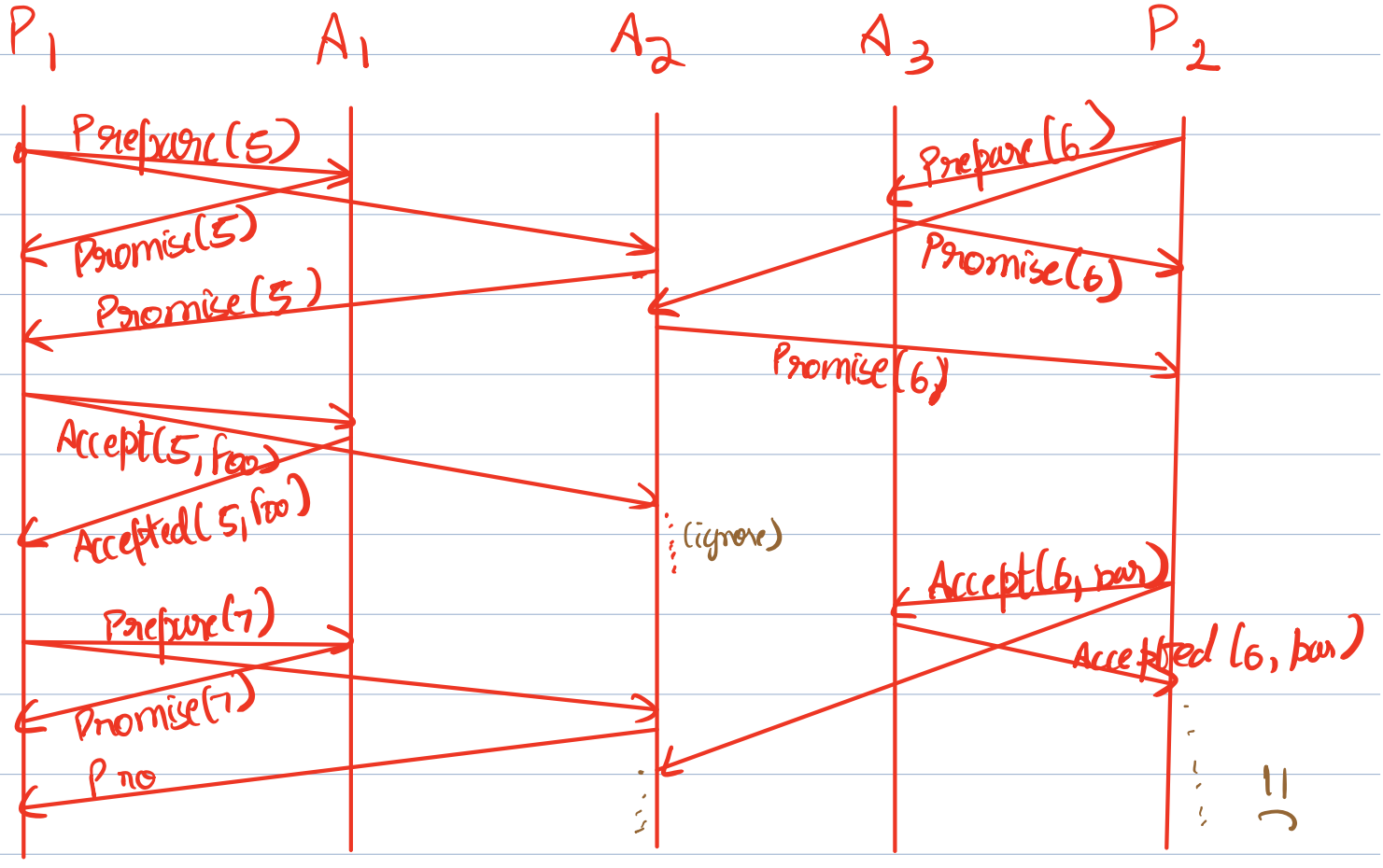
Agenda:

- Paxos: Dueling proposers
- Multi - Paxos
- Fault tolerance in Paxos
- other consenseus protocols
- Passive vs Active Replication
  state Machine Replication




" Dueling Proposers "

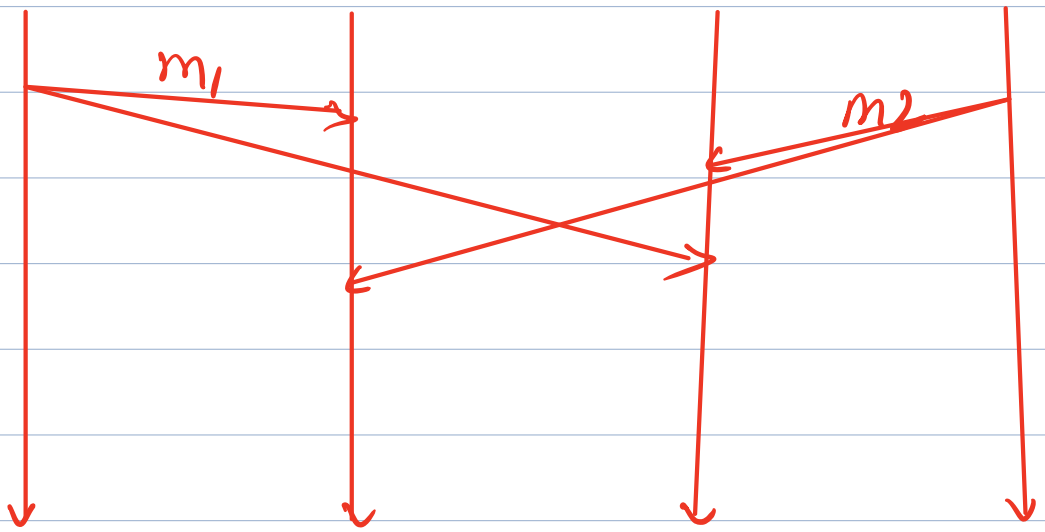
Two proposers are never able to get a Accepted message from majority of acceptors

Why not just have one proposer?

- If we want only one proposer, someone needs to declare themselves as a leader.
- For this, we need to PICK A LEADER. This, in itself, will require a consensus algorithm i.e our consensus algorithm will depend on a consensus algorithm.

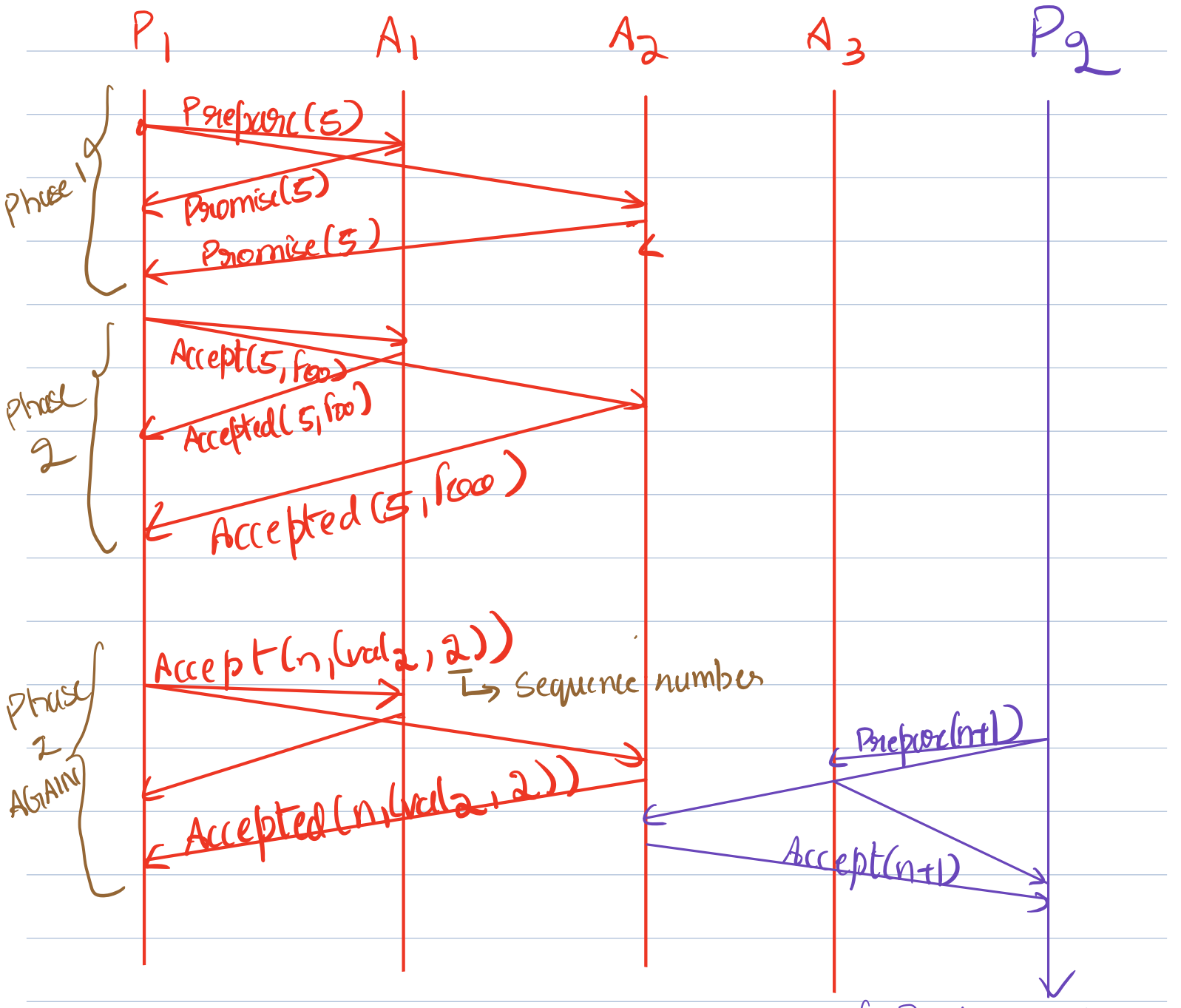
## Multi-Paxos

What if you need to decide on a sequence of values?




violation of totally ordered delivery

# In Paxos, for one value?




Reign of $P_1$ has ended. It will eventually timeout & have to restart Paxos.

For one value, we need to perform 2 round trips with the acceptors. This process is SLOW!!

Can we avoid going through all this for every single value you want to get consensus on?

- Once a process completes one full run of Paxos (Phase 1 & Phase 2), it already has consensus with a majority of acceptors!!
- Due to this, $P_1$ can continue running Phase 2, as long as it doesn't crash!
- If P2 now comes in (as shown in purple), and manages to get consensus from majority of acceptors, then P1 will no longer be in reign. P1 will eventually timeout, I have to start Paxos Phase, again.

- But, till the time the above happens, P1 can continue doing Phase 2. This is called Multi Paxos.

- In practice, Multi-Paxos is used more often.

## Fault tolerance of Paxos

Why can't we have just one acceptor?

- Fault Tolerance! One acceptor might crash!

What if you have 3 acceptors?

- One can crash.

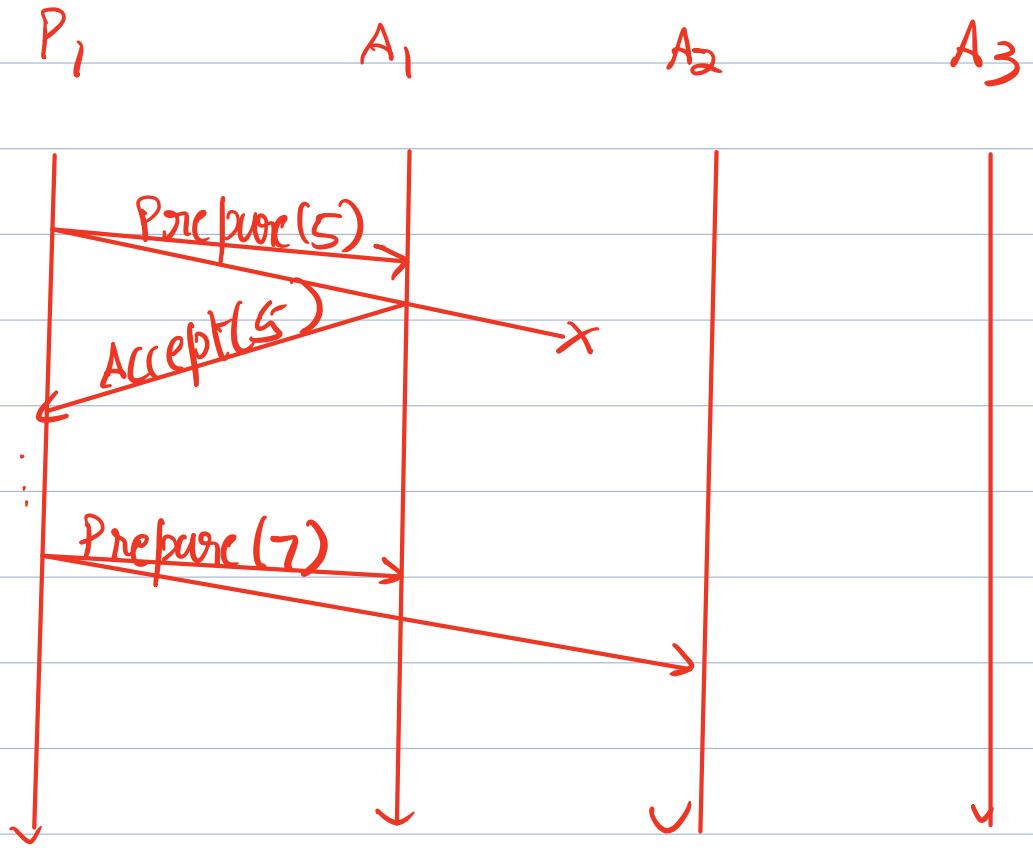
Paxos can tolerate a minority of acceptors can crash.

'f' is the number of acceptor crashes you want to tolerate, 2f+1 is the number of total acceptors required.

What about proposers? How many proposers can crash?

- All but one.
- Just need f+1 total proposers.

# How does Paxos do under omission faults?




If some messages get lost, Paxos can still work

If all messages lost, just send new proposal with higher Proposal number

Paxos does OK under omission faults!

| | safe | not safe |
|---|---|---|
| live | | |
| not live | fail-safe | |

Paxos is here!

## Other Consensus Protocols

- Raft
  - Diego Ongaro & John Ousterhout, 2014
  - Easier to understand than Paxos.
  - Based on Viewstamped Replication.
- Zab (Zookeeper Atomic broadcast) → Also called Totally ordered
  - Yahoo Research, 2000's
- Viewstamped Replication.
  (Brian Oki & Barbara Liskov, 1988

- All of these are for a sequence of values, like Multi-Paxos
- All do leader election

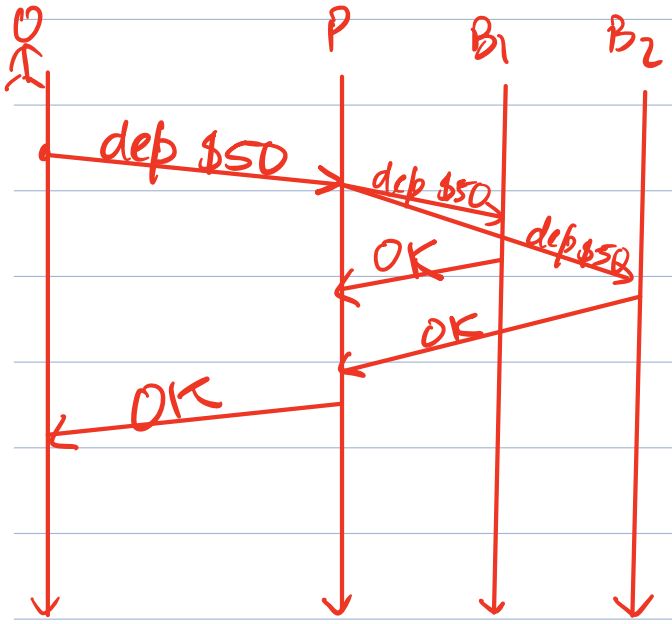
"Viva La Difference" – Van Renesse, Schiper & Schneider (2014)

↓

Compare & contrast differences between different consensus algorithms.

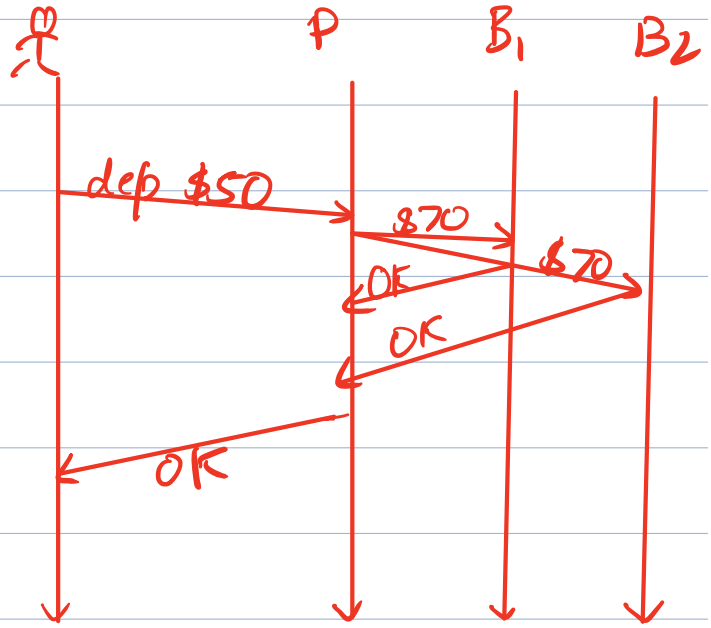
# Active vs Passive Replication

## Active Replication:




Execute an operation on each replica.

## Passive Replication




State update gets sent to backups.

| | PB | CR |
|---|---|---|
| active | ✓ | ✓ |
| passive | ✓ | ✓ |

- Active replication is better when the updated state might be large
- Passive replication could be better if the computation is expensive to do.
- If the operation depends on local process state, passive replication would be better.

Active Replication is also called as state machine Replication.