash(hc.chainDb, i)
        }
        // Overwrite any stale canonical number assignments
        var (
                headHash   = header.ParentHash
                headNumber = header.Number.Uint64() - 1
                headHeader = hc.GetHeader(headHash, headNumber)
        )
        for GetCanonicalHash(hc.chainDb, headNumber) != headHash {
                WriteCanonicalHash(hc.chainDb, headHash, headNumber)

                headHash = headHeader.ParentHash
                headNumber = headHeader.Number.Uint64() - 1
                headHeader = hc.GetHeader(headHash, headNumber)
        }
        // Extend the canonical chain with the new header
        if err := WriteCanonicalHash(hc.chainDb, hash, number); err != nil {
                log.Crit("Failed to insert header number", "err", err)
        }
```



Sompolinsky, Yonatan, and Aviv Zohar. "Secure high-rate transaction processing in bitcoin." In *International conference on financial cryptography and data security*, pp. 507-527. Springer, Berlin, Heidelberg, 2015.

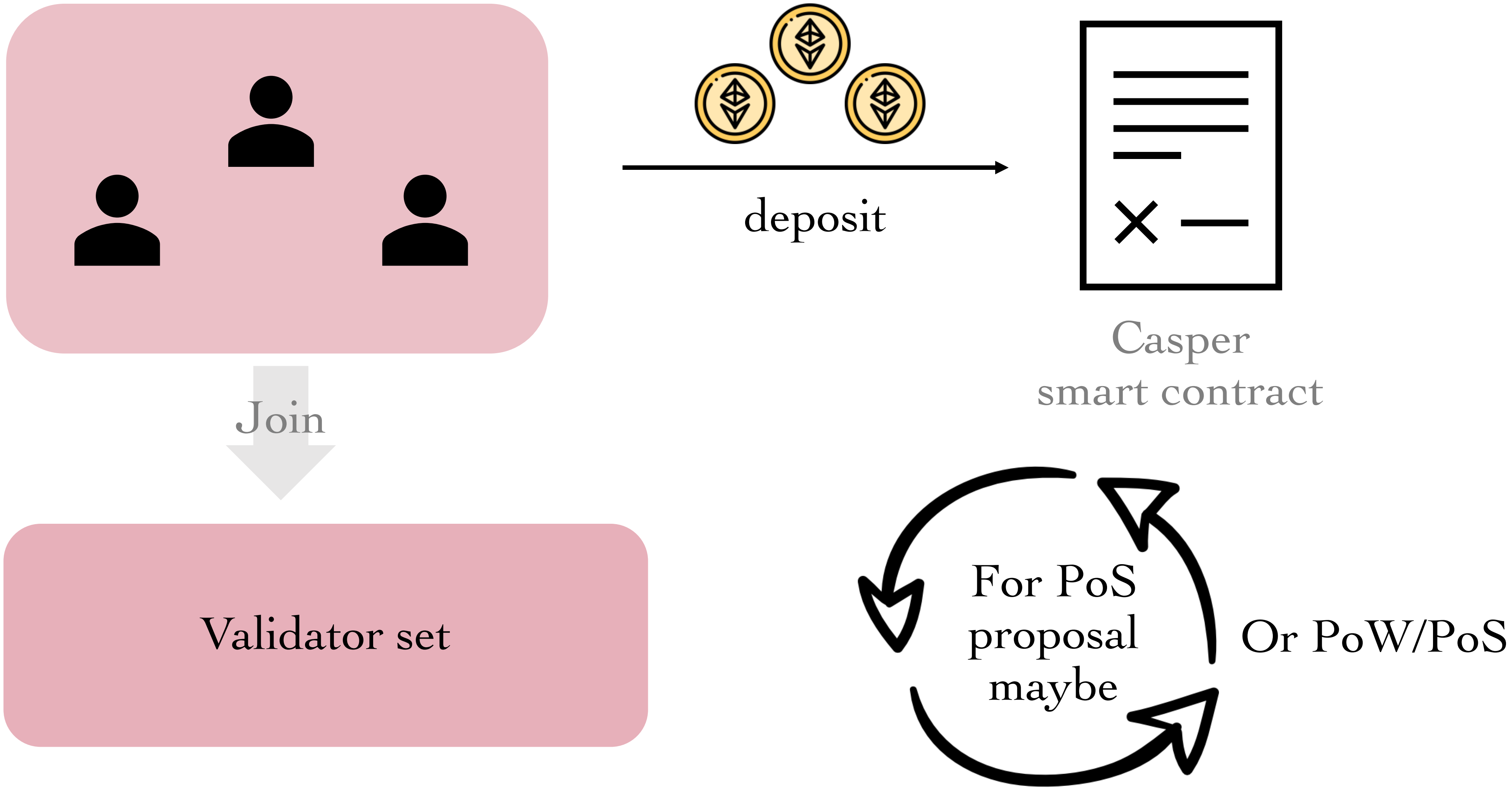
Casper FFG



Who can propose and vote ?
Vote for what ?
Voting rule
Dynamic participants
Possible attacks

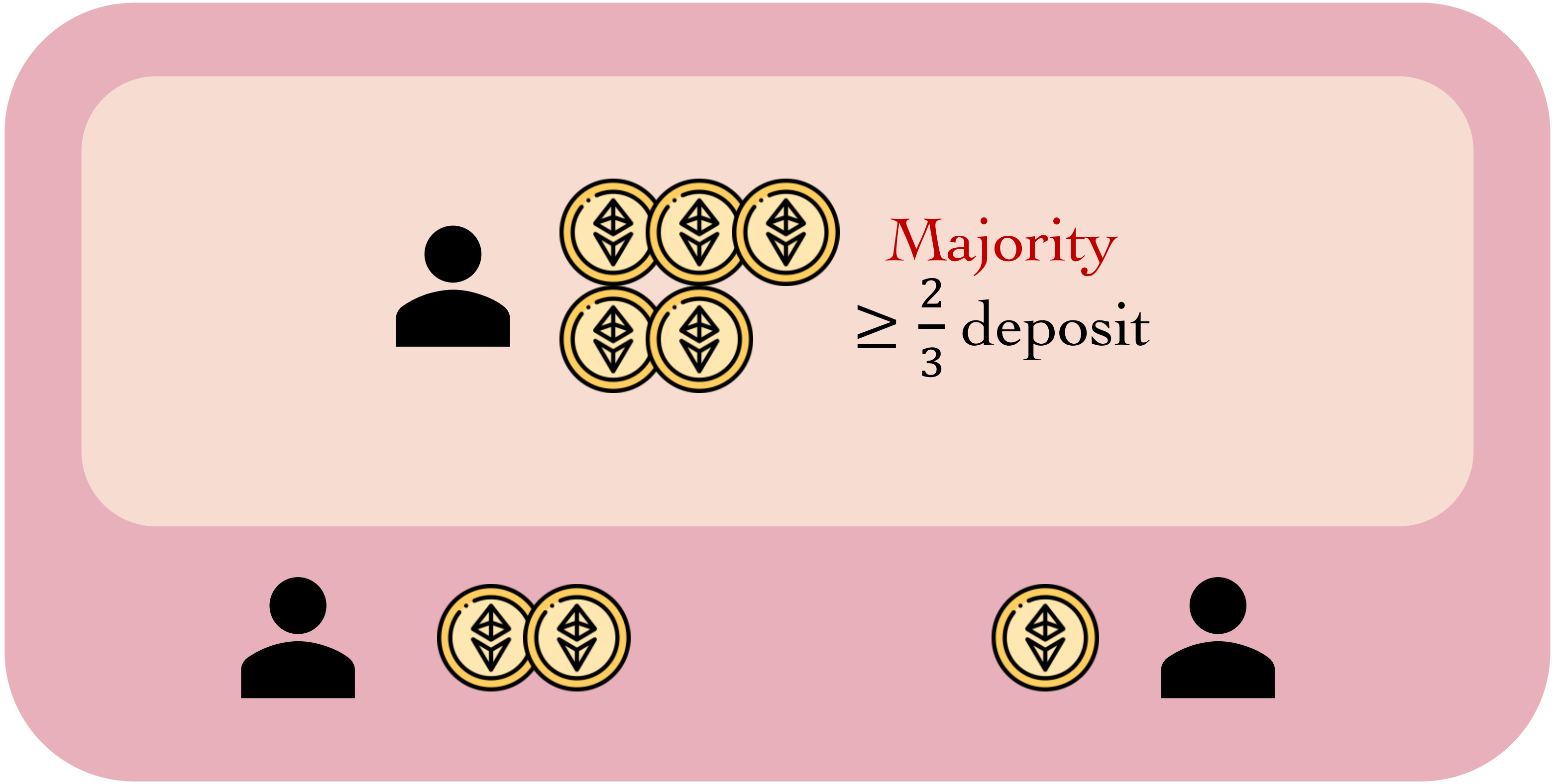
Buterin, Vitalik, and Virgil Griffith. "Casper the friendly finality gadget." *arXiv preprint arXiv:1710.09437* (2017).

Validators



deposit
Casper
smart contract
Join
Validator set
For PoS
proposal
maybe
Or PoW/PoS

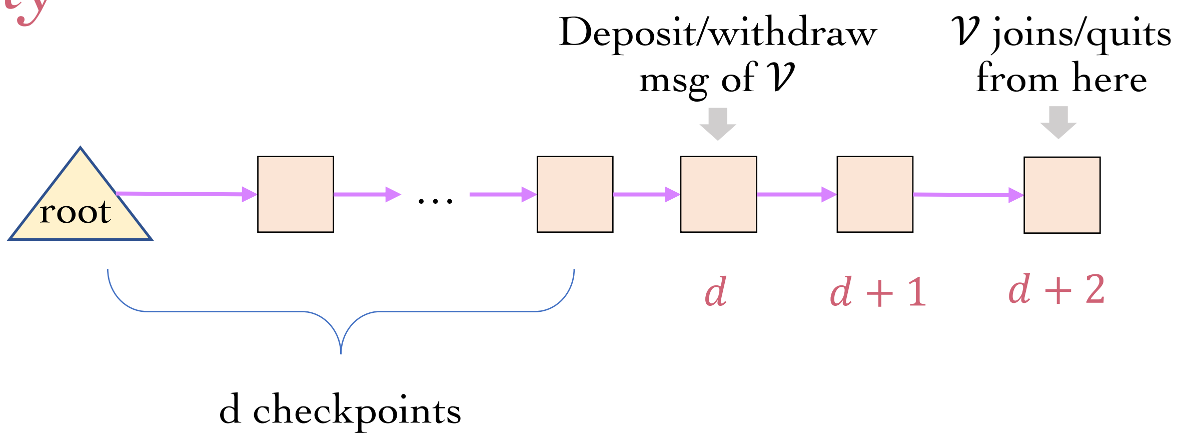
Validator Set



A committee for block producing & finalizing

Dynamic Validator Set

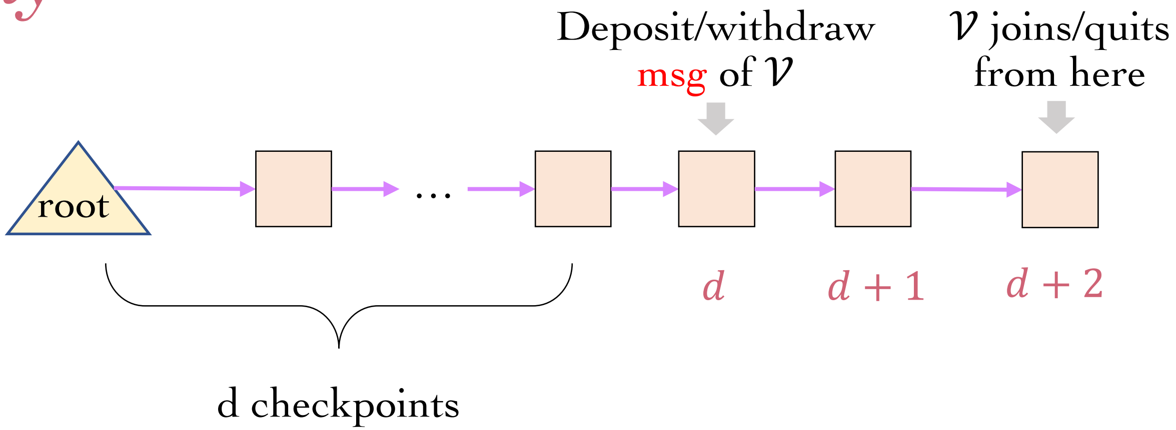
❀ Dynasty



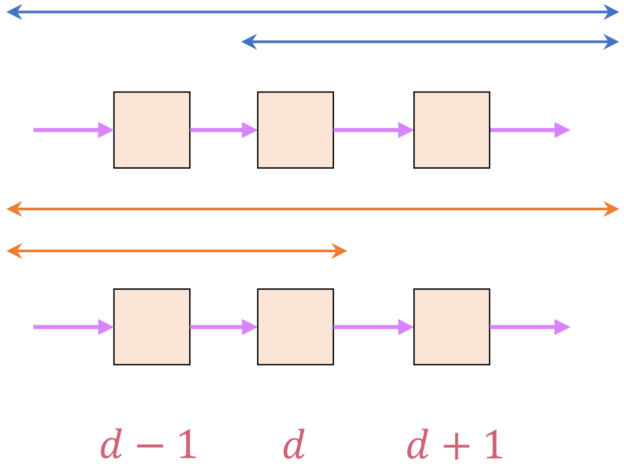
DS($\mathcal{V}$) Or DE($\mathcal{V}$) $= d + 2$

Dynamic Validator Set

- Dynasty



$\mathrm{DS}(\mathcal{V})$ Or $\mathrm{DE}(\mathcal{V})$ $= d+2$



$$\mathcal{V}_{\mathrm{f}}(d) \equiv \{\nu : \mathrm{DS}(\nu) \leq d < \mathrm{DE}(\nu)\}$$

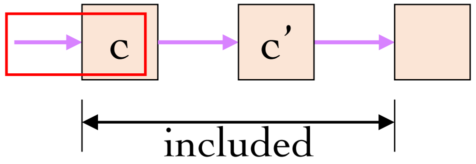
$$\mathcal{V}_{\mathrm{r}}(d) \equiv \{\nu : \mathrm{DS}(\nu) < d \leq \mathrm{DE}(\nu)\}$$

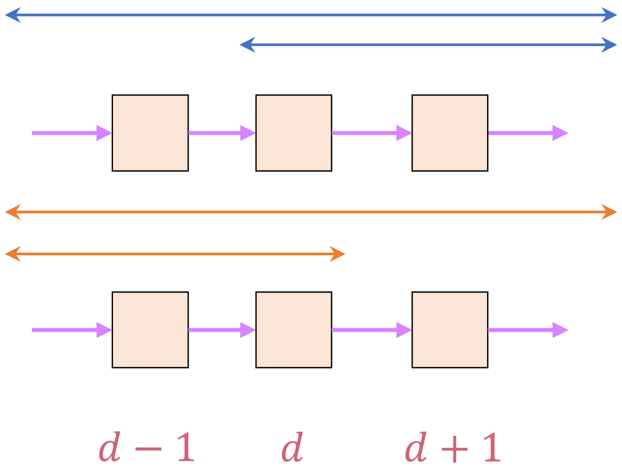
Dynamic Validator Set

Dynasty

Supermajority link: both $\geq \frac{2}{3}$ $\mathcal{V}_{\mathrm{f}}(d)$ and $\mathcal{V}_{\mathrm{r}}(d)$ vote for it.

Justified:

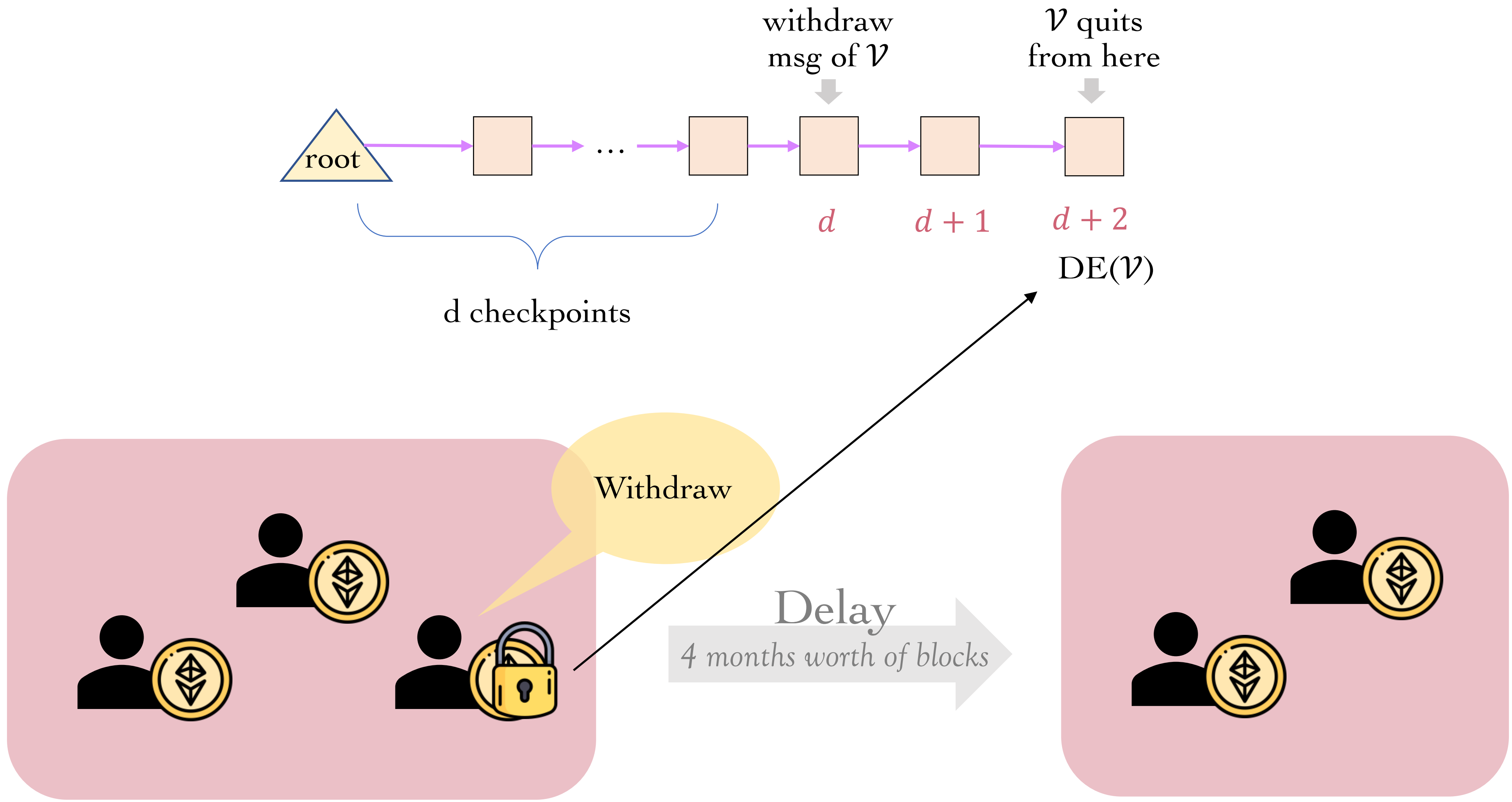




$$\mathcal{V}_{\mathrm{f}}(d) \equiv \{\nu : \mathrm{DS}(\nu) \leq d < \mathrm{DE}(\nu)\}$$

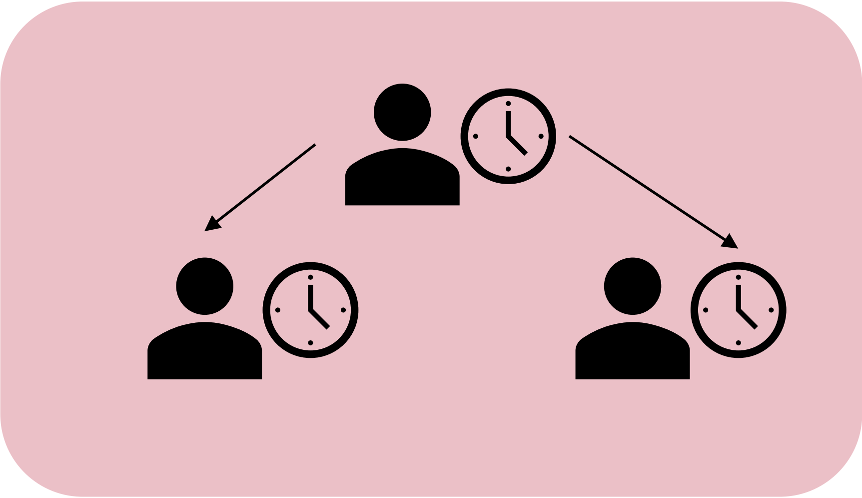
$$\mathcal{V}_{\mathrm{r}}(d) \equiv \{\nu : \mathrm{DS}(\nu) < d \leq \mathrm{DE}(\nu)\}$$

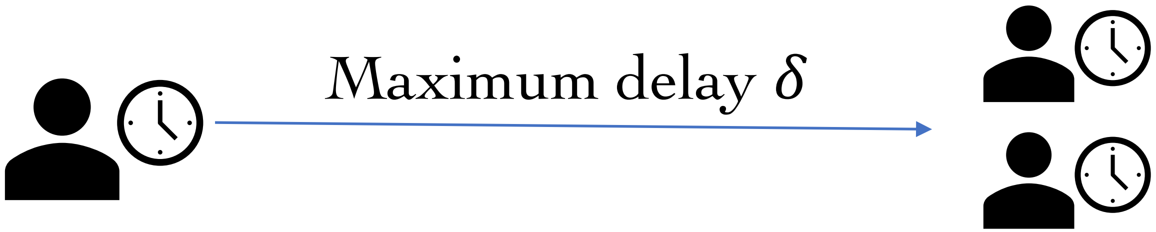
Withdraw Delay



✿ *Is the validator still participating during withdraw delay?*

Synchronous



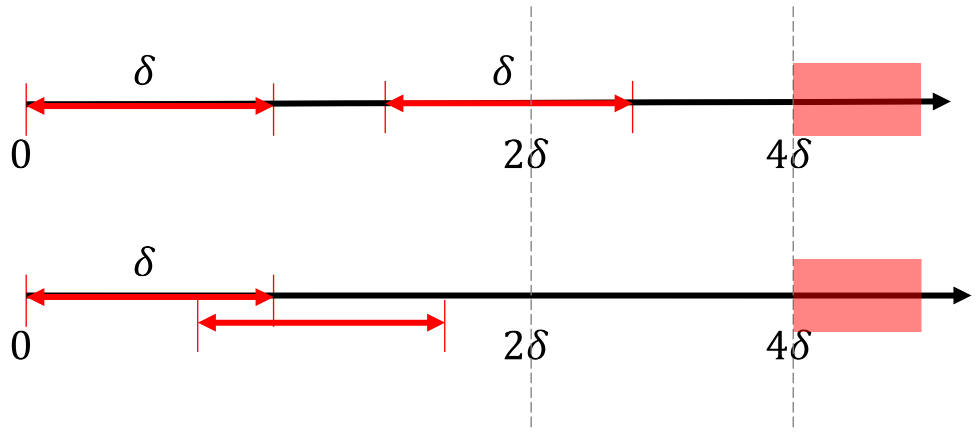


Send at t

Receive **before** $t + \delta$

Stop Long Revision Attack

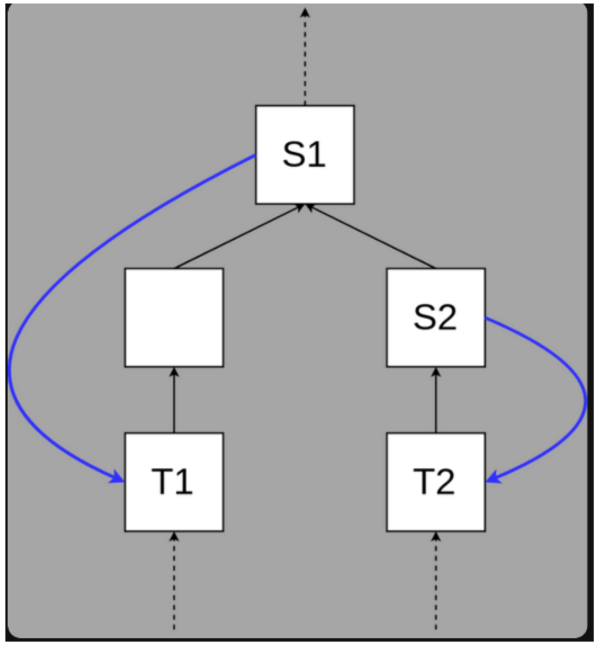
- If msg is heard by one client at $t = 0$, all others are guaranteed to have heard it by δ.
- Message delivery time window: $[0, \delta]$



Slashing Conditions

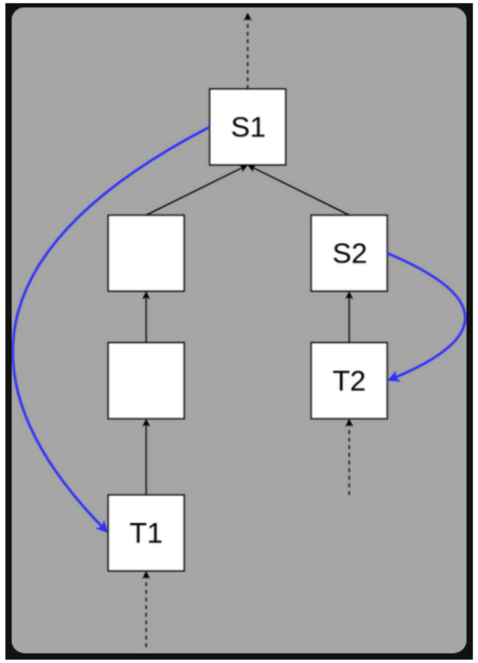
- No double vote

$$h(t_1) = h(t_2)$$



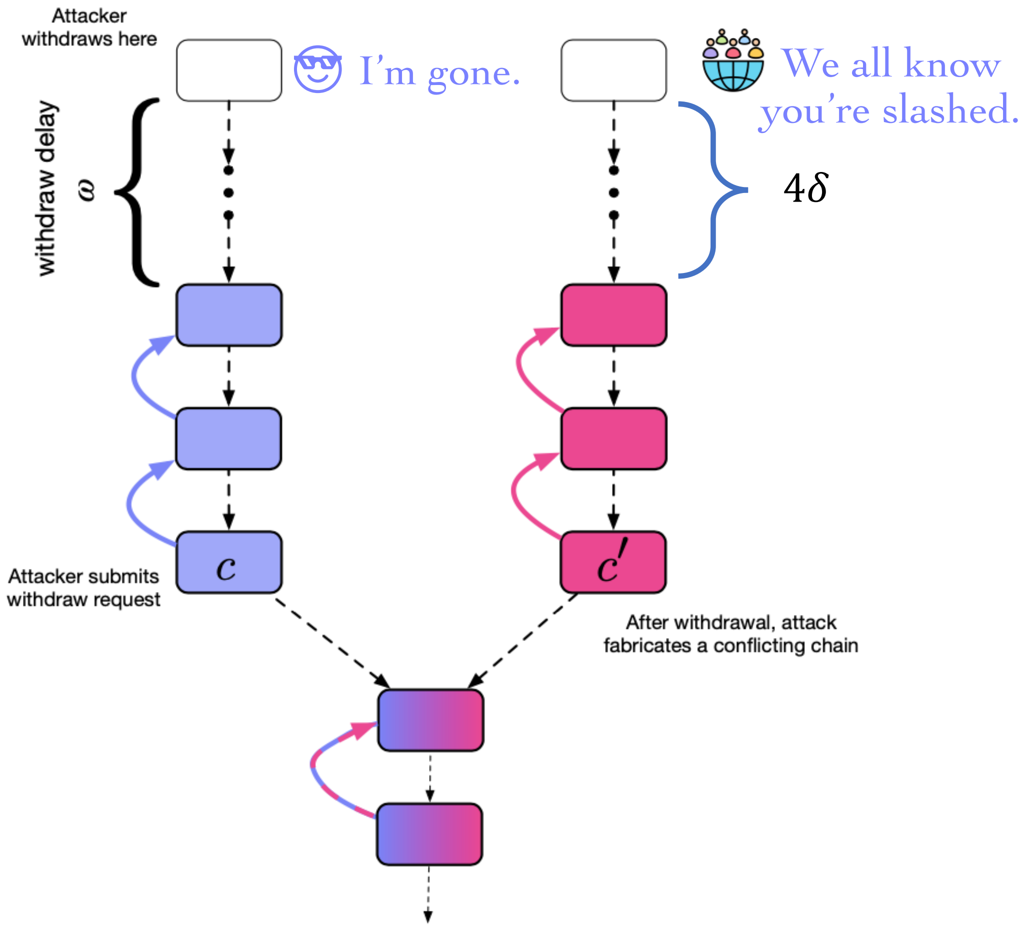
No surround vote

$$h(s\_1\ ) < h(s\_2\ ) < h(t\_2\ ) < h(t\_1)$$



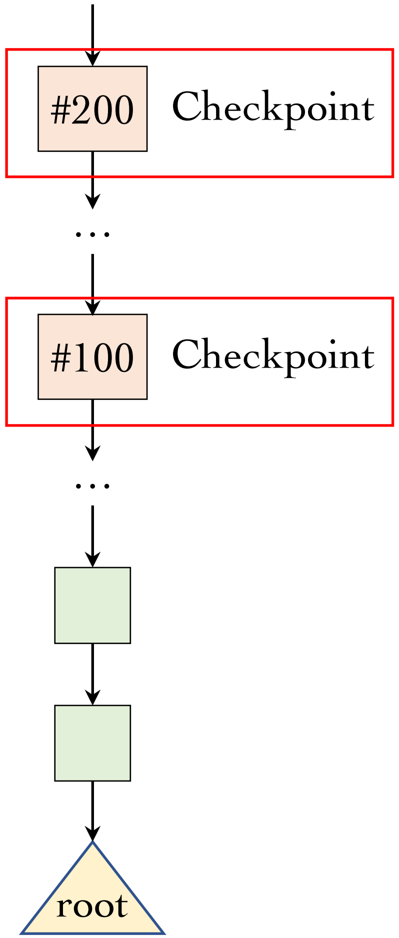
Stop Long Revision Attack

- If msg is heard by one client at $t = 0$, all others are guaranteed to have heard it by δ.
- Message delivery time window: $[0, \delta]$
- Withdraw delay $\omega > 4\delta$



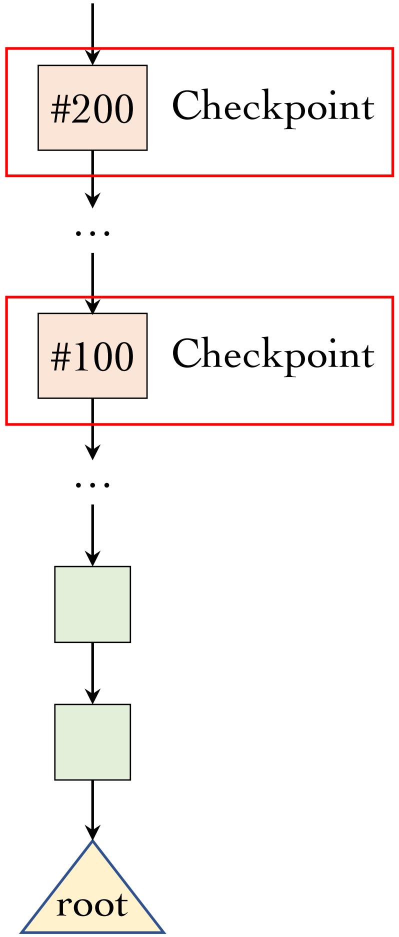
Simplify the Chain

- Checkpoints



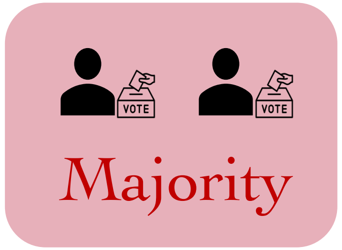
Simplify the Chain

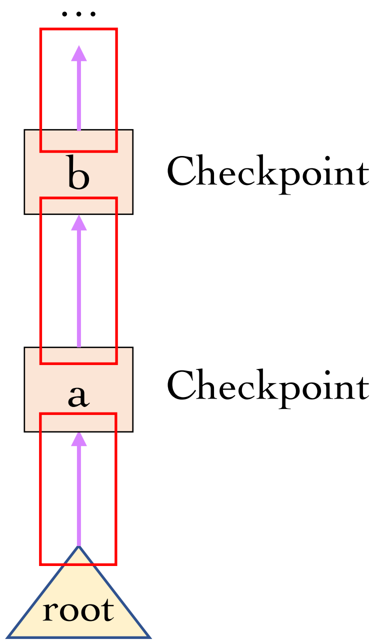
✿ Checkpoints



✿ Supermajority links

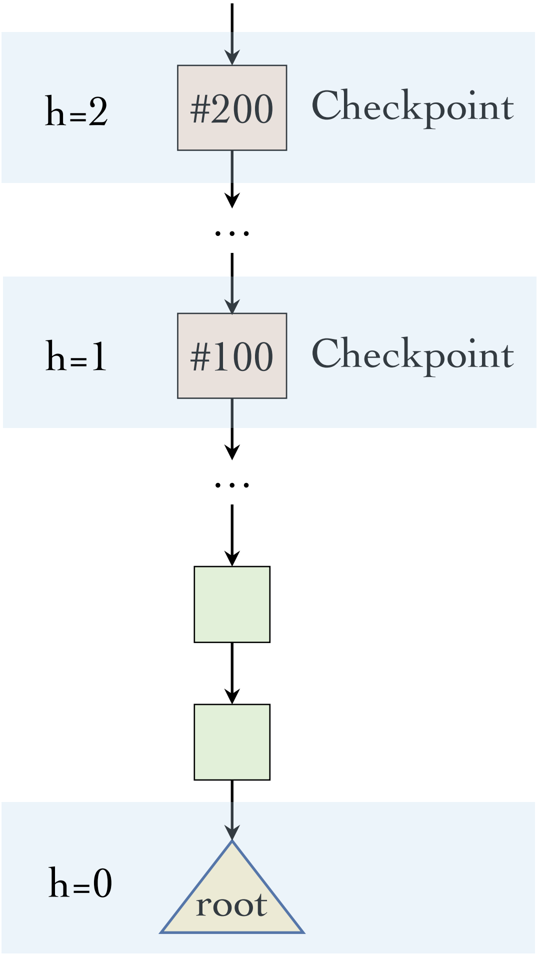
$\geq \frac{2}{3}$ vote





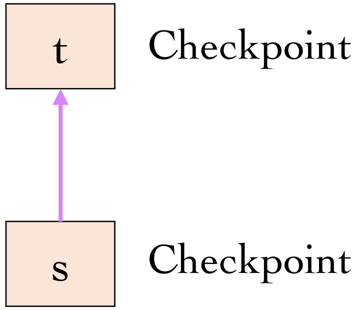
Simplify the Chain

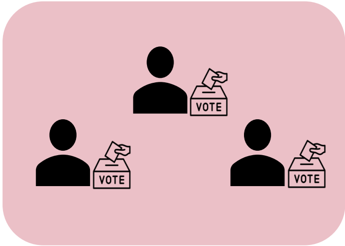
❀ height of checkpoints



h=#/100

❀ Votes





$$< \mathcal{V}, s, t, h(s), h(t) >$$

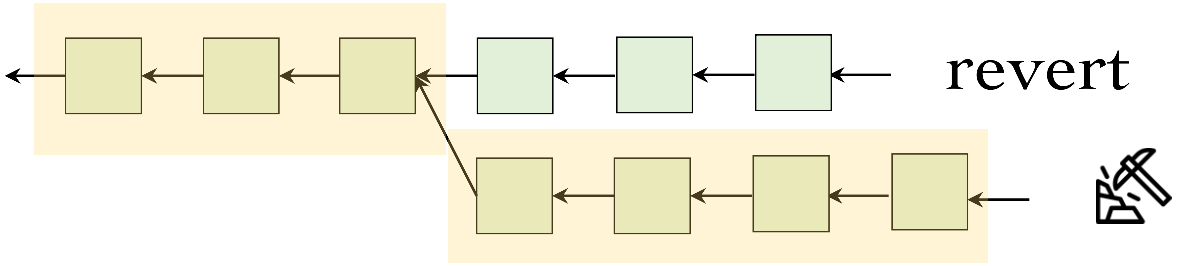
Vote: from source to target

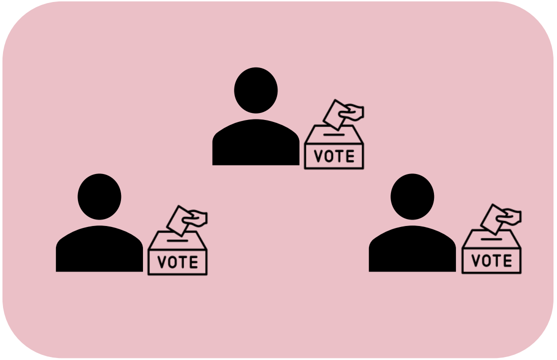
(s, t) or $s \rightarrow t$

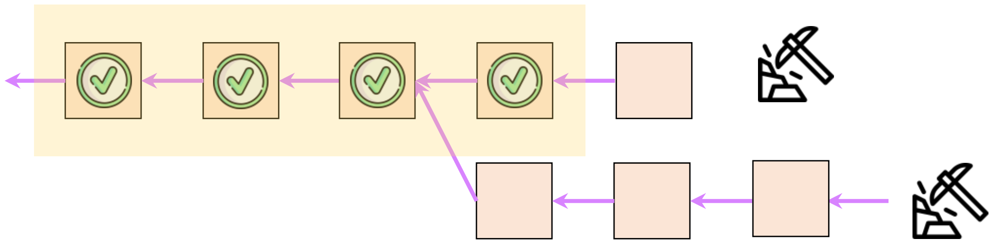
Block Finality

❀ BFT based finality, to prevent chain Reverting

e.g., “Longest chain”
(no finality)



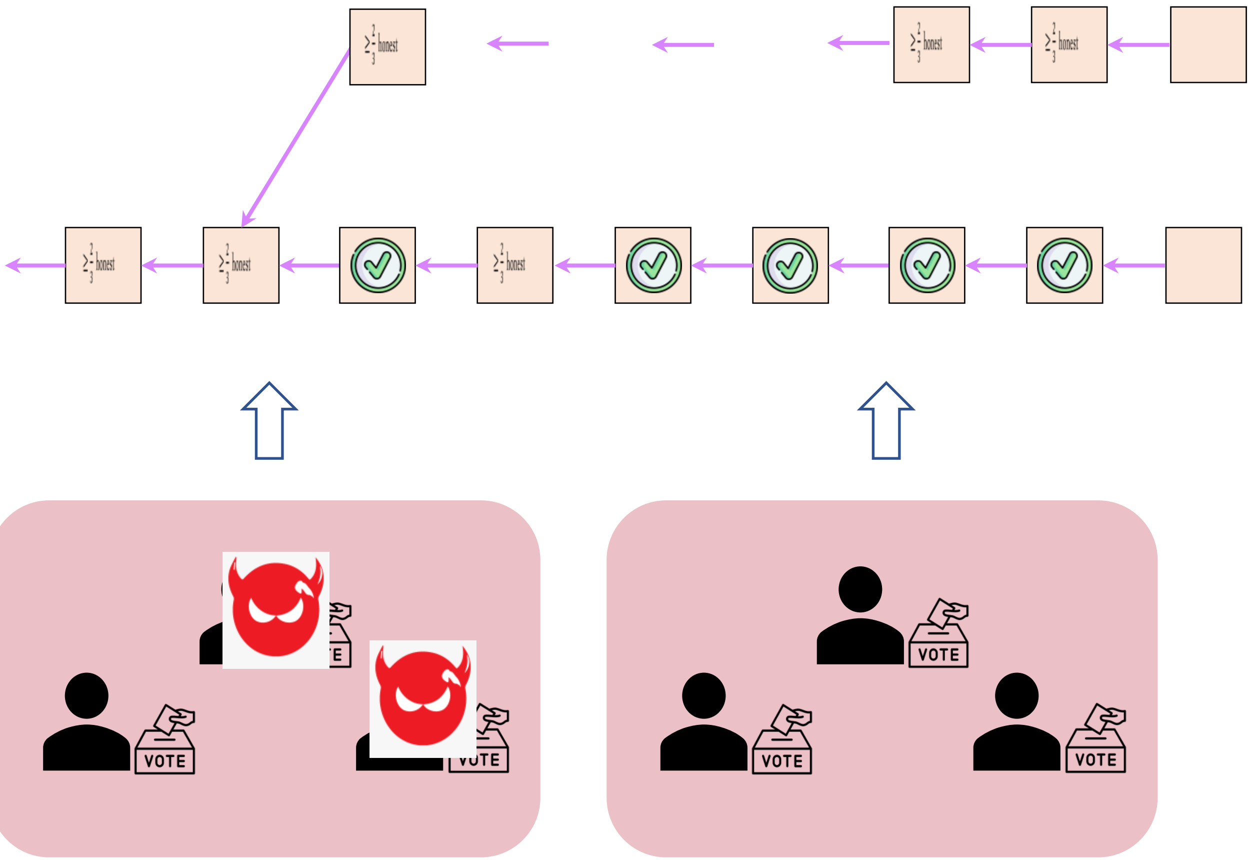




not revert

if $\geq \frac{2}{3}$ honest

Long-Range Attacks

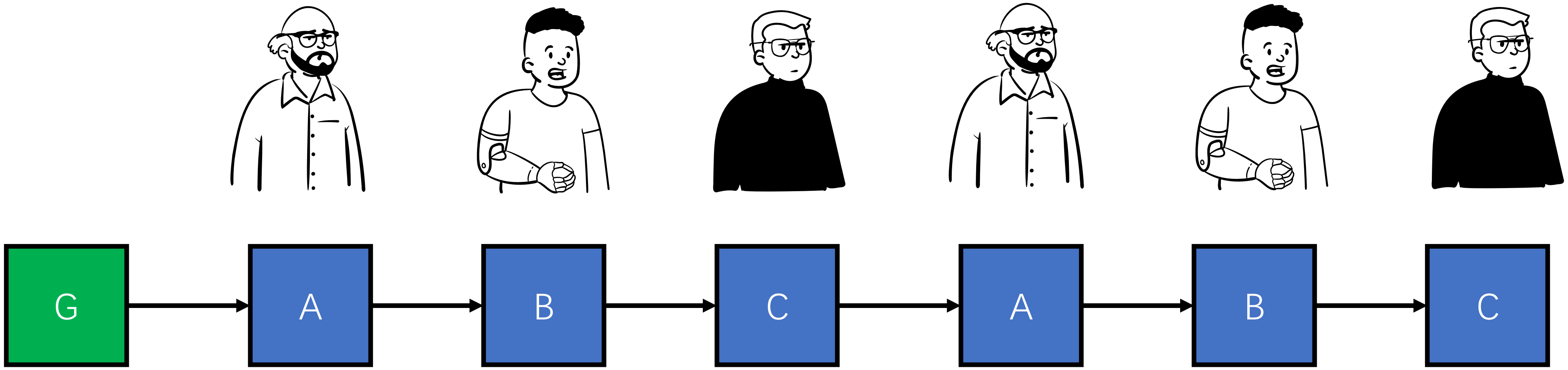


$\geq \frac{2}{3}$ honest

$\geq \frac{2}{3}$ honest

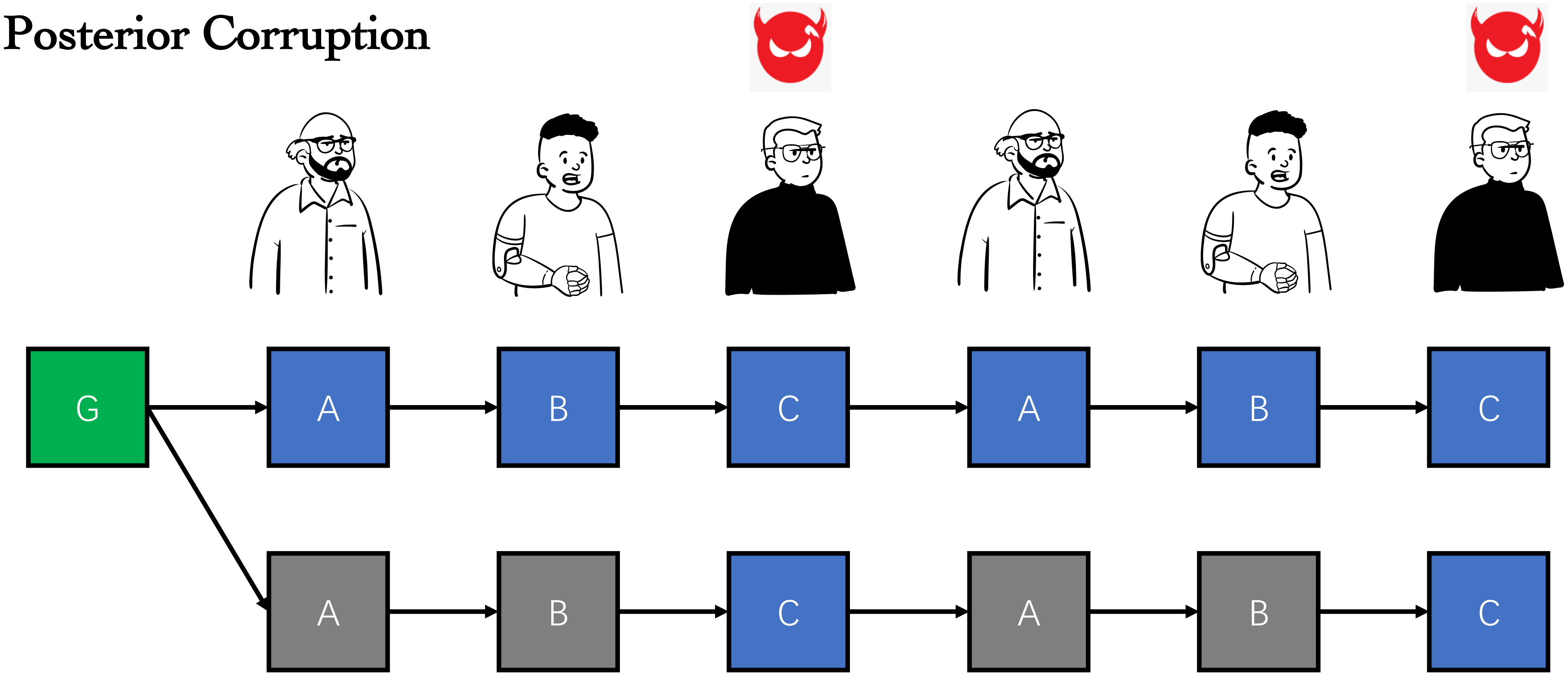
Long-Range Attacks Example

Normal Case



Long-Range Attacks Example

Posterior Corruption

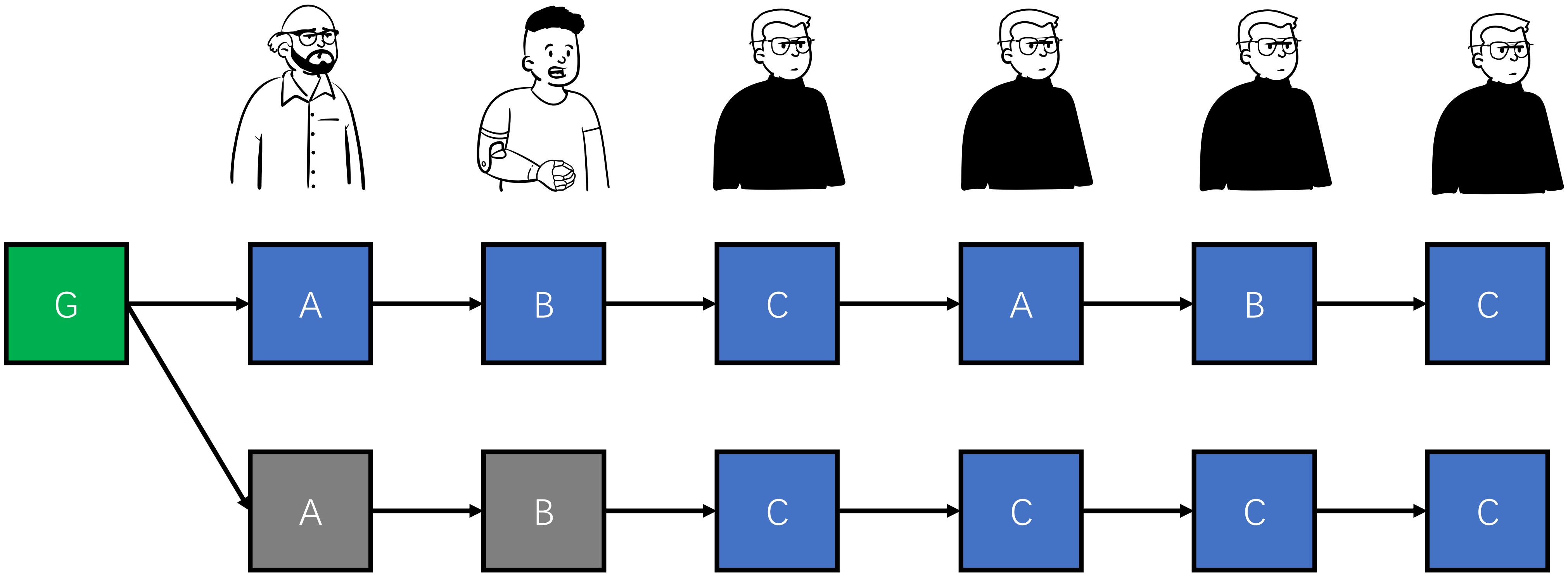


Private keys

Long-Range Attacks Example

Stake Bleeding

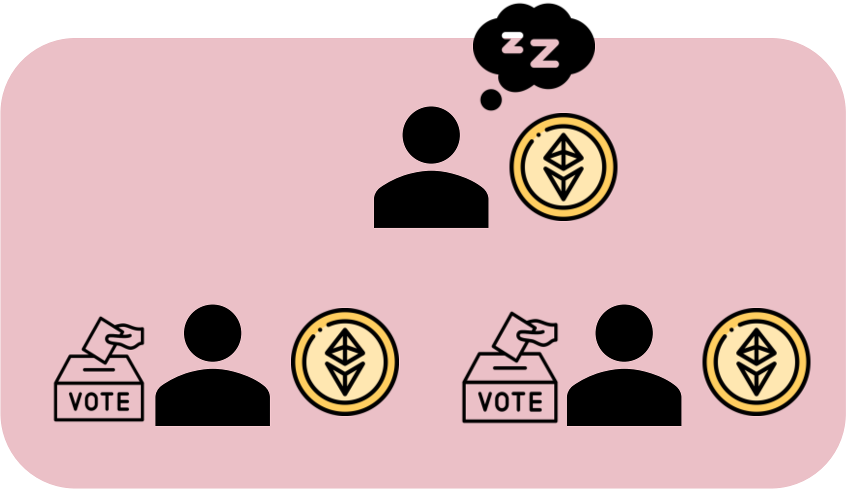
Other nodes will not get any rewards from the system, and her stake will gradually decrease



Gaži, Peter, Aggelos Kiayias, and Alexander Russell. "Stake-bleeding attacks on proof-of-stake blockchains." In *2018 Crypto Valley conference on Blockchain technology (CVCBT)*, pp. 85-92. IEEE, 2018.

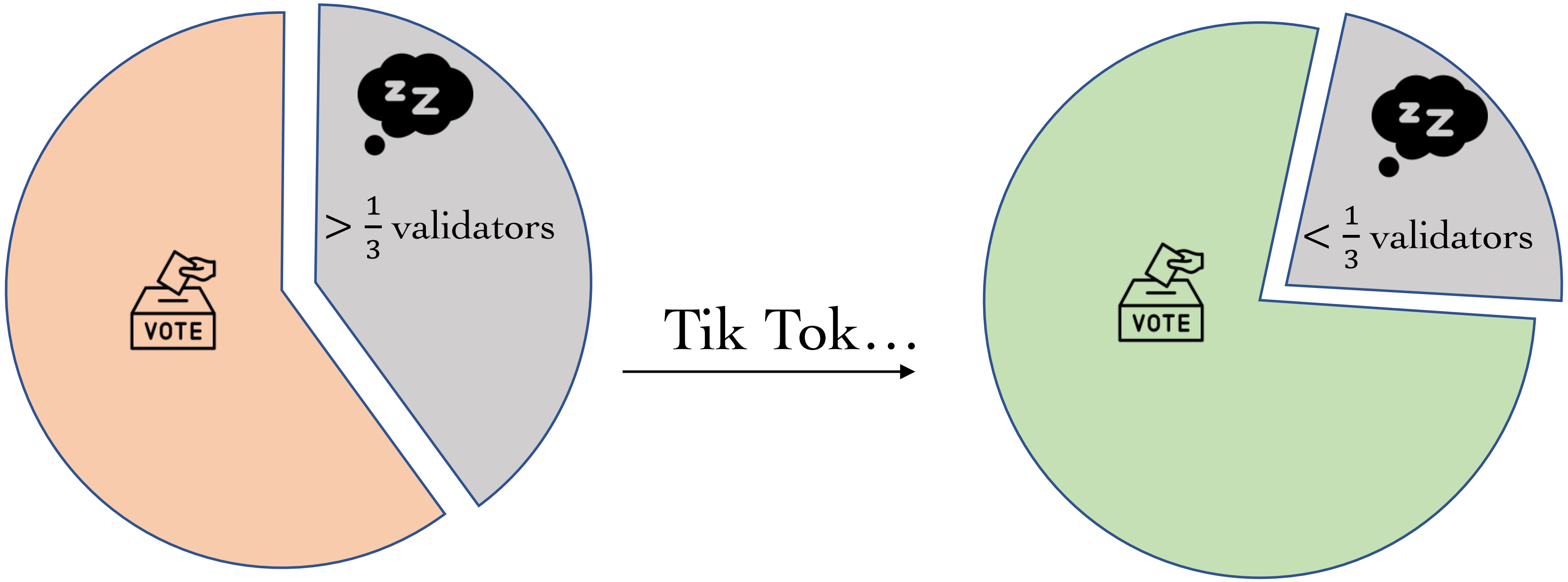
Leaking (Stop catastrophic crashes)

- A validator's deposit leaks slowly if it does not vote for checkpoints.



Leaking (Catastrophic crashes)

- A validator's deposit leaks slowly if it does not vote for checkpoints.
- Comparison of :



Chain stuck

Chain continues…

Comparison

Proof of Work Bitcoin = Computation + Longest Chain

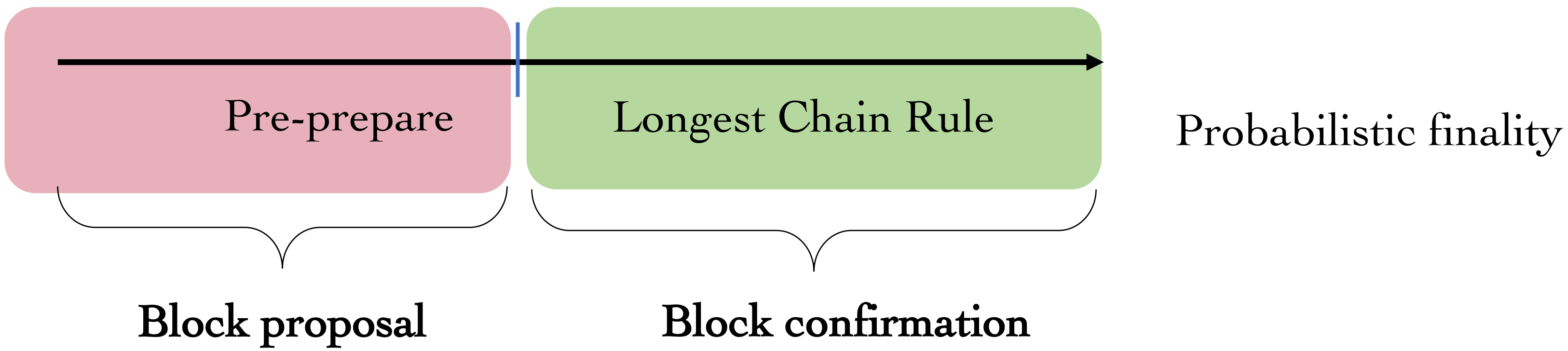
Proof of Work Ethereum = Computation + GHOST

Proof of Stake Ethereum = Rotation ? + LMD-GHOST + Casper FFG

Proof of Authority

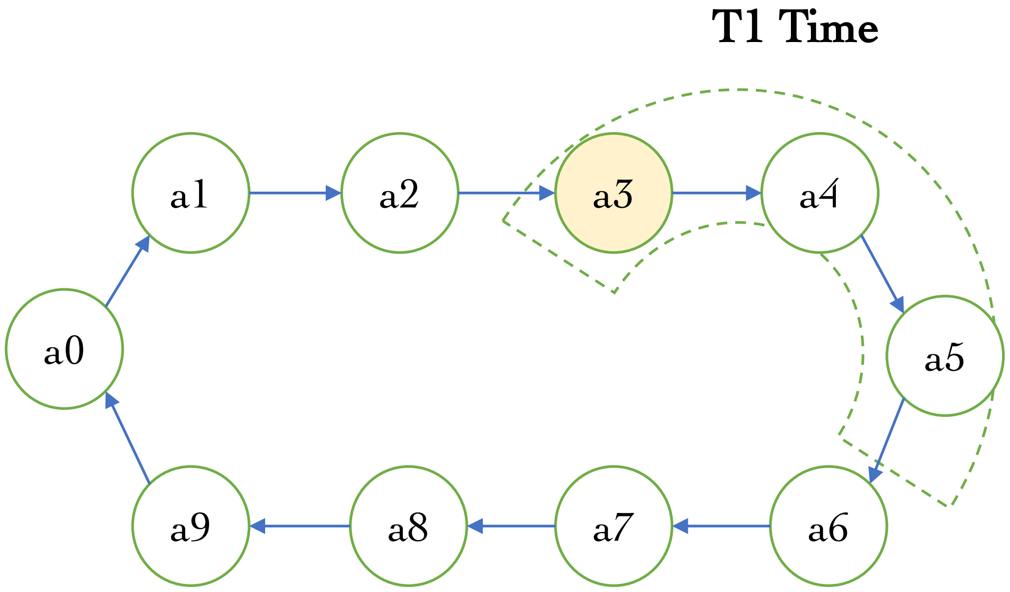
Proof of Authority Clique

- Permissioned ✔
- Leader-based ✔
- Communication-based ✖
- Safety-First

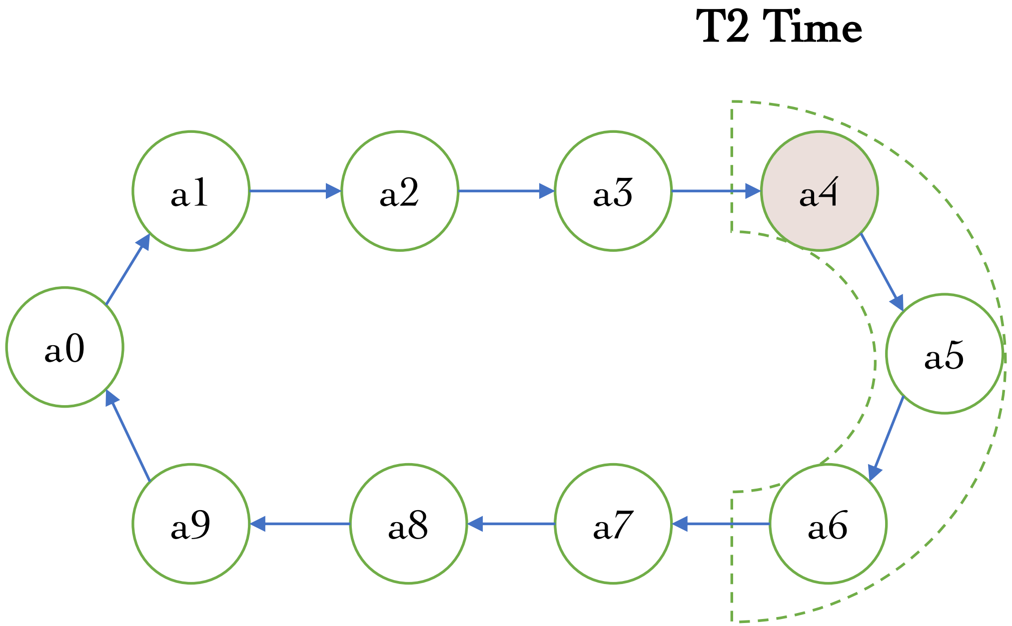


- Permissionless
- Leaderless
- Computation-based ✖
- Liveness-First ✔

Clique Rotation Schema

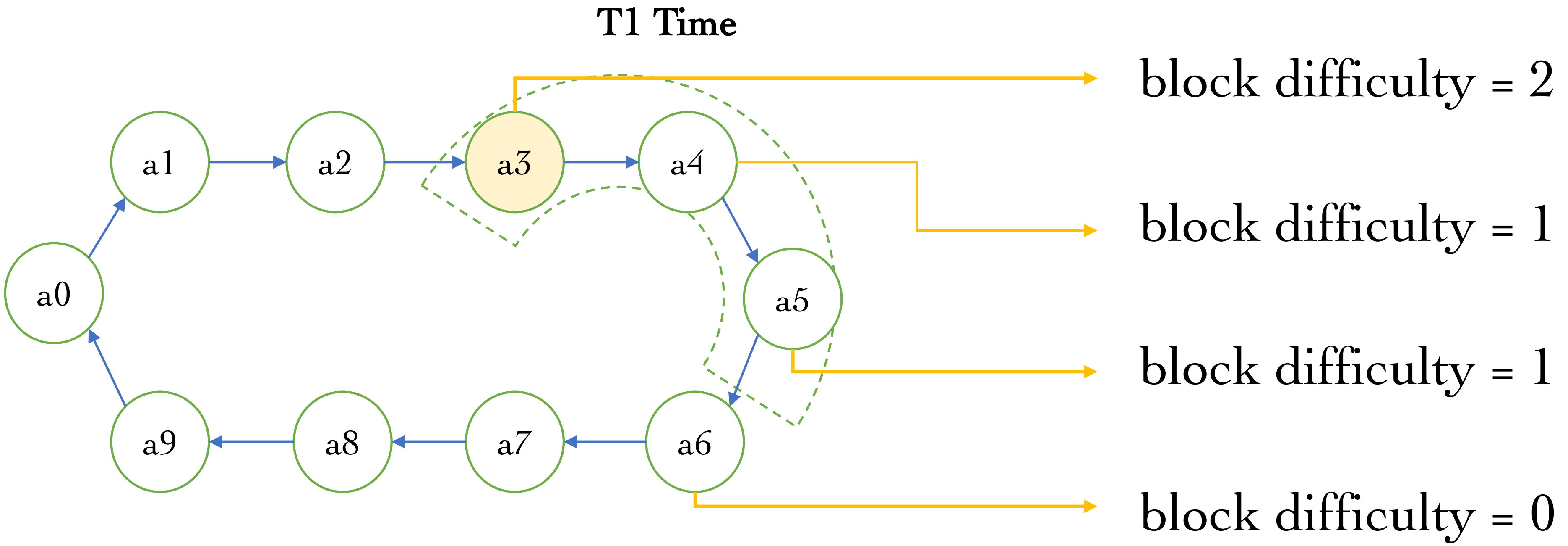


in-turn sealer: a3
edge-turn sealer : a4
edge-turn sealer: a5



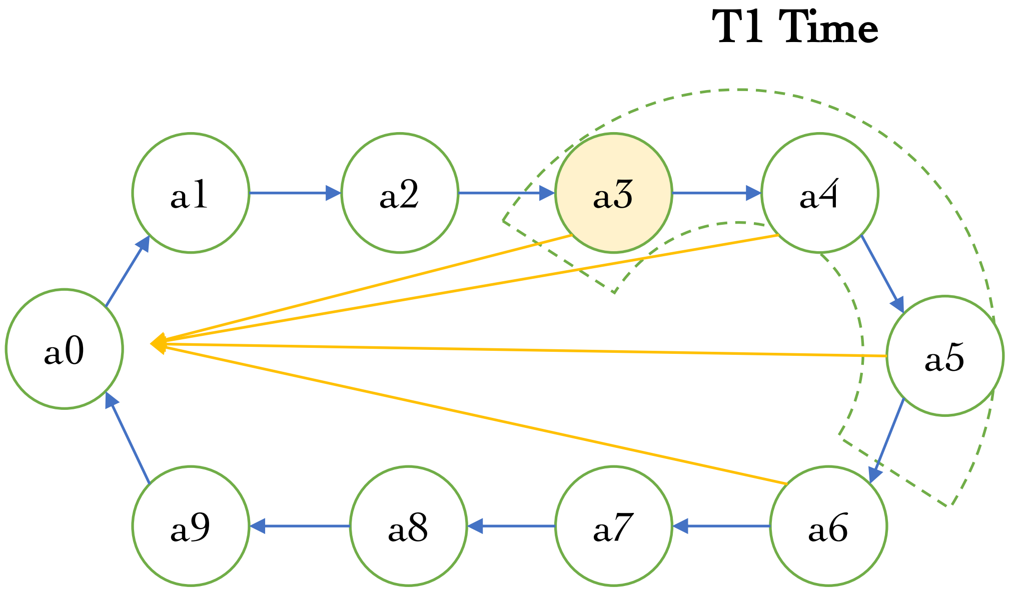
in-turn sealer: a4
edge-turn sealer : a5
edge-turn sealer: a6

Delay and Difficulty Mechanism



in-turn sealer: a3
edge-turn sealer : a4
edge-turn sealer: a5

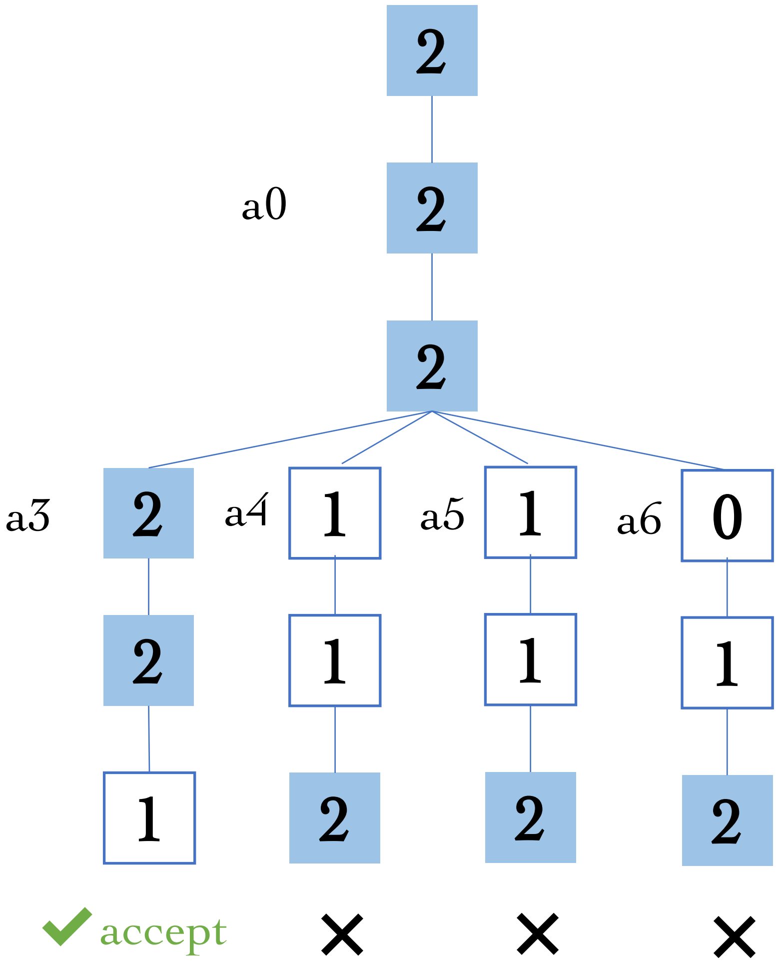
Delay and Difficulty Mechanism



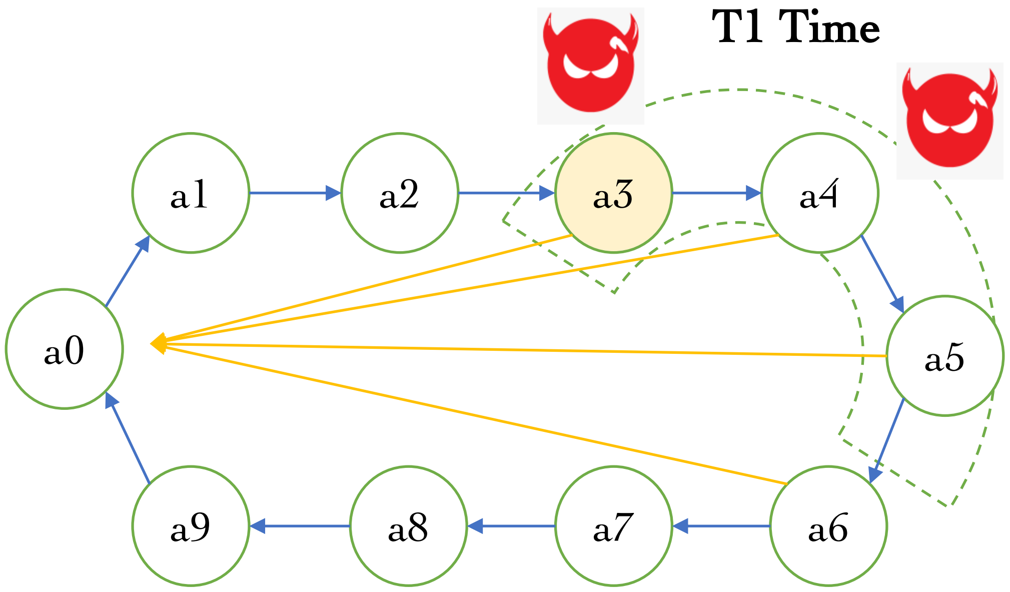
Priority parameters

$block.diff$=2 or 1

$delay\ time$



Our Work



Priority parameters

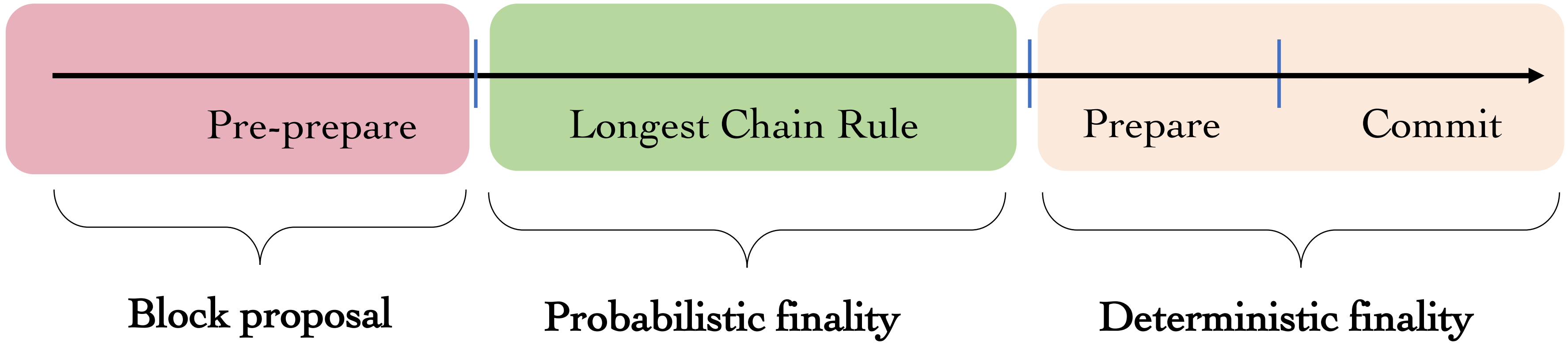
$block.diff$=2 or 1

delay time

Exploring Unfairness on Proof of Authority: Order Manipulation Attacks and Remedies. 17th ACM ASIA Conference on Computer and Communications Security (ACM ASIACCS 2022)
Qin Wang*, Rujia Li*, Shiping Chen, Qi Wang, Yang Xiang (*equal contribution)

Frontrunning Block Attack in PoA Clique: A Case Study. 4th IEEE International Conference on Blockchain and Cryptocurrency (ICBC 2022)
Xinrui Zhang, Qin Wang, Rujia Li, Qi Wang

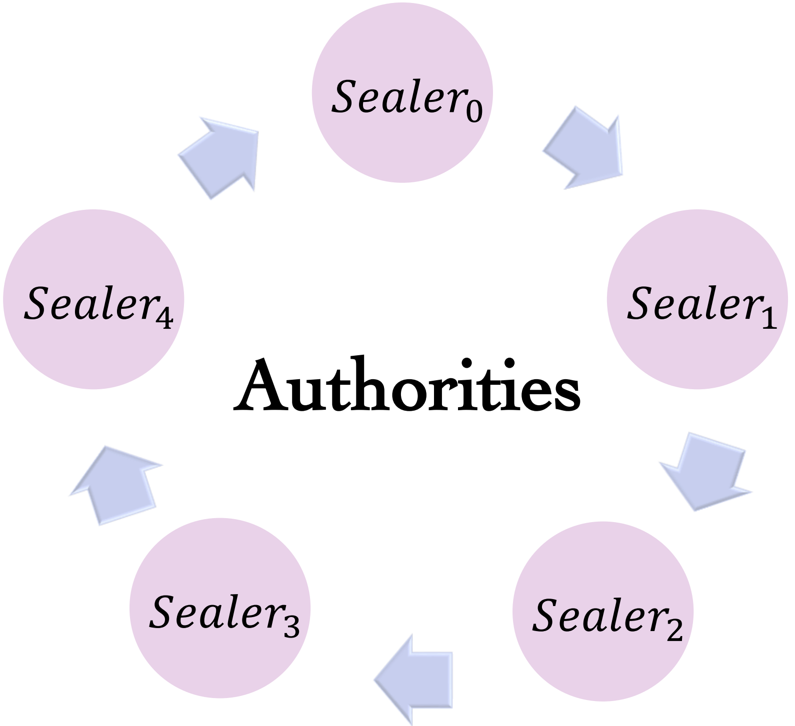
Proof of Authority Aura



Aura Sealer Rotation

Sealer rotation:

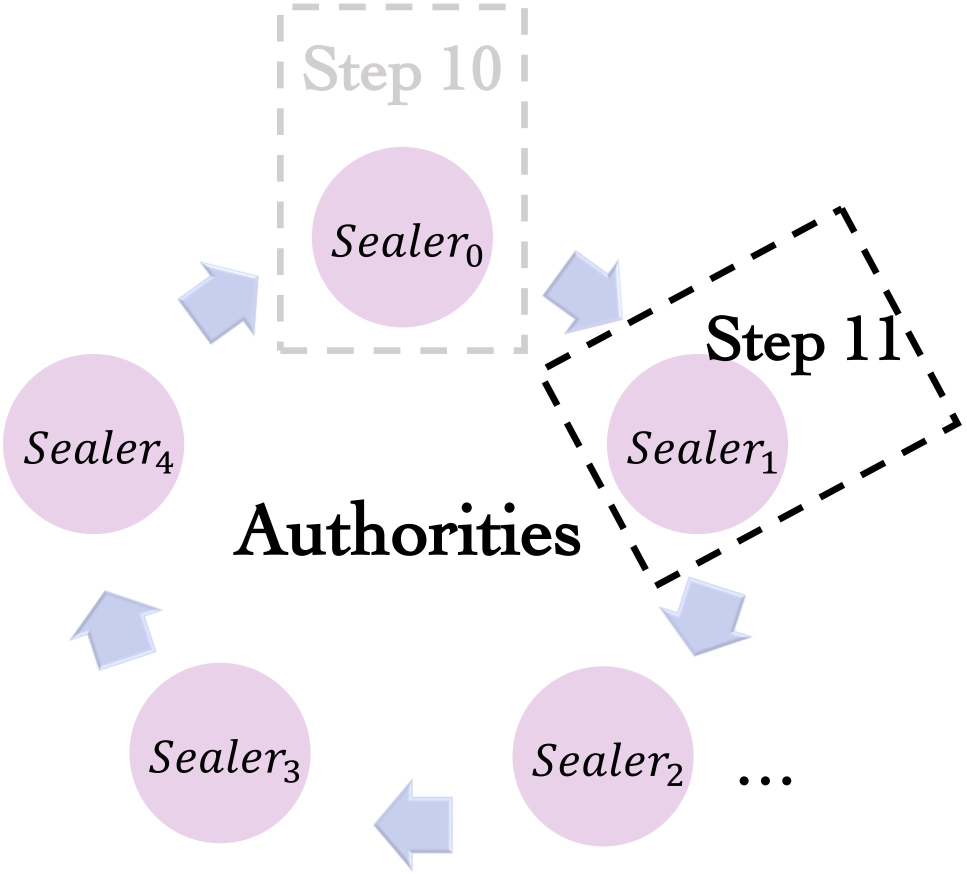
1. $\text{Step} = \frac{\text{clock\_time}}{\text{step\_duration}}$
2. n = |Sealers|
3. i = Step *mod* n
4. $Sealer_i$'s turn



Aura Sealer Rotation

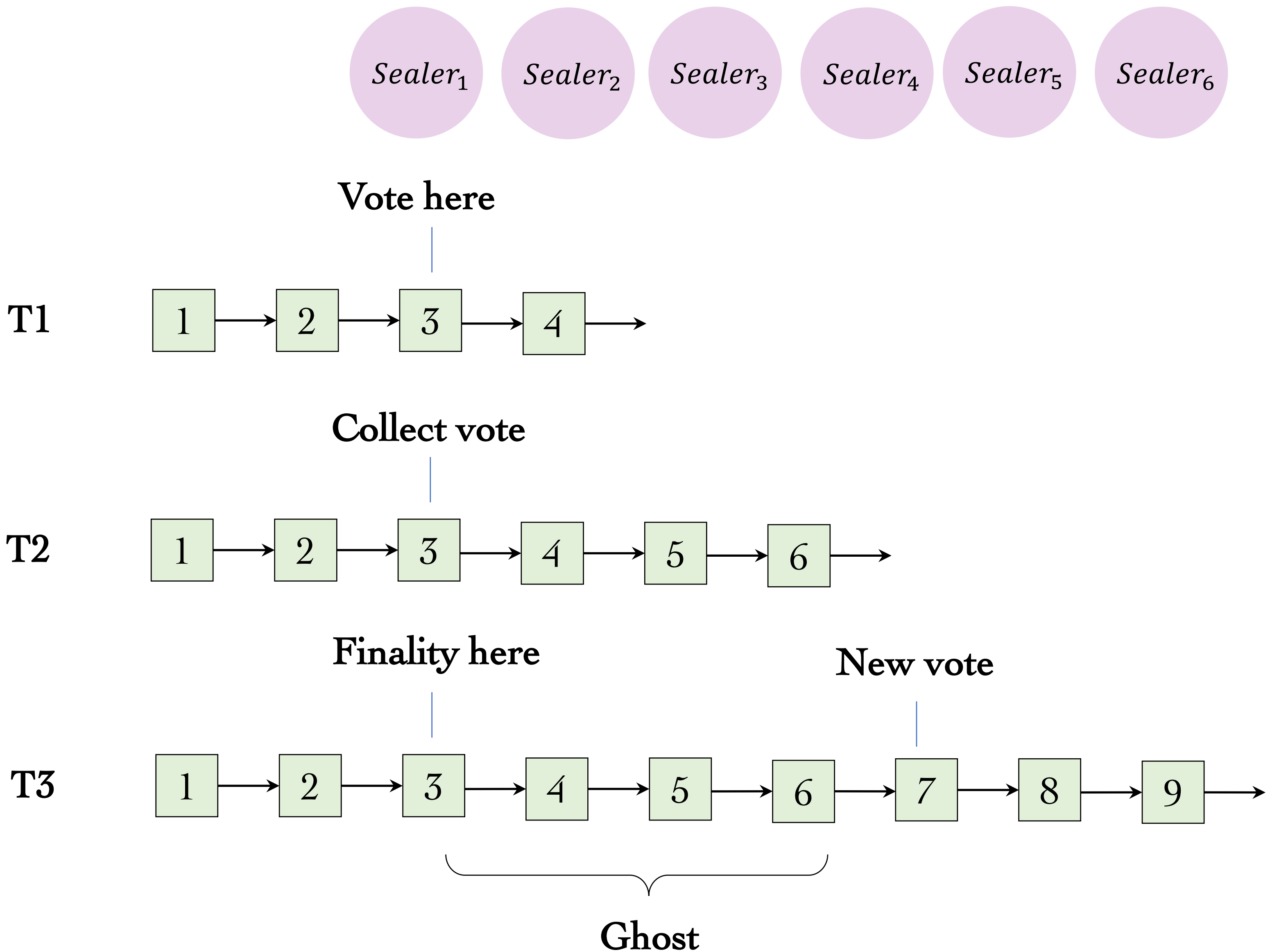
Sealer election:

1. $\text{Step} = \frac{\text{clock\_time}}{\text{step\_duration}}$
2. n = |Sealers|
3. i = Step *mod* n
4. $Sealer_i$'s turn

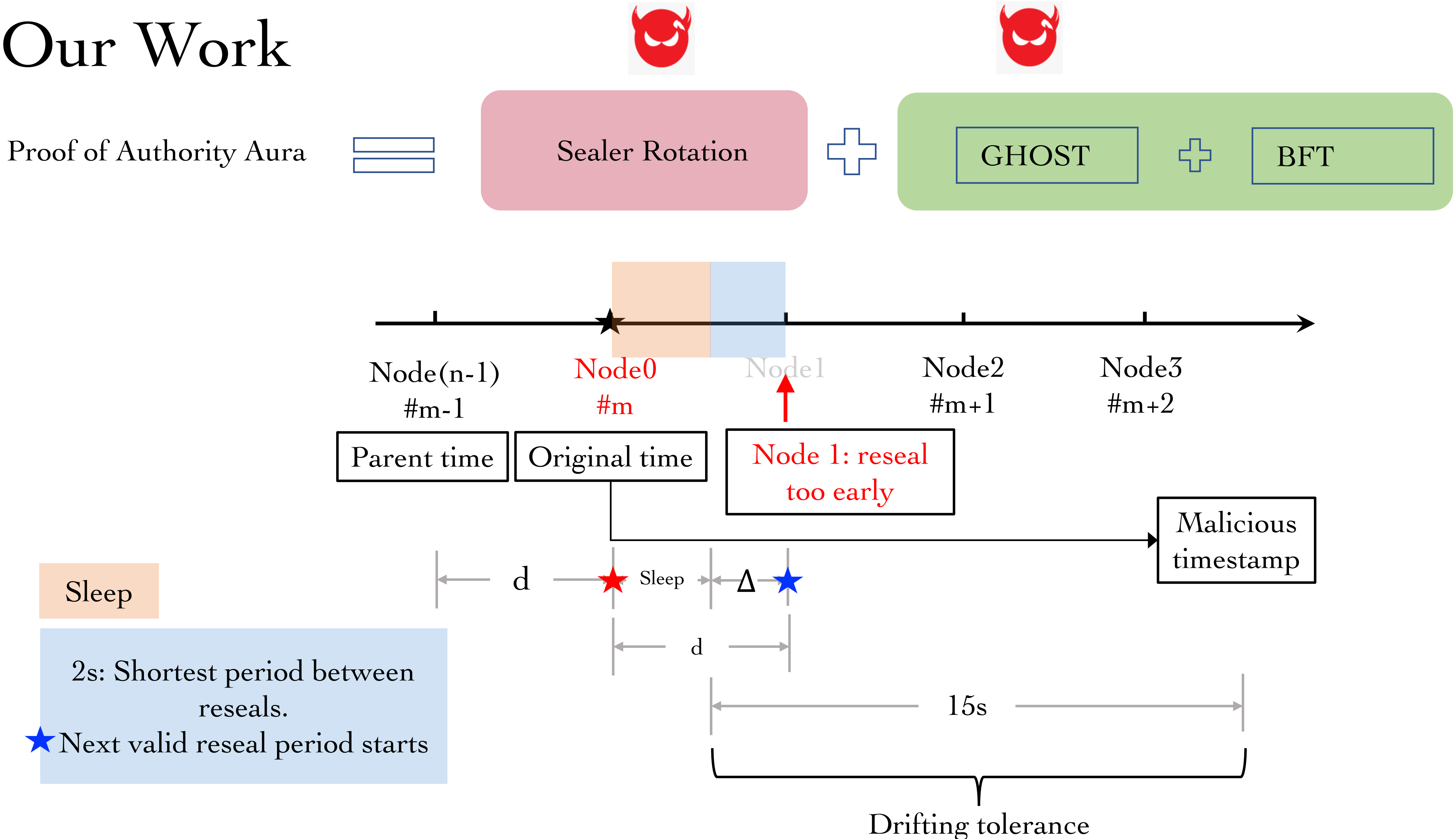


e.g., Step=11
|Sealer|=5
Step mod |Sealer|=**1**
$Sealer_1$'s step

Proof of Authority Aura

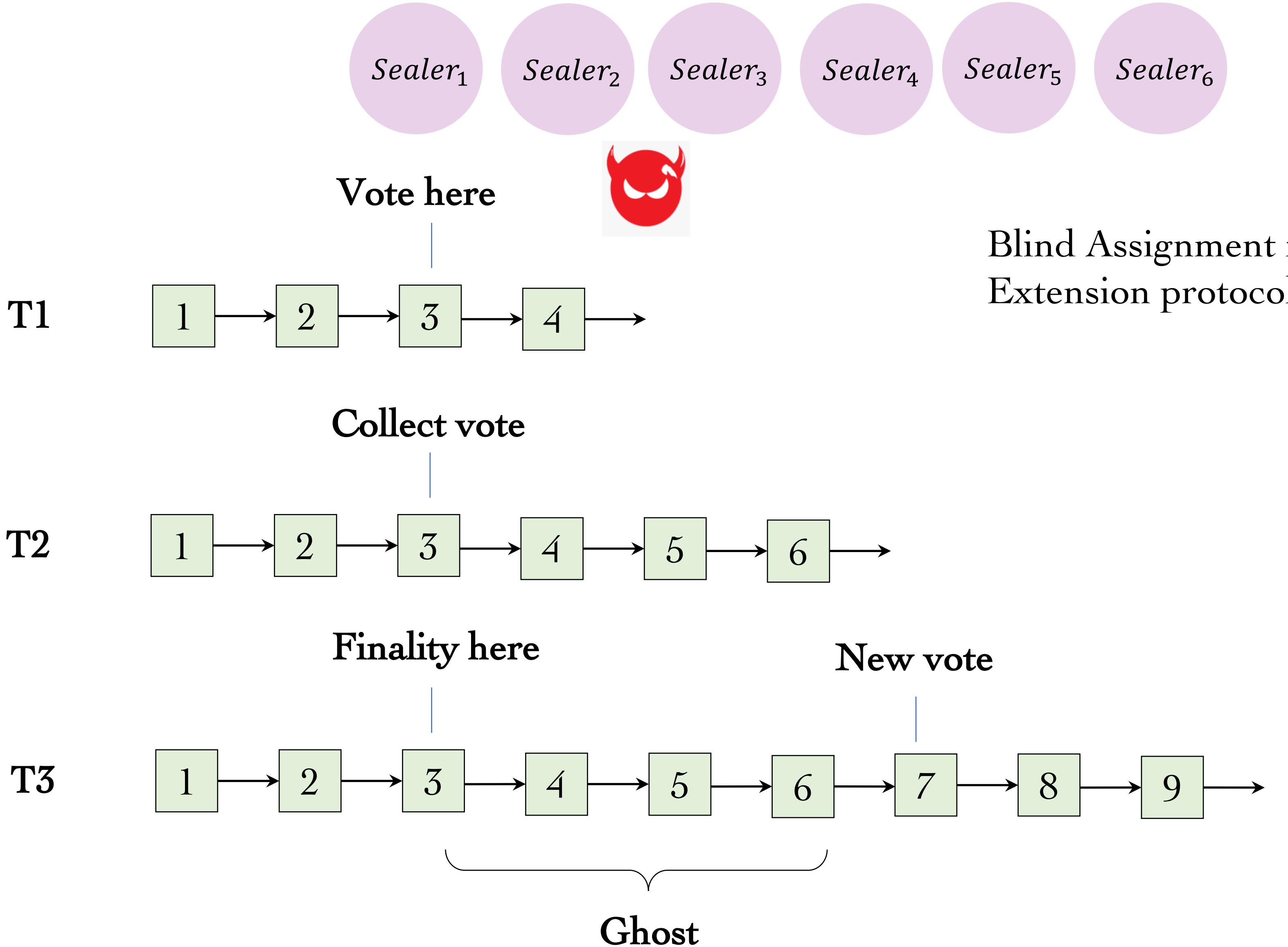


Our Work



Proof of Authority Aura
Sealer Rotation
GHOST
BFT
Node(n-1)
#m-1
Node0
#m
Node1
Node2
#m+1
Node3
#m+2
Parent time
Original time
Node 1: reseal too early
Malicious timestamp
d
Sleep
Δ
d
15s
Drifting tolerance
Sleep
2s: Shortest period between reseals.
Next valid reseal period starts

Enhanced Aura



Blind Assignment for Blockchain Extension protocol (BABE)

Comparison

Proof of Work Bitcoin	=	Computation	+	Longest Chain
Proof of Authority Clique	=	Sealer Rotation	+	GHOST
Proof of Authority Aura	=	Sealer Rotation	+	GHOST + BFT
Polkadot Enhanced Aura	=	BABE	+	GHOST + BFT

GRANDPA