MODULE *FastPaxos*

This is a simplified specification of *Leslie Lamport*'s Fast *Paxos* protocol. The following papers, Fast *Paxos* by *Leslie Lamport* and Fast *Paxos* Made Easy: Theory and Implementation by *Zhao Wenbing* was referenced in writing this specification.

This simplified specification was written by *Lim Ngian Xin Terry* & *Gaurav Gandhi*.

The following assumptions are made in this simplified specification.

1. There is a unique coordinator in the system. Therefore, Phase $1a$ and $1b$ can be omitted.
2. All agents in the system can communicate with one another.
3. Agents must have some stable storage that survives failure and restarts. An agent restores its state from stable storage when it restarts, so the failure of an agent is indistinguishable from its simply pausing. There is thus no need to model failures explicitly.

EXTENDS *Paxos*

CONSTANTS *FastQuorums; FastBallots*

VARIABLES *cValue* Value chosen by coordinator.

$ClassicBallots \triangleq Ballots \setminus FastBallots$ The set of ballots of classic rounds.

$$
\begin{aligned}
&FastAssume \triangleq \\
&\quad \land \forall\, q \in FastQuorums : q \subseteq Replicas \\
&\quad \land \forall\, q;\ r \in FastQuorums : q \cap r \neq \{\} \\
&\quad \land \forall\, q \in FastQuorums : (3 * Cardinality(Replicas)) \div 4 \leq Cardinality(q) \\
&\quad \land \forall\, q \in Quorums : \forall\, r;\ s \in FastQuorums : q \cap r \cap s \neq \{\}
\end{aligned}
$$

ASSUME $PaxosAssume \land FastAssume$

$$IsMajorityValue(M;\ v) \triangleq Cardinality(M) \div 2 < Cardinality(\{m \in M : m.value = v\})$$

Phase $2a$ (Fast):

The coordinator starts a fast round by sending a $P2a$ "Any" message, if no other values has been proposed before.

$$
\begin{aligned}
&FastAny \triangleq \\
&\quad \land \text{UNCHANGED } \langle decision;\ maxBallot;\ maxVBallot;\ maxValue;\ cValue \rangle \\
&\quad \land \exists\, f \in FastBallots : \\
&\qquad \land SendMessage([type \mapsto \text{“P2a”};\ ballot \mapsto f;\ value \mapsto any])
\end{aligned}
$$

Phase $2b$ (Fast):

Acceptors can reply to a $P2a$ "Any" message with a $P2b$ message containing their proposed value.

$$
\begin{aligned}
&FastPropose \triangleq \\
&\quad \land \text{UNCHANGED } \langle decision;\ cValue \rangle \\
&\quad \land \exists\, a \in Replicas;\ m \in p2aMessages;\ v \in Values :
\end{aligned}
$$

$\land\ m.value = any$
$\land\ maxBallot[a] \leq m.ballot$
$\land\ maxValue[a] = none \lor maxValue[a] = v$
$\land\ maxBallot' = [maxBallot \text{ EXCEPT } ![a] = m.ballot]$
$\land\ maxVBallot' = [maxVBallot \text{ EXCEPT } ![a] = m.ballot]$
$\land\ maxValue' = [maxValue \text{ EXCEPT } ![a] = v]$
$\land\ \forall\, n \in p2bMessages : \neg(n.ballot = m.ballot \land n.acceptor = a)$
$\land\ SendMessage([type \mapsto \text{“P2b”};$
$ballot \mapsto m.ballot;$
$acceptor \mapsto a;$
$value \mapsto v])$

A value is chosen if a fast quorum of acceptors proposed that value in a fast round.

Because the quorum size of a fast round and classic round is different, we assume that the acceptor distinguishes a fast round and classic round based on the $P2a$ message it receives. If the $P2a$ message contains the special value “any”, it is a fast round. Else it is a classic round.

$FastDecide \triangleq$
$\land$ UNCHANGED $\langle messages; maxBallot; maxVBallot; maxValue; cValue\rangle$
$\land\ \exists\, b \in FastBallots;\ q \in FastQuorums :$
LET $M \triangleq \{m \in p2bMessages : m.ballot = b \land m.acceptor \in q\}$
$V \triangleq \{w \in Values : \exists\, m \in M : w = m.value\}$
IN $\land\ \forall\, a \in q : \exists\, m \in M : m.acceptor = a$
$\land\ 1 = Cardinality(V)$
$\land\ \exists\, m \in M : decision' = m.value$

Phase $2a$ (Classic)

If more than one value has been proposed, the collision is resolved using the following rules:

1. If the proposals contain different values, a value must be selected if the majority of acceptors in the fast quorum have casted a vote for that value.

2. Otherwise, the coordinator is free to select any value.

$ClassicAccept \triangleq$
$\land$ UNCHANGED $\langle decision; maxBallot; maxVBallot; maxValue\rangle$
$\land\ \exists\, b \in ClassicBallots;\ f \in FastBallots;\ q \in FastQuorums;\ v \in Values :$
$\land\ f < b$ There was a fast round before this classic round.
$\land\ cValue = none \lor cValue = v$
$\land\ cValue' = v$
$\land\ \forall\, m \in p2aMessages : m.ballot \neq b$
$\land$ LET $M \triangleq \{m \in p2bMessages : m.ballot = f \land m.acceptor \in q\}$
$V \triangleq \{w \in Values : \exists\, m \in M : w = m.value\}$
IN $\land\ \forall\, a \in q : \exists\, m \in M : m.acceptor = a$
$\land\ 1 < Cardinality(V)$ Collision occured.
$\land$ IF $\exists\, w \in V : IsMajorityValue(M; w)$
THEN $IsMajorityValue(M; v)$ Choose majority in quorum.
ELSE $v \in V$ Choose any.

$$\land\ SendMessage([type \mapsto \text{“P2a”};\ ballot \mapsto b;\ value \mapsto v])$$

Phase 2*b* (Classic)

Same as in *Paxos*.

$$ClassicAccepted \triangleq \land \text{UNCHANGED } \langle cValue \rangle \land PaxosAccepted$$

Consensus is achieved when a majority of acceptors accept the same ballot number.

Functionally similar to *PaxosDecide* in *Paxos.tla*, but we also have to ensure that it can only occur in classic rounds and not fast rounds.

$$\begin{array}{l} ClassicDecide \triangleq \\ \quad \land \text{UNCHANGED } \langle messages;\ maxBallot;\ maxVBallot;\ maxValue;\ cValue \rangle \\ \quad \land \exists\, b \in ClassicBallots;\ q \in Quorums : \\ \qquad \text{LET } M \triangleq \{m \in p2bMessages : m.ballot = b \land m.acceptor \in q\} \\ \qquad \text{IN } \land \forall\, a \in q : \exists\, m \in M : m.acceptor = a \\ \qquad\quad \land \exists\, m \in M : decision' = m.value \end{array}$$

$$\begin{array}{l} FastTypeOK \triangleq \land PaxosTypeOK \\ \qquad\qquad\quad \land cValue \in Values \cup \{none\} \end{array}$$

$$\begin{array}{l} FastInit \triangleq \land PaxosInit \\ \qquad\quad \land cValue = none \end{array}$$

$$\begin{array}{l} FastNext \triangleq \lor FastAny \\ \qquad\quad \lor FastPropose \\ \qquad\quad \lor FastDecide \\ \qquad\quad \lor ClassicAccept \\ \qquad\quad \lor ClassicAccepted \\ \qquad\quad \lor ClassicDecide \end{array}$$

$$\begin{array}{l} FastSpec \triangleq \land FastInit \\ \qquad\quad \land \Box[FastNext]_{\langle messages,\, decision,\, maxBallot,\, maxVBallot,\, maxValue,\, cValue \rangle} \\ \qquad\quad \land \text{SF}_{\langle messages,\, decision,\, maxBallot,\, maxVBallot,\, maxValue,\, cValue \rangle}(FastDecide) \\ \qquad\quad \land \text{SF}_{\langle messages,\, decision,\, maxBallot,\, maxVBallot,\, maxValue,\, cValue \rangle}(ClassicDecide) \end{array}$$

Non-triviality safety property: Only proposed values can be learnt.

$$\begin{array}{l} FastNontriviality \triangleq \lor decision = none \\ \qquad\qquad\qquad\quad \lor \exists\, m \in p2bMessages : m.value = decision \land m.ballot \in FastBallots \end{array}$$