

CS-438

Decentralized Systems  
Engineering

Fall 2021

Week 8

# Replication, consensus, consistency

---

"Consensus" aka "agreement":

- Permissioned: well-defined group of nodes
- Permissionless: "anyone can join"

---

Examples: Paxos, Raft (fail-stop) ✓  
PBFT, HotStuff, ... (Byzantine)

---

Goal: given failures, want rest (threshold) of members to agree on a value

○ ○ ○  
○ ○ ○  
○ ○ ○

- safety (consistency): all (live) nodes choose the same value

○ ○ ○

- liveness (progress): all (live) nodes succeed in reaching agreement

# Paxos

naive "straw man" designs

→ A

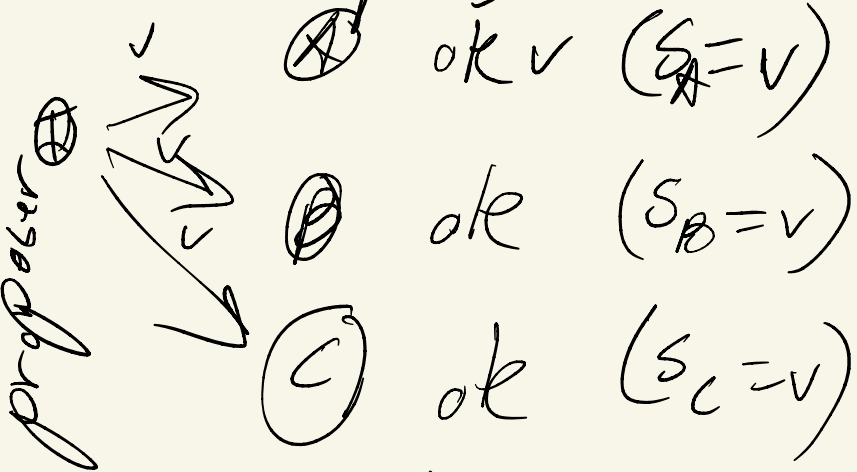
B

→ C

$f$ : # failures = |  $n > 2f$   
 $n$ : # nodes = 3

need "quorum" property:  
at least a majority  $(f+1)$  nodes  
must coordinate in any decision

Strawman: 1: roles: 1. proposer  
acceptors 2. acceptor



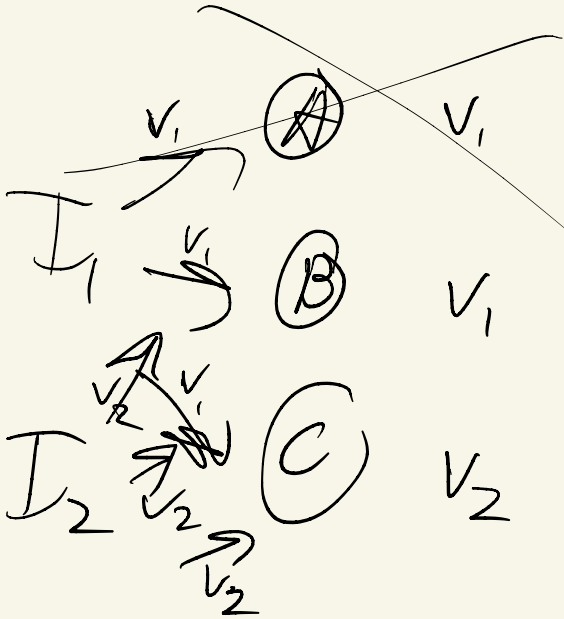
algorithm:

choose value  $v$

ask each acceptor to accept  $v$

Strawman 2:

$I_1$  proposes  $v_1$ , wait for quorum (majority) of acceptors to respond



$I_1 \quad v_1 \rightarrow \textcircled{A} v_1$

$v_2 \rightarrow \textcircled{B} v_2$

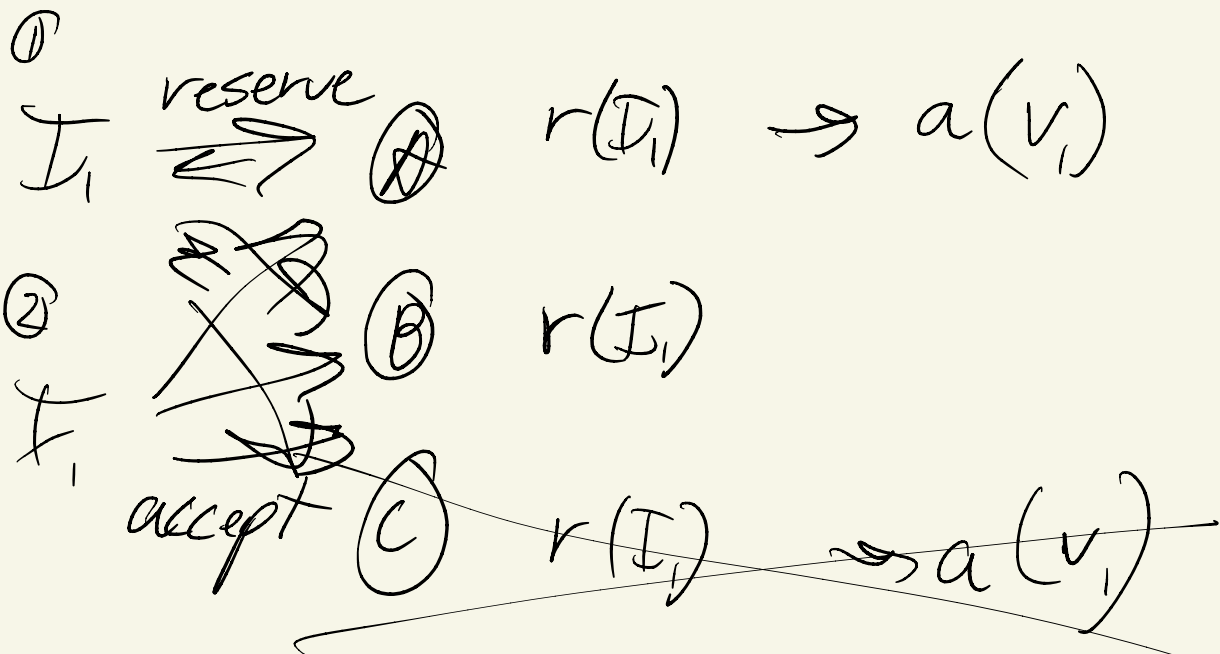
$v_3 \rightarrow \textcircled{C} v_3$

split universe

---

need 2 phases (Paxos)

# Strawman 3:



Paxos / Raft / PBFT / HotStuff / ...  
all leader-based  
but leaders can fail!  
have: 1. leader election  
2. view change

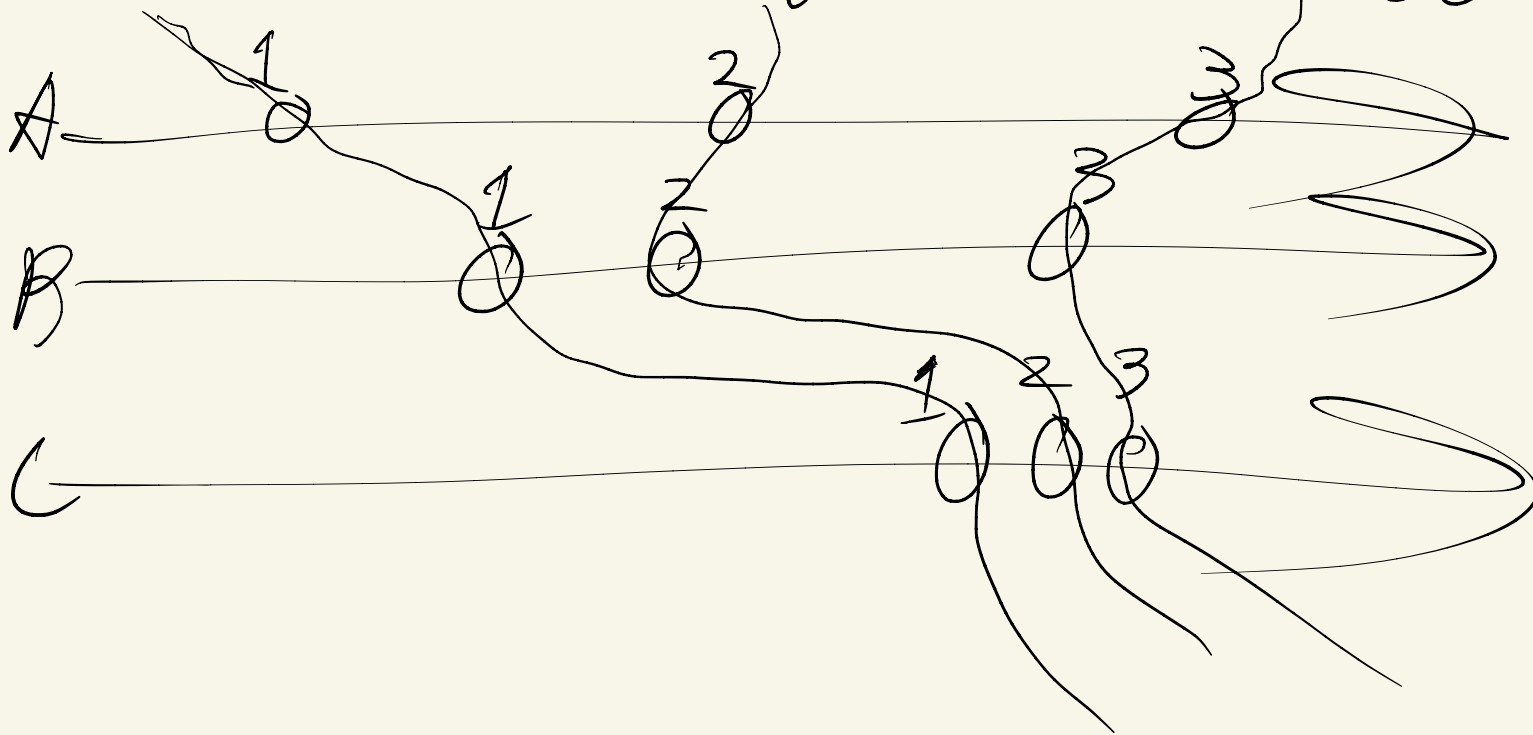
Algorithm:

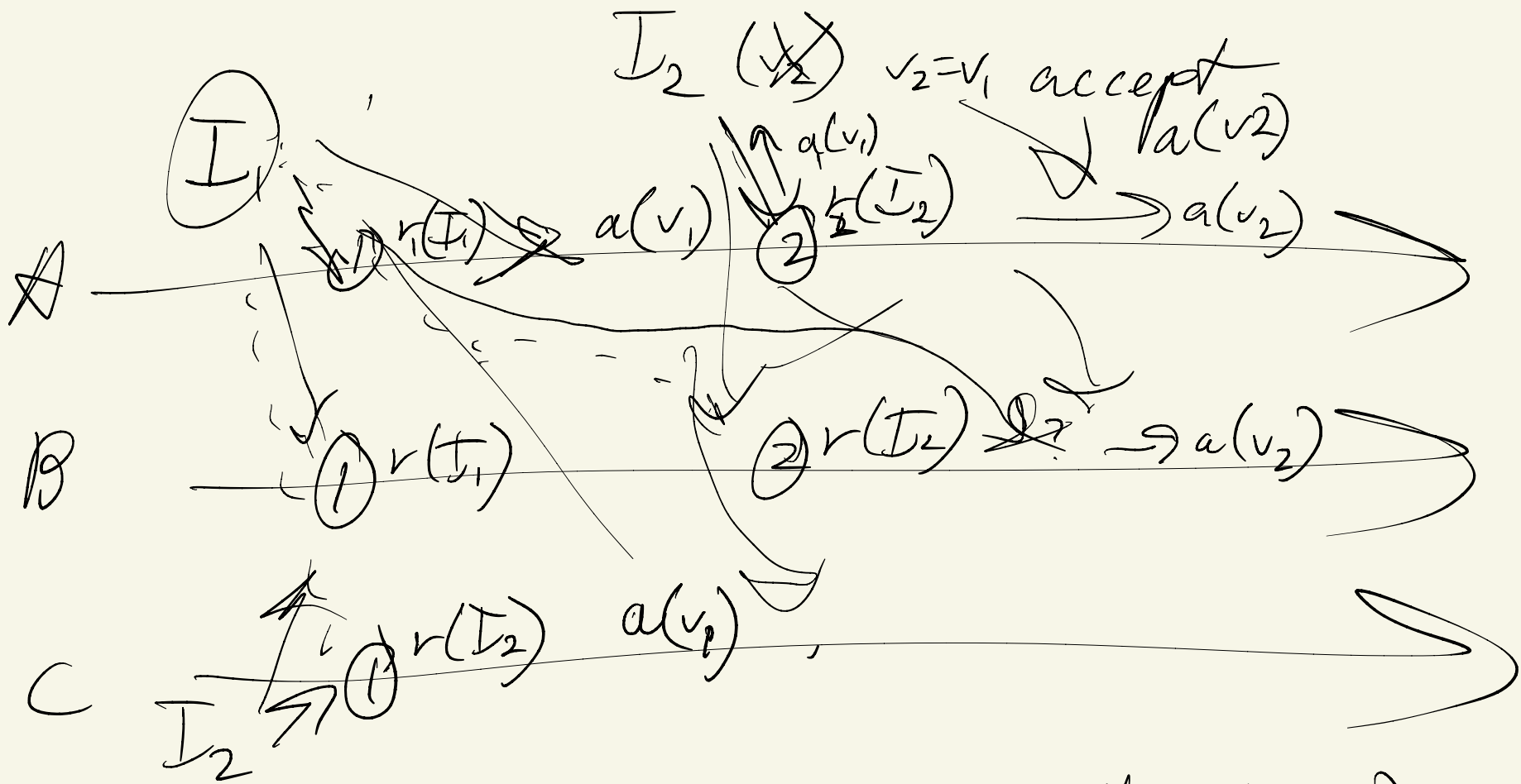
1. ask all nodes to reserve for them  
wait for quorum to answer "yes"
2. ask all nodes to accept  $v_1$   
wait for quorum of "yes"

Retry without confusion  $\rightarrow$  labeling

"try" # / ballot # / step #

- "threshold logical clock"  $\rightarrow$  "design pattern"





Algorithm: (Initiator  $I_1$  with value  $v_1$ )

for each step  $s=1, 2, 3, \dots$

- ask quorum of acceptors to reserve  $s$
- ask quorum of acceptors to accept  $v_1$

on failure