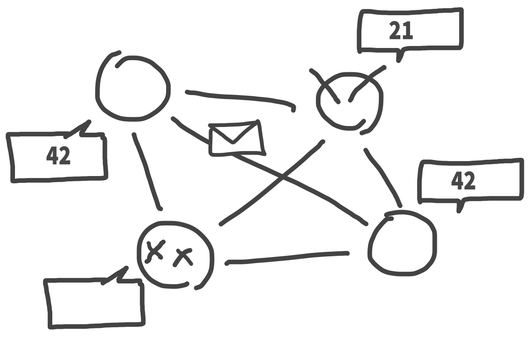
Formal Methods & The Blockchain

Byzantine Consensus Protocols



- distributed system reach **consensus** on state 42
- despite malicious & failing **byzantine** participants
- 3f + 1 participants to tolerate f byzantine participants
- sending signed messages over a network

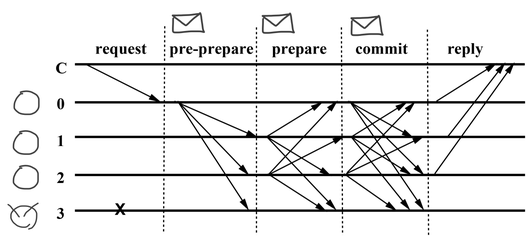


Figure. Happy path in PBFT [from Castro & Liskow, 1999]

Agreement. No two correct servers decide differently.

Integrity. No correct server decides twice.

Weak validity. If all servers are correct and propose the same value x, then no correct server decides a value different from x. Furthermore, if all servers are correct and some server decides x, then x was proposed by some server.

Termination. Every correct server eventually decides some value.

(weak byzantine consensus [Cachin et al, 2011])

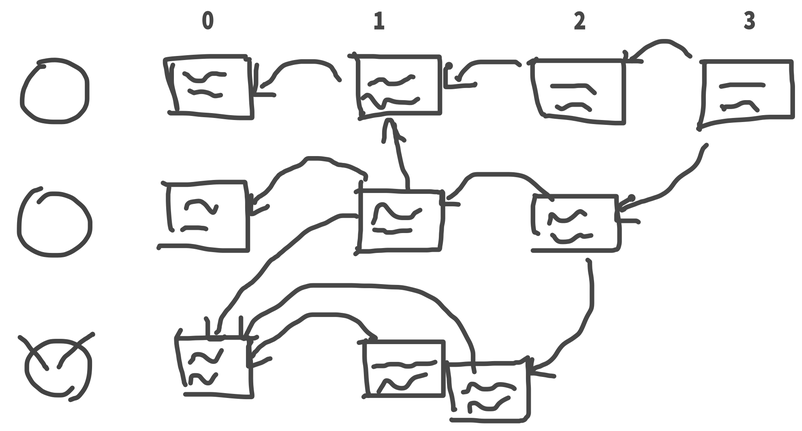
Castro M & Liskov B. "Practical Byzantine Fault Tolerance" OSDI (1999)

Cachin C, Guerraoui R, Rodrigues L. "Introduction to Reliable and Secure Distributed Programming" (2011)

Blockmania QED

joint work with George Danezis

- Blockmania consensus protocol [Danezis et al, 2018]

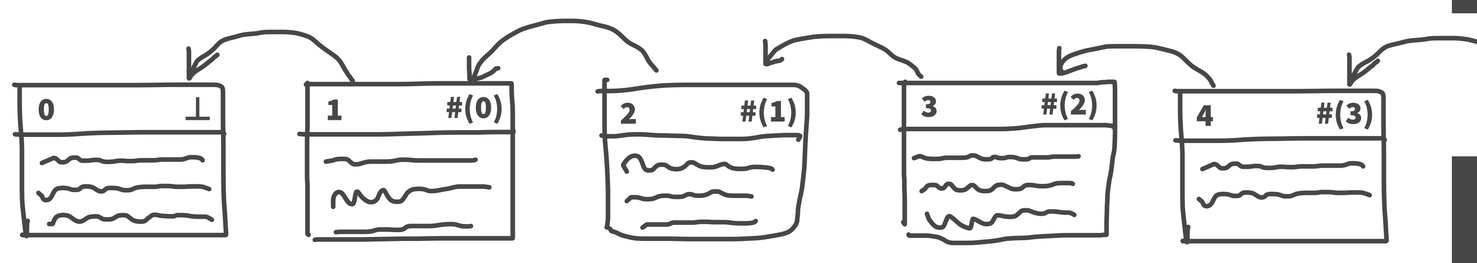


- shared block DAG built by gossiping blocks referencing blocks
- to **interpret** protocol P e.g. consensus or broadcast, by interpreting references to block as message from to
- **todo.** formal proof of correctness

¿ Which P can be interpreted? ¿ How to model ? ¿ ... ?

Danezis G & Hrycyszyn D. "Blockmania: from Block DAGs to Consensus" (2018)

Blockchains

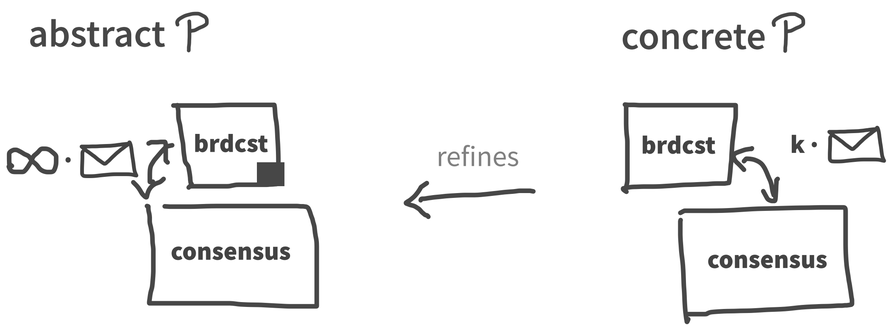


- ledger shared in a distributed system with untrusted members
- tamper-resistant through cryptographic hashes of the previous block **#(i-1)**
- to reach **consensus** on == transactions || smart contract byte code & calls

Federated Consensus

joint work with Álvaro García-Pérez & Alexey Gotsman

- based on the Stellar consensus protocol [Mazières, 2015]



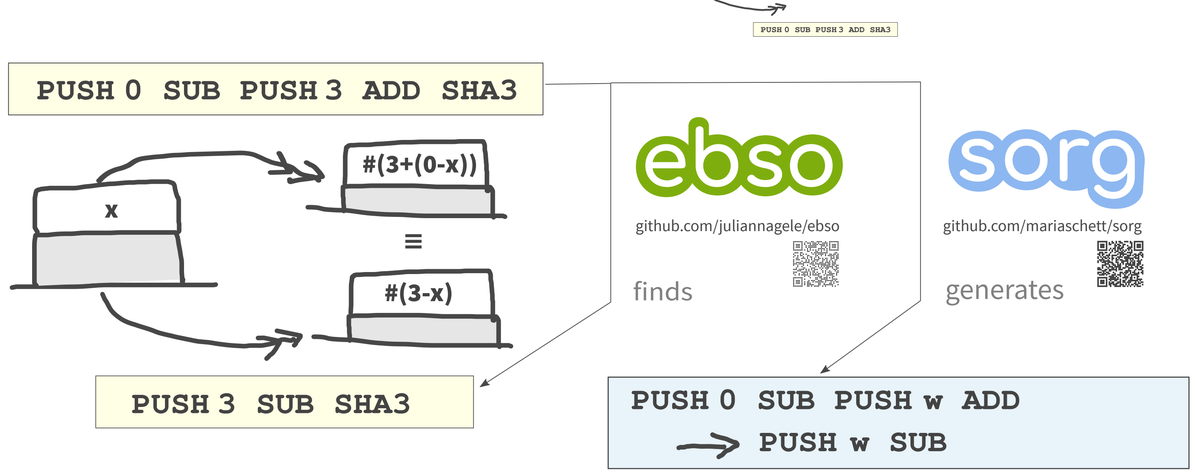
- showed abstract protocol P "implements" weak byzantine consensus using broadcast as (nearly) a blackbox ■
- showed concrete P **refines** abstract P
- showed concrete protocol P implements weak byzantine consensus

Mazières D. "The Stellar Consensus Protocol: A Federated Model for Internet-level Consensus" Whitepaper (2015)

Optimizing EVM byte code

joint work with Julian Nagele

- Ethereum Virtual Machine (EVM) executes byte code for gas == $



- find cheaper, observationally equvivalent ≡ EVM byte code **unbounded superoptimization** [Jangda et al, 2017] with Z3
- found **938** optimizations in 2500 smart contracts on Ethereum blockchain
- generated **397** peephole optimization/rewrite rules shown terminating by **Wanda** & non-confluent by **CSI**

Jangda A & Yorsh G. "Unbounded Superoptimization". Onward! (2017)

UCL

mail@maria-a-schett.net
maria-a-schett.net