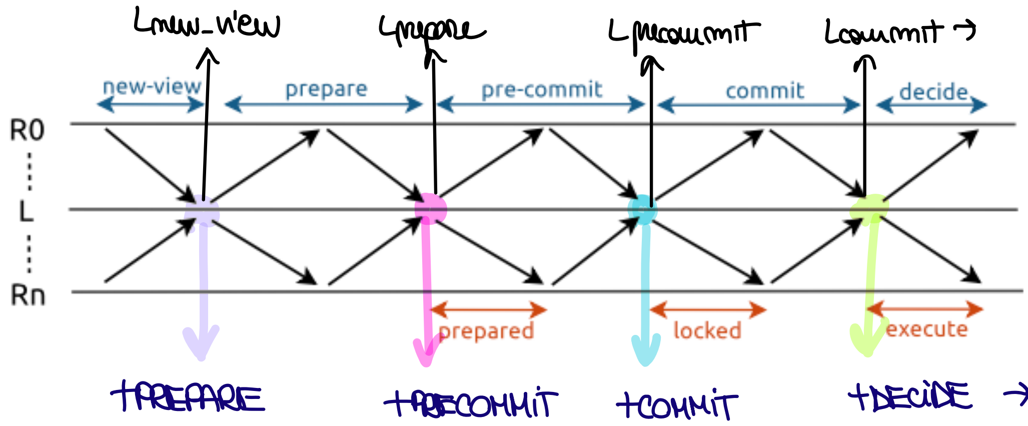

→ times of quorum formation

Lx → latency vector retained by leader where Lx[i] → time registered by leader when receives message of type x from replica i,

=> t(nextX) = findQuorum(LX)

↓

priority queue on LX to get the fastest replicas

FINAL TIME OF ONE VIEW → (tPREPARE + tPRECOMMIT + tCOMMIT + tDECIDE)

↓

Obs: Assume the time of sending messages to replicas is the same for each replica and considered zero for this experiment.

EXPERIMENT SETUP

A) Basic Hotstuff with "weights" 1

B) Weighted Hotstuff with binary weights Vmax|Vmin and Δ additional replicas (AWARE style)

EXPERIMENTS

1. → LEADER 0, ONE VIEW {BASIC / WEIGHTED} HOTSTUFF

2. → LEADER 0, ONE VIEW {WEIGHTED / BASIC} HOTSTUFF

↓

+ get the avg difference between the two

→ RUN N (10000) simulations

3. → LEADER ROTATION, 10 VIEWS, BASIC VS WEIGHTED HOTSTUFF
configurable

4. → GENERATE ONE NETWORK SETUP with LEADER 0, ONE VIEW
→ BASIC HOTSTUFF
→ WEIGHTED HOT STUFF
→ SIMULATED ANNEALING to get best configuration of the weights → best latency

5. → LEADER 0, ONE VIEW, RUN N (10 000) simulation of Basic Hotstuff → AVG LATENCY

6. → LEADER 0, ONE VIEW, RUN N (10 000) simulation of weighted Hotstuff → AVG LATENCY