MODULE *AbstractBalloting*

This is a formalization of SCP 's abstract balloting protocol described in Section 3.5 of the IETF draft at:

https://datatracker.ietf.org/doc/html/draft-mazieres-dinrg-scp-05#section-3.5

The goal if to then refine this specification to one that closely matches the concrete SCP protocol.

We provide an inductive invariant showing that, by following the 2 "restrictions on voting" described in Section 3.5 of the above document, safety is guaranteed.

Note that it is not true that a validator never votes to commit and abort the same ballot. This can happen when a validator votes to commit a ballot, but then accepts to abort it because a blocking set accepted to abort it. Moreover, it is necessary for liveness to allow this.

EXTENDS *DomainModel*

VARIABLES
voteToAbort
, *acceptedAborted*
, *voteToCommit*
, *acceptedCommitted*
, *externalized*
, *byz*

$$
\begin{aligned}
&\mathit{TypeOK} \triangleq \\
&\quad \land\ \mathit{voteToAbort} \in [N \to \text{SUBSET } \mathit{Ballot}] \\
&\quad \land\ \mathit{acceptedAborted} \in [N \to \text{SUBSET } \mathit{Ballot}] \\
&\quad \land\ \mathit{voteToCommit} \in [N \to \text{SUBSET } \mathit{Ballot}] \\
&\quad \land\ \mathit{acceptedCommitted} \in [N \to \text{SUBSET } \mathit{Ballot}] \\
&\quad \land\ \mathit{externalized} \in [N \to \text{SUBSET } \mathit{Ballot}] \\
&\quad \land\ \mathit{byz} \in \text{SUBSET } N
\end{aligned}
$$

$$
\begin{aligned}
&\mathit{Init} \triangleq \\
&\quad \land\, \mathit{voteToAbort} = [n \in N \mapsto \{\}] \\
&\quad \land\, \mathit{acceptedAborted} = [n \in N \mapsto \{\}] \\
&\quad \land\, \mathit{voteToCommit} = [n \in N \mapsto \{\}] \\
&\quad \land\, \mathit{acceptedCommitted}\ = [n \in N \mapsto \{\}] \\
&\quad \land\, \mathit{externalized} = [n\ \ \in N \mapsto \{\}] \\
&\quad \land\, \mathit{byz} \in \mathit{FailProneSet}
\end{aligned}
$$

$$
\begin{aligned}
&\mathit{IsPrepared}(n,\ b1) \triangleq \\
&\qquad \lor\ \forall\, b2 \in \mathit{Ballot} : \mathit{LessThanAndIncompatible}(b2,\ b1) \Rightarrow \\
&\qquad\qquad \exists\, Q \in \mathit{Quorum} : \forall\, m \in Q \setminus \mathit{byz} : b2 \in \mathit{acceptedAborted}[m] \\
&\qquad \lor\ b1.\mathit{counter} = 1 \quad \text{Initially, we can skip the prepare phase} \\
&\qquad \lor \exists\, \mathit{cnt} \in \mathit{BallotNumber} : \\
&\qquad\quad \land\ \mathit{cnt} < b1.\mathit{counter} \\
&\qquad\quad \land [\mathit{counter} \mapsto \mathit{cnt},\ \mathit{value} \mapsto b1.\mathit{value}] \in \mathit{acceptedCommitted}[n] \\
&\qquad\quad \text{NOTE: is } \mathit{cnt} < b1.\mathit{counter} \text{ necessary?}
\end{aligned}
$$

$Step(n) \triangleq$

$\land$ UNCHANGED $\langle byz \rangle$

NOTE for TLC, we must update $acceptedAborted$ before $voteToAbort$,
because updating $voteToAbort$ depends on $acceptedAborted'$:

$\land \exists\, B \in \text{SUBSET } Ballot :$
$\quad \land \forall\, b \in B :$
$\quad\quad \land \lor \exists\, Q \in Quorum : \forall\, m \in Q \setminus byz : b \in voteToAbort[m] \cup acceptedAborted[m]$
$\quad\quad\quad \lor \exists\, Bl \in BlockingSet : \forall\, m \in Bl \setminus byz : b \in acceptedAborted[m]$
$\quad \land acceptedAborted' = [acceptedAborted \text{ EXCEPT } ![n] = @ \cup B]$
$\land \exists\, B \in \text{SUBSET } Ballot :$
$\quad \land \forall\, b \in B : b \notin voteToCommit[n] \lor b \in acceptedAborted'[n]$
$\quad \land voteToAbort' = [voteToAbort \text{ EXCEPT } ![n] = @ \cup B]$

NOTE for TLC, we must update $acceptedCommitted$ before $voteToCommit$,
because updating $voteToCommit$ depends on $acceptedCommitted'$:

$\land \exists\, B \in \text{SUBSET } Ballot :$
$\quad \land \forall\, b \in B :$
$\quad\quad \land \lor \exists\, Q \in Quorum : \forall\, m \in Q \setminus byz : b \in voteToCommit[m] \cup acceptedCommitted[m]$
$\quad\quad\quad \lor \exists\, Bl \in BlockingSet : \forall\, m \in Bl \setminus byz : b \in acceptedCommitted[m]$
$\quad \land acceptedCommitted' = [acceptedCommitted \text{ EXCEPT } ![n] = @ \cup B]$
$\land \exists\, B \in \text{SUBSET } Ballot :$
$\quad \land \forall\, b \in B :$
$\quad\quad \land b.counter > 0$ we start at ballot 1
if the ballot is already aborted, don't vote to commit
(using the primed version ensures we don't vote to commit and abort at the same time):
$\quad\quad \land b \notin voteToAbort'[n] \cup acceptedAborted'[n]$
the prime allows us to consider prepared something we accepted committed in this very step:
$\quad\quad \land IsPrepared(n, b)'$
$\quad \land voteToCommit' = [voteToCommit \text{ EXCEPT } ![n] = @ \cup B]$
we vote to commit at most one value per ballot number:
$\quad \land \forall\, b1, b2 \in voteToCommit'[n] : b1.counter = b2.counter \Rightarrow b1.value = b2.value$
$\land \exists\, B \in \text{SUBSET } Ballot :$
$\quad \land \forall\, b \in B : \exists\, Q \in Quorum :$
$\quad\quad \forall\, m \in Q \setminus byz : b \in acceptedCommitted[m]$
$\quad \land externalized' = [externalized \text{ EXCEPT } ![n] = @ \cup B]$

$ByzantineHavoc \triangleq$

$\land \exists\, x \in [byz \to \text{SUBSET } Ballot] :$
$\quad voteToAbort' = [n \in N \mapsto \text{IF } n \in byz \text{ THEN } x[n] \text{ ELSE } voteToAbort[n]]$
$\land \exists\, x \in [byz \to \text{SUBSET } Ballot] :$
$\quad acceptedAborted' = [n \in N \mapsto \text{IF } n \in byz \text{ THEN } x[n] \text{ ELSE } acceptedAborted[n]]$
$\land \exists\, x \in [byz \to \text{SUBSET } Ballot] :$
$\quad voteToCommit' = [n \in N \mapsto \text{IF } n \in byz \text{ THEN } x[n] \text{ ELSE } voteToCommit[n]]$
$\land \exists\, x \in [byz \to \text{SUBSET } Ballot] :$
$\quad acceptedCommitted' = [n \in N \mapsto \text{IF } n \in byz \text{ THEN } x[n] \text{ ELSE } acceptedCommitted[n]]$
$\land$ UNCHANGED $\langle externalized, byz \rangle$

$$
\begin{array}{l}
\mathit{Next} \triangleq \\
\quad \lor \exists\, n \in N : \mathit{Step}(n) \\
\quad \lor \ \mathit{ByzantineHavoc}
\end{array}
$$

$$
\mathit{vars} \triangleq \langle \mathit{voteToAbort},\ \mathit{acceptedAborted},\ \mathit{voteToCommit},\ \mathit{acceptedCommitted},\ \mathit{externalized},\ \mathit{byz} \rangle
$$

$$
\mathit{Spec} \triangleq \mathit{Init} \land \Box[\mathit{Next}]_{\mathit{vars}}
$$

$$
\begin{array}{l}
\mathit{Safety} \triangleq \\
\quad \forall\, n1,\ n2 \in N \setminus \mathit{byz} : \forall\, b1,\ b2 \in \mathit{Ballot} : \\
\qquad b1 \in \mathit{externalized}[n1] \land b2 \in \mathit{externalized}[n2] \Rightarrow b1.\mathit{value} = b2.\mathit{value}
\end{array}
$$

Inductive invariant proving safety:

$$
\begin{array}{l}
\mathit{Invariant} \triangleq \\
\quad \land \ \mathit{TypeOK} \\
\quad \land \ \mathit{byz} \in \mathit{FailProneSet} \\
\quad \land \ \forall\, n \in N \setminus \mathit{byz} : \\
\qquad \land \ \forall\, b \in \mathit{Ballot} : \\
\qquad\quad \land \ b \in \mathit{voteToCommit}[n] \Rightarrow b \notin \mathit{voteToAbort}[n] \lor b \in \mathit{acceptedAborted}[n] \\
\qquad\quad \land \ b \in \mathit{voteToCommit}[n] \cup \mathit{acceptedCommitted}[n] \cup \mathit{externalized}[n] \Rightarrow b.\mathit{counter} > 0 \\
\qquad\quad \land \ \forall\, b2 \in \mathit{Ballot} : \\
\qquad\qquad b \in \mathit{voteToCommit}[n] \land b2 \in \mathit{voteToCommit}[n] \land b \neq b2 \Rightarrow b.\mathit{counter} \neq b2.\mathit{counter} \\
\qquad\quad \land \ b \in \mathit{acceptedAborted}[n] \Rightarrow \exists\, Q \in \mathit{Quorum} : \\
\qquad\qquad \forall\, m \in Q \setminus \mathit{byz} : b \in \mathit{voteToAbort}[m] \\
\qquad\quad \land \ b \in \mathit{acceptedCommitted}[n] \Rightarrow \exists\, Q \in \mathit{Quorum} : \\
\qquad\qquad \forall\, m \in Q \setminus \mathit{byz} : b \in \mathit{voteToCommit}[m] \\
\qquad\quad \land \ b \in \mathit{externalized}[n] \Rightarrow \exists\, Q \in \mathit{Quorum} : \\
\qquad\qquad \forall\, m \in Q \setminus \mathit{byz} : b \in \mathit{acceptedCommitted}[m] \\
\qquad\quad \land \ b \in \mathit{voteToCommit}[n] \Rightarrow \\
\qquad\qquad \lor \ b.\mathit{counter} = 1 \\
\qquad\qquad \lor \ \forall\, b2 \in \mathit{Ballot} : \mathit{LessThanAndIncompatible}(b2,\ b) \Rightarrow \\
\qquad\qquad\qquad \exists\, Q \in \mathit{Quorum} : \forall\, m \in Q \setminus \mathit{byz} : b2 \in \mathit{acceptedAborted}[m] \\
\qquad\qquad \lor \ \exists\, \mathit{cnt} \in \mathit{BallotNumber} : \\
\qquad\qquad\quad \land \ \mathit{cnt} < b.\mathit{counter} \\
\qquad\qquad\quad \land \ [\mathit{counter} \mapsto \mathit{cnt},\ \mathit{value} \mapsto b.\mathit{value}] \in \mathit{acceptedCommitted}[n] \\
\qquad\quad \land \ b \in \mathit{acceptedAborted}[n] \Rightarrow \forall\, Q \in \mathit{Quorum} : \exists\, m \in Q \setminus \mathit{byz} : b \notin \mathit{voteToCommit}[m] \\
\quad \land \ \mathit{Safety}
\end{array}
$$