MODULE *EagerVoting*

EXTENDS *Sets*

CONSTANT *Value*, *Acceptor*, *Quorum*

ASSUME *QuorumAssumption* $\triangleq$
$\land \quad \forall\, Q \in Quorum : Q \subseteq Acceptor$
$\land \quad \forall\, Q1,\, Q2 \in Quorum : Q1 \cap Q2 \neq \{\}$

THEOREM *QuorumNonEmpty* $\triangleq \forall\, Q \in Quorum : Q \neq \{\}$
BY *QuorumAssumption*

$Ballot \triangleq Nat$

VARIABLES *votes*, *maxBal*

$TypeOK \triangleq \land votes \in [Acceptor \to \text{SUBSET } (Ballot \times Value)]$
$\land maxBal \in [Acceptor \to Ballot \cup \{-1\}]$

$VotedFor(a,\, b,\, v) \triangleq \langle b,\, v\rangle \in votes[a]$

$DidNotVoteAt(a,\, b) \triangleq \forall\, v \in Value : \neg VotedFor(a,\, b,\, v)$

$ShowsSafeAt(Q,\, b,\, v) \triangleq$
$\land \forall\, a \in Q : maxBal[a] \geq b$ have promised
$\land \exists\, c \in -1 \,..\, (b-1) :$
$\land (c \neq -1) \Rightarrow \exists\, a \in Q : VotedFor(a,\, c,\, v)$
$\land \forall\, d \in (c+1) \,..\, (b-1),\, a \in Q : DidNotVoteAt(a,\, d)$

$Init \triangleq$
$\land votes = [a \in Acceptor \mapsto \{\}]$
$\land maxBal = [a \in Acceptor \mapsto -1]$

$IncreaseMaxBal(a,\, b) \triangleq$
$\land b > maxBal[a]$
$\land maxBal' = [maxBal \text{ EXCEPT } ![a] = b]$ make promise
$\land$ UNCHANGED *votes*

The only difference between *EagerVoting* and Voting is:

In Voting, we have $maxBal' = [maxBal \text{ EXCEPT } ![a] = b]$.

$VoteFor(a,\, b,\, v) \triangleq$
$\land \quad maxBal[a] \leq b$ keep promise
$\land \quad \forall\, vt \in votes[a] : vt[1] \neq b$
$\land \quad \forall\, c \in Acceptor \setminus \{a\} :$
$\forall\, vt \in votes[c] : (vt[1] = b) \Rightarrow (vt[2] = v)$
$\land \quad \exists\, Q \in Quorum : ShowsSafeAt(Q,\, b,\, v)$ safe to vote
$\land \quad votes' = [votes \text{ EXCEPT } ![a] = votes[a] \cup \{\langle b,\, v\rangle\}]$ vote

$\land \quad \exists\, c \in \mathit{Ballot} :$
$\quad\quad \land\, c \geq b$
$\quad\quad \land\, \mathit{maxBal}' = [\mathit{maxBal} \text{ EXCEPT } ![a] = c]$ make promise

---

$\mathit{Next} \triangleq$
$\quad \exists\, a \in \mathit{Acceptor},\, b \in \mathit{Ballot} :$
$\quad\quad \lor\, \mathit{IncreaseMaxBal}(a,\, b)$
$\quad\quad \lor\, \exists\, v \in \mathit{Value} : \mathit{VoteFor}(a,\, b,\, v)$

$\mathit{Spec} \triangleq \mathit{Init} \land \Box[\mathit{Next}]_{\langle \mathit{votes},\, \mathit{maxBal} \rangle}$

$\mathit{ChosenAt}(b,\, v) \triangleq$
$\quad \exists\, Q \in \mathit{Quorum} : \forall\, a \in Q : \mathit{VotedFor}(a,\, b,\, v)$

$\mathit{chosen} \triangleq \{v \in \mathit{Value} : \exists\, b \in \mathit{Ballot} : \mathit{ChosenAt}(b,\, v)\}$

$\mathit{Consistency} \triangleq \mathit{chosen} = \{\} \lor \exists\, v \in \mathit{Value} : \mathit{chosen} = \{v\}$ $\mathit{Cardinality}(\mathit{chosen}) \leq 1$

$\mathit{CannotVoteAt}(a,\, b) \triangleq$
$\quad \land\, \mathit{maxBal}[a] > b$
$\quad \land\, \mathit{DidNotVoteAt}(a,\, b)$

$\mathit{NoneOtherChoosableAt}(b,\, v) \triangleq$
$\quad \exists\, Q \in \mathit{Quorum} :$
$\quad\quad \forall\, a \in Q : \mathit{VotedFor}(a,\, b,\, v) \lor \mathit{CannotVoteAt}(a,\, b)$

$\mathit{SafeAt}(b,\, v) \triangleq$
$\quad \forall\, c \in 0 \,..\, (b-1) : \mathit{NoneOtherChoosableAt}(c,\, v)$

$\mathit{VotesSafe} \triangleq$
$\quad \forall\, a \in \mathit{Acceptor},\, b \in \mathit{Ballot},\, v \in \mathit{Value} :$
$\quad\quad \mathit{VotedFor}(a,\, b,\, v) \Rightarrow \mathit{SafeAt}(b,\, v)$

$\mathit{OneVote} \triangleq$
$\quad \forall\, a \in \mathit{Acceptor},\, b \in \mathit{Ballot},\, v,\, w \in \mathit{Value} :$
$\quad\quad \mathit{VotedFor}(a,\, b,\, v) \land \mathit{VotedFor}(a,\, b,\, w) \Rightarrow (v = w)$

$\mathit{OneValuePerBallot} \triangleq$
$\quad \forall\, a1,\, a2 \in \mathit{Acceptor},\, b \in \mathit{Ballot},\, v1,\, v2 \in \mathit{Value} :$
$\quad\quad \mathit{VotedFor}(a1,\, b,\, v1) \land \mathit{VotedFor}(a2,\, b,\, v2) \Rightarrow (v1 = v2)$

$\mathit{Inv} \triangleq \mathit{TypeOK} \land \mathit{VotesSafe} \land \mathit{OneValuePerBallot}$

THEOREM $\mathit{AllSafeAtZero} \triangleq \forall\, v \in \mathit{Value} : \mathit{SafeAt}(0,\, v)$
BY DEF SafeAt

THEOREM $\mathit{ChoosableThm} \triangleq$
$\quad\quad \forall\, b \in \mathit{Ballot},\, v \in \mathit{Value} :$

$ChosenAt(b,\, v) \Rightarrow NoneOtherChoosableAt(b,\, v)$
BY DEF $ChosenAt$, $NoneOtherChoosableAt$

THEOREM $OneVoteThm \triangleq OneValuePerBallot \Rightarrow OneVote$
BY DEF $OneValuePerBallot$, $OneVote$

THEOREM $VotesSafeImpliesConsistency \triangleq$
ASSUME $VotesSafe$, $OneVote$, $chosen \neq \{\}$
PROVE $\exists\, v \in Value : chosen = \{v\}$
$\langle 1\rangle 1$. PICK $v \in Value : v \in chosen$
BY DEF $chosen$
$\langle 1\rangle 2$. SUFFICES ASSUME NEW $w \in chosen$
PROVE $w = v$
BY $\langle 1\rangle 1$, $\langle 1\rangle 2$
$\langle 1\rangle 3$. ASSUME NEW $b1 \in Ballot$, NEW $b2 \in Ballot$, $b1 < b2$,
NEW $v1 \in Value$, NEW $v2 \in Value$,
$ChosenAt(b1,\, v1) \land ChosenAt(b2,\, v2)$
PROVE $v1 = v2$
$\langle 2\rangle 1$. $SafeAt(b2,\, v2)$
BY $\langle 1\rangle 3$, $QuorumAssumption$, SMT DEF $ChosenAt$, $VotesSafe$
$\langle 2\rangle 2$. QED
BY $\langle 1\rangle 3$, $\langle 2\rangle 1$, $QuorumAssumption$, $Z3$
DEFS $CannotVoteAt$, $DidNotVoteAt$, $OneVote$,
$ChosenAt$, $NoneOtherChoosableAt$, $Ballot$, $SafeAt$
$\langle 1\rangle 4$. QED
BY $QuorumAssumption$, $\langle 1\rangle 1$, $\langle 1\rangle 2$, $\langle 1\rangle 3$, $Z3$
DEFS $Ballot$, $ChosenAt$, $OneVote$, $chosen$

THEOREM $ShowsSafety \triangleq$
$TypeOK \land VotesSafe \land OneValuePerBallot \Rightarrow$
$\forall\, Q \in Quorum,\, b \in Ballot,\, v \in Value :$
$ShowsSafeAt(Q,\, b,\, v) \Rightarrow SafeAt(b,\, v)$
BY $QuorumAssumption$, $Z3$
DEFS $Ballot$, $TypeOK$, $VotesSafe$, $OneValuePerBallot$, $SafeAt$,
$ShowsSafeAt$, $CannotVoteAt$, $NoneOtherChoosableAt$, $DidNotVoteAt$

THEOREM $SafeAtStable \triangleq Inv \land Next \land TypeOK' \Rightarrow$
$\forall\, b \in Ballot,\, v \in Value :$
$SafeAt(b,\, v) \Rightarrow SafeAt(b,\, v)'$
OMITTED

THEOREM $Invariant \triangleq Spec \Rightarrow \Box Inv$
$\langle 1\rangle$ USE DEF Inv
$\langle 1\rangle 1$. $Init \Rightarrow Inv$
BY DEF $Init$, $TypeOK$, $VotesSafe$, $OneValuePerBallot$, $VotedFor$
$\langle 1\rangle 2$. $Inv \land [Next]_{\langle votes,\, maxBal\rangle} \Rightarrow Inv'$

⟨2⟩ SUFFICES ASSUME Inv, $[Next]_{\langle votes,\, maxBal\rangle}$
PROVE Inv'

OBVIOUS

⟨2⟩1.CASE $Next$

⟨3⟩ SUFFICES ASSUME NEW $a \in Acceptor$, NEW $b \in Ballot$,
$\lor\ IncreaseMaxBal(a,\, b)$
$\lor\ \exists\, v \in Value : VoteFor(a,\, b,\, v)$
PROVE Inv'

BY ⟨2⟩1 DEF $Next$

⟨3⟩1.CASE $IncreaseMaxBal(a,\, b)$

⟨4⟩1. $TypeOK'$
BY ⟨3⟩1 DEF $TypeOK$, $IncreaseMaxBal$

⟨4⟩2. $VotesSafe'$

⟨5⟩ SUFFICES ASSUME NEW $a\_1 \in Acceptor'$, NEW $b\_1 \in Ballot'$, NEW $v \in Value'$
PROVE $VotedFor(a\_1,\, b\_1,\, v)' \Rightarrow SafeAt(b\_1,\, v)'$

BY DEF $VotesSafe$

⟨5⟩1. $\forall\, aa \in Acceptor,\, bb \in Ballot,\, vv \in Value :$
$VotedFor(aa,\, bb,\, vv) \equiv VotedFor(aa,\, bb,\, vv)'$
BY ⟨3⟩1 DEF $IncreaseMaxBal$, $VotedFor$

⟨5⟩2. $\forall\, aa \in Acceptor,\, bb \in Ballot :$
$maxBal[aa] > bb \Rightarrow maxBal'[aa] > bb$
BY ⟨3⟩1 DEF $IncreaseMaxBal$, $TypeOK$, $Ballot$

⟨5⟩3. $\forall\, aa \in Acceptor,\, bb \in Ballot :$
$DidNotVoteAt(aa,\, bb) \Rightarrow DidNotVoteAt(aa,\, bb)'$
BY ⟨3⟩1 DEF $IncreaseMaxBal$, $DidNotVoteAt$, $VotedFor$

⟨5⟩4. $\forall\, aa \in Acceptor,\, bb \in Ballot :$
$CannotVoteAt(aa,\, bb) \Rightarrow CannotVoteAt(aa,\, bb)'$
BY ⟨3⟩1, ⟨5⟩2, ⟨5⟩3 DEF $IncreaseMaxBal$, $CannotVoteAt$

⟨5⟩5. $\forall\, bb \in Ballot,\, vv \in Value :$
$NoneOtherChoosableAt(bb,\, vv) \Rightarrow NoneOtherChoosableAt(bb,\, vv)'$
BY ⟨5⟩1, ⟨5⟩4, $QuorumAssumption$DEFS $NoneOtherChoosableAt$

⟨5⟩6. QED
BY ⟨5⟩1, ⟨5⟩5 DEF $TypeOK$, $Ballot$, $VotesSafe$, $SafeAt$

⟨4⟩3. $OneValuePerBallot'$
BY ⟨3⟩1 DEF $IncreaseMaxBal$, $OneValuePerBallot$, $VotedFor$

⟨4⟩4. QED
BY ⟨4⟩1, ⟨4⟩2, ⟨4⟩3 DEF Inv

⟨3⟩2. ASSUME NEW $v \in Value$,
$VoteFor(a,\, b,\, v)$
PROVE Inv'

⟨4⟩ SUFFICES ASSUME NEW $Q \in Quorum$,
$ShowsSafeAt(Q,\, b,\, v)$
PROVE Inv'

BY ⟨3⟩2 DEF $VoteFor$

⟨4⟩1. $TypeOK'$

BY $\langle 3\rangle 2$ DEF *TypeOK*, *VoteFor*

$\langle 4\rangle 2$. *VotesSafe*′ Using *OneValuePerBallot* in *SafeAtStable*

$\langle 5\rangle$ SUFFICES ASSUME NEW $aa \in Acceptor'$, NEW $bb \in Ballot'$, NEW $vv \in Value'$, $VotedFor(aa, bb, vv)'$
PROVE $SafeAt(bb, vv)'$

BY DEF *VotesSafe*

$\langle 5\rangle 1$.CASE $VotedFor(aa, bb, vv)$

$\langle 6\rangle 1$. $SafeAt(bb, vv)$

BY $\langle 5\rangle 1$ DEF *VotesSafe*

$\langle 6\rangle$ QED

BY $\langle 4\rangle 1$, $\langle 6\rangle 1$, *SafeAtStable* DEF *Next*

$\langle 5\rangle 2$.CASE $\neg VotedFor(aa, bb, vv)$

$\langle 6\rangle 1$. $aa = a \wedge bb = b \wedge vv = v \wedge VotedFor(a, b, v)'$

BY $\langle 3\rangle 2$, $\langle 4\rangle 1$, $\langle 5\rangle 2$ DEF *VoteFor*, *VotedFor*, *TypeOK*

$\langle 6\rangle$ QED

BY $\langle 4\rangle 1$, $\langle 6\rangle 1$, *ShowsSafety*, *SafeAtStable* DEF *VoteFor*, *Next*

$\langle 5\rangle$ QED

BY $\langle 5\rangle 1$, $\langle 5\rangle 2$

$\langle 4\rangle 3$. *OneValuePerBallot*′

BY $\langle 3\rangle 2$ DEF *VoteFor*, *OneValuePerBallot*, *VotedFor*, *TypeOK*

$\langle 4\rangle 4$. QED

BY $\langle 3\rangle 2$, $\langle 4\rangle 1$, $\langle 4\rangle 2$, $\langle 4\rangle 3$ DEF *Inv*

$\langle 3\rangle 3$. QED

BY $\langle 2\rangle 1$, $\langle 3\rangle 1$, $\langle 3\rangle 2$

$\langle 2\rangle 2$.CASE UNCHANGED $\langle votes, maxBal\rangle$

BY $\langle 2\rangle 2$

DEFS *TypeOK*, *Next*, *VotesSafe*, *OneValuePerBallot*, *VotedFor*, *SafeAt*, *NoneOtherChoosableAt*, *CannotVoteAt*, *DidNotVoteAt*, *IncreaseMaxBal*, *VoteFor*

$\langle 2\rangle 3$. QED

BY $\langle 2\rangle 1$, $\langle 2\rangle 2$

$\langle 1\rangle 3$. QED

BY $\langle 1\rangle 1$, $\langle 1\rangle 2$, *PTL* DEF *Spec*

THEOREM *Consistent* $\triangleq$ $Spec \Rightarrow \Box Consistency$

$\langle 1\rangle$ USE DEF *Ballot*

$\langle 1\rangle 1$. $Inv \Rightarrow Consistency$

$\langle 2\rangle$ SUFFICES ASSUME *Inv*
PROVE *Consistency*

OBVIOUS

$\langle 2\rangle$ QED

BY *VotesSafeImpliesConsistency*, *OneVoteThm* DEF *Inv*, *Consistency*

$\langle 1\rangle 2$. QED

BY *Invariant*, $\langle 1\rangle 1$, *PTL*

$C \triangleq$ INSTANCE *Consensus* WITH $chosen \leftarrow chosen$

THEOREM $Refinement \triangleq Spec \Rightarrow C!Spec$
⟨1⟩1. $Init \Rightarrow C!Init$
BY *QuorumAssumption*, *SetExtensionality*, *IsaM*("force")
DEF *Init*, $C!Init$, *chosen*, *ChosenAt*, *VotedFor*
⟨1⟩2. $TypeOK' \land Consistency' \land [Next]_{\langle votes,\, maxBal\rangle} \Rightarrow [C!Next]_{chosen}$
⟨2⟩1. UNCHANGED $\langle votes, maxBal\rangle \Rightarrow$ UNCHANGED *chosen*
BY DEF *chosen*, *ChosenAt*, *VotedFor*
⟨2⟩2. $TypeOK' \land Consistency' \land Next \Rightarrow C!Next \lor$ UNCHANGED *chosen*
⟨3⟩1. SUFFICES ASSUME $TypeOK'$, $Consistency'$, *Next*
PROVE $C!Next \lor$ UNCHANGED *chosen*
OBVIOUS
⟨3⟩2. $chosen \subseteq chosen'$
BY ⟨3⟩1, *QuorumAssumption*, *Z3*
DEFS *Next*, *IncreaseMaxBal*, *VoteFor*, *Inv*, *TypeOK*, *chosen*, *ChosenAt*, *VotedFor*, *Ballot*
⟨3⟩3. $chosen' = \{\} \lor \exists\, v \in Value : chosen' = \{v\}$
BY ⟨3⟩1 DEF *Consistency*
⟨3⟩4. QED
BY ⟨3⟩1, ⟨3⟩2, ⟨3⟩3 DEF $C!Next$
⟨2⟩3. QED
BY ⟨2⟩1, ⟨2⟩2
⟨1⟩3. QED
BY ⟨1⟩1, ⟨1⟩2, *Invariant*, *Consistent*, *PTL* DEF *Spec*, *Inv*, $C!Spec$