paxos-algorithm

Nguyen Ngoc Khanh

January 2022 - revisit June 2025

1 The Paxos Algorithm

In Paxos algorithm, there are two main roles: **proposer** and **acceptor**. Let there be Q acceptors and P proposers. The goal of Paxos is to make the Q acceptors agree on a single value $v \in V$ through unreliable communication
The algorithm below describes the simple version of Paxos algorithm

1.1 PROPOSER

Let's label each proposer by an integer $p \in [0, P-1]$

1.1.1 phase 1: prepare

input: a value $v \in V$ and a round number $r \in \mathbb{N}$
1 choose the proposal number $n := rP + p$
2 broadcast the prepare message to all acceptors

$$\text{PREPARE\_REQUEST } (n)$$

3 if it receives responses from a majority of acceptors, do **phase 2**

1.1.2 phase 2: accept

4 receives

$$\text{PREPARE\_RESPONSE } \{(n_j, v_j)\}_{j \in J \subseteq [0, Q-1]}$$

5 if all $v_j = \text{null}$, pick $w := v$. otherwise, pick the non-null value $w := v_j$ corresponds to the highest n_j
6 broadcast the accept message to all acceptors

$$\text{ACCEPT\_REQUEST } (n, w)$$

7 (optional) if proposer receives the responses from majority of acceptors then it knows that the consensus has reached to value w

1.2 ACCEPTOR

an acceptor j holds two values $(n_j, v_j) \in \mathbb{N} \times (V \cap \{\text{null}\})$ in its stable storage. Initially, $(n_j, v_j) = (0, \text{null})$

1.2.1 on receiving prepare request

input: PREPARE_REQUEST (n)
1 if $n_j < n$, then set its state into (n, v_j) and reply

PREPARE_RESPONSE (n_j, v_j)

1.2.2 on receiving accept request

input: ACCEPT_REQUEST (n, w)
1 if $n_j \leq n$, the set its state into (n, w) and reply

ACCEPT_RESPONSE

2 PROOF

Theorem 2.1
If an accept request (m, u) is accepted majority of acceptors, then any accept request (n, v) with $m < n$ satisfies $v = u$.

Proof. We will prove by induction
induction step: Suppose that for every $k = m, ..., n-1$, any accept request (k, v) satisfies $v = u$.
Let A be the majority set of acceptors that accepted (m, u). Let accept request (n, v) be sent by proposer p.

1. The prepare request n must be acknowledged from some majority set of acceptors, namely B
2. Let $q \in A \cap B$ be an acceptor, since $m < n$, then q accepts (m, u) **before** receiving prepare request n
3. Up receiving prepare request n, q reply prepare response (k_1, u_1) for some $k_1 \in [m, n)$.
4. By induction hypothesis, (k_1, u_1) satisfies $u_1 = u$
5. **After** receiving reponse for prepare request n, at proposer p, let (k_2, u_2) be the prepare response with non-null value and highest proposer number.
6. Then, $k_1 \leq k_2$ since (k_1, u) is one of the prepare responses with non-null value.
7. On the other hand, $k_1 < n$ since (k_1, u_1) is from an acceptor $q_1 \in B$ replying to a prepare request n.
8. By induction hypothesis, (k_2, u_2) satisfies $u_2 = u$
9. $u_2 = u$ implies the accept request (n, u_2) sent by p must satisfies $u_2 = u$

□

Corollary 2.2
If an accept request (m, u) is accepted majority of acceptors, then acceptors reach consensus at value u