block proposal number

①	①② me-commit prepare	①②③ commit me-commit prepare	①②③④ decide commit me-commit prepare	②③④⑤ decide commit me-commit prepare	- - -
prepare					

VIEW   1   2   3   4   5
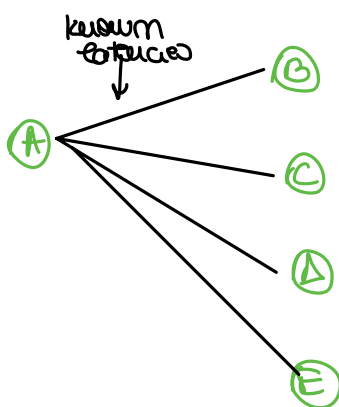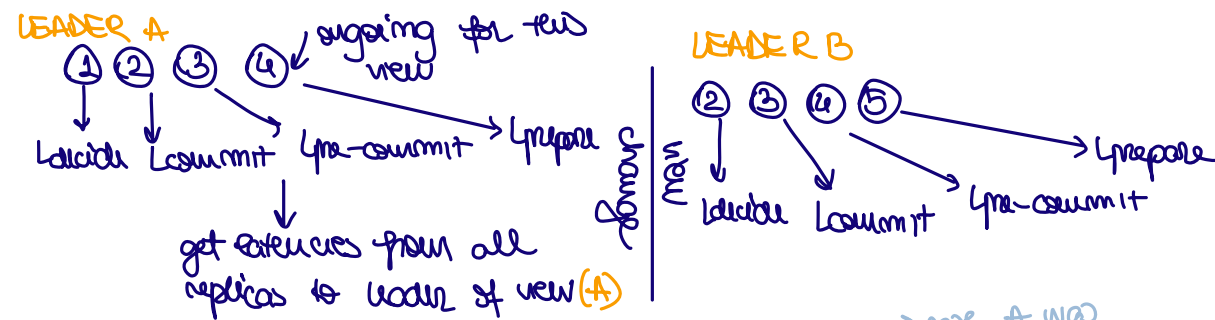
leader change   leader change   leader change   leader change   leader change

BASIC CHAINED HOTSTUFF
↳ "weight" 1 for each replica

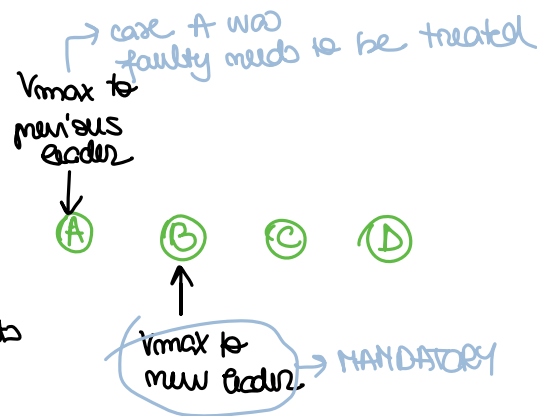# WEIGHTED CHAINED HOTSTUFF

For each new view → NEW LEADER

A) Randomly assign Vmax/Vmin weights

B) Dynamically assign weights → BASED ON SO FAR PERFORMANCE OF REPLICAS → OPTIMISATION PROBLEM

LEADER A
① ② ③ ④ ← ongoing for this view
↓ ↓ ↘   ↘
Ldecide Lcommit Lpre-commit → Lprepare
                      ↓
get latencies from all
replicas to leader of view (A)

LEADER B
② ③ ④ ⑤
↓ ↘ ↘   ↘
Ldecide Lcommit Lpre-commit   Lprepare

(new leader change)

quorum latencies
Ⓐ —→ Ⓑ
    —→ Ⓒ
    —→ Ⓐ
    —→ Ⓔ

randomly chosen
Ⓑ becomes ——————→
leader → how
to assign weights

→ case A was faulty needs to be treated

Vmax to previous leader
↓
Ⓐ   Ⓑ   Ⓒ   Ⓓ
     ↑
Vmax to new leader → MANDATORY

→ RANK nodes based on latencies to the previous leader → fastest performing replicas will get Vmax