MODULE *DomainModel*

Common definitions for DAG-based consensus protocols

EXTENDS *FiniteSets*, *Integers*

CONSTANTS

N The set of nodes

, F Byzantine nodes

, R set of rounds

, $Quorum$ The set of quorums (*e.g.* cardinality $\geq 2f+1$)

, $Blocking$ The set of blocking set (*e.g.* cardinality $\geq f+1$)

, $Leader(\_)$ operator mapping each round to its leader

ASSUME $\exists\, n \in R : R = 0 \,..\, n$

DAG notions

DAG vertices are just pairs consisting of a node and a round:

$V \triangleq N \times R$

$Node(v) \triangleq v[1]$

$Round(v) \triangleq v[2]$

A *digraph* is just a set of edges:

$IsDigraph(digraph) \triangleq \forall\, e \in digraph :$

$\quad \land\ e = \langle e[1],\, e[2] \rangle$

$\quad \land\ \{e[1],\, e[2]\} \subseteq V$

$Vertices(digraph) \triangleq \text{UNION } \{\{e[1],\, e[2]\} : e \in digraph\}$

$Children(v,\, digraph) \triangleq$

$\quad \{c \in V : \langle v,\, c \rangle \in digraph\}$

RECURSIVE $Reachable(\_, \_, \_)$

$Reachable(v1,\, v2,\, dag) \triangleq$

$\quad \lor\ v1 = v2$

$\quad \lor\ \exists\, c \in Children(v1,\, dag) : Reachable(c,\, v2,\, dag)$

$Parents(v,\, digraph) \triangleq$

$\quad \{e[1] : e \in \{e \in digraph : e[2] = v\}\}$

$SubDAG(vs,\, es) \triangleq$ vertices *vs* form a sub-DAG (no missing children) of *DAG es*

$\quad \forall\, v \in vs : Children(v,\, es) \subseteq vs$

Other stuff

An arbitrary ordering of the nodes:

$NodeSeq \triangleq \text{CHOOSE } s \in [1 \,..\, Cardinality(N) \rightarrow N] :$

$\quad \forall\, i, j \in 1 \,..\, Cardinality(N) : i \neq j \Rightarrow s[i] \neq s[j]$

$NodeIndex(n) \triangleq \text{CHOOSE } i \in 1 \mathrel{..} Cardinality(N) : NodeSeq[i] = n$

An arbitrary ordering of the nodes with the leader last:

$$\begin{aligned}
&NodeSeqLeaderLast(r) \triangleq \text{CHOOSE } s \in [1 \mathrel{..} Cardinality(N) \rightarrow N] :\\
&\qquad \land\ s[Cardinality(N)] = Leader(r)\\
&\qquad \land\ \forall\, i, j \in 1 \mathrel{..} Cardinality(N) : i \neq j \Rightarrow s[i] \neq s[j]\\
&NodeIndexLeaderLast(n, r) \triangleq \text{CHOOSE } i \in 1 \mathrel{..} Cardinality(N) : NodeSeqLeaderLast(r)[i] = n
\end{aligned}$$