MODULE *ReachabilityProofs*

This module contains several lemmas about the operator *ReachableFrom* defined in module *Reachability*. Their proofs have been checked with the *TLAPS* proof system. The proofs contain comments explaining how such proofs are written.

Lemmas *Reachable*1, *Reachable*2, and *Reachable*3 are used to prove correctness of the algorithm in module Reachable. Lemma *Reachable*0 is used in the proof of lemmas *Reachable*1 and *Reachable*3. You might want to read the proofs in module Reachable before reading any further.

All the lemmas except *Reachable*1 are obvious consequences of the definition of *ReachableFrom*.

EXTENDS *Reachability*, *NaturalsInduction*, *TLAPS*

This lemma is quite trivial. It's a good warmup exercise in using *TLAPS* to reason about data structures.

LEMMA $Reachable0 \triangleq$
$\forall S \in \text{SUBSET } Nodes :$
$\forall n \in S : n \in ReachableFrom(S)$

Applying the Decompose Proof command to the lemma generates the following statement.

$\langle 1 \rangle$ SUFFICES ASSUME NEW $S \in \text{SUBSET } Nodes$,
NEW $n \in S$
PROVE $n \in ReachableFrom(S)$

OBVIOUS

By definition of Reachable, we have to show that there exists a path from some node m in S to n. We obviously want to use n for m.

$\langle 1 \rangle 1.$ $ExistsPath(n, n)$

To convince *TLAPS* that there exists a path from n to n, we have to give it the path. That path is obviously $\langle n \rangle$. A convenient way to tell *TLAPS* to use that path is with the statement:

$\langle 2 \rangle$ WITNESS $\langle n \rangle \in Seq(Nodes)$

We can use this statement because the current goal is $ExistsPath(n, n)$ which by definition of *ExistsPath* and *IsPathFromTo* equals $\exists p \in Seq(Nodes) : F(p)$, with the obvious meaning of $F(p)$. The body of this WITNESS statement is an abbreviation for:

$\langle 2 \rangle$ SUFFICES $F(\langle n \rangle)$
$\langle 3 \rangle 1. \langle n \rangle \in Seq(Nodes)$
OBVIOUS
$\langle 3 \rangle 2.$ QED
BY $\langle 3 \rangle 1$

The WITNESS statement takes no proof. Since correctness of the equivalent SUFFICES step depends on the definitions of *ExistsPath* and *IsPathFromTo*, we need to tell *TLAPS* to use those definitions by putting the following USE statement before the WITNESS step.

$\langle 2 \rangle$ USE DEF *ExistsPath*, *IsPathFromTo*
$\langle 2 \rangle$ WITNESS $\langle n \rangle \in Seq(Nodes)$
$\langle 2 \rangle$ QED
OBVIOUS

$\langle 1 \rangle 2.$ QED
PROOF BY $\langle 1 \rangle 1$ DEF *ReachableFrom*, *ExistsPath*

The following lemma lies at the heart of the correctness of the algorithm in module Reachable. The lemma is not obviously true. To write a proof that *TLAPS* can check, we need to start with an informal proof and then formalize that proof in TLA+. A mathematician should be able to devise an informal proof of this lemma in her head. Others will have to write it down. The informal proof that I came up with appears as comments placed at the appropriate points in the TLA+ proof. I devised the informal proof before I started writing the TLA+ proof. But it's easier to read that informal proof by using the higher levels of the TLA+ proof to give it the proper hierarchical structure. The best way to read the proof hierarchically is in the *Toolbox*, clicking on the little + and - icons beside a step to show and hide the step's proof. Start by executing the Hide Current Subtree command on the lemma.

LEMMA $Reachable1 \triangleq$
$$\forall\, S,\ T \in \text{SUBSET } Nodes :$$
$$(\forall\, n \in S : Succ[n] \subseteq (S \cup T))$$
$$\Rightarrow (S \cup ReachableFrom(T)) = ReachableFrom(S \cup T)$$

An informal proof usually begins by implicitly assuming the following step.

$\langle 1 \rangle$ SUFFICES ASSUME NEW $S \in$ SUBSET $Nodes$, NEW $T \in$ SUBSET $Nodes$,
$$\forall\, n \in S : Succ[n] \subseteq (S \cup T)$$
PROVE $(S \cup ReachableFrom(T)) = ReachableFrom(S \cup T)$

OBVIOUS

The goal is that two sets are equal. The most common way to prove this is to prove that each set is a subset of the other.

$\langle 1 \rangle 1$. $(S \cup ReachableFrom(T)) \subseteq ReachableFrom(S \cup T)$

This is pretty obvious from the definitions. I realized that it follows immediately from two easily proved facts:

- $ReachableFrom(S \cup T) = ReachableFrom(S) \cup ReachableFrom(T)$
- $S \subseteq ReachableFrom(S)$

However, I tried to see if *TLAPS* could prove it more directly. It couldn't prove it directly from the definitions, but it could when I told it to first prove step $\langle 2 \rangle 1$. I then noticed that the same step occurred in the proof of lemma *Reachable*3, which I had already proved. (It's a good idea to prove the simplest theorems first.) So, I pulled that step and its proof out into lemma *Reachable*0.

$\langle 2 \rangle 1$. $\forall\, n \in S : n \in ReachableFrom(S)$
BY *Reachable*0

$\langle 2 \rangle 2$. QED
BY $\langle 2 \rangle 1$ DEF *ReachableFrom*

$\langle 1 \rangle 2$. $ReachableFrom(S \cup T) \subseteq (S \cup ReachableFrom(T))$

To prove that a set U is a subset of a set V, we prove that every element of U is an element of V. This is proved by letting n be any element of U and proving that it's an element of V. This leads to the following reduction of what has to be proved.

$\langle 2 \rangle$ SUFFICES ASSUME NEW $n \in ReachableFrom(S)$
PROVE $n \in S \cup ReachableFrom(T)$

BY DEF *ReachableFrom*

The assumption that n is in $ReachableFrom(S)$ tells us that there exists an element m in S and a path p from m to n. We need to prove that the existence of such an m and p implies that n is in S or in $ReachableFrom(T)$, using the assumption that $succ[m]$ is a subset of $S \cup T$ (which follows from the lemma's hypothesis).

A lot of thought convinced me that the only way of proving this is by induction. In general, there are many ways to reason by induction. For example, if S is a finite set, we can prove our goal by induction on S. However, there's no need to assume that S or T are finite. So, the obvious approach was then induction on the length of the path p. We can do that by defining

$R(i) \triangleq$ For any m in S and q in *Nodes*, if there is a path of length i from m to q then q is in $S \cup \mathit{ReachableFrom}(T)$

and then proving that $R(i)$ holds for all positive integers by proving $R(1)$ and $R(i) \Rightarrow R(i+1)$. However, the *NaturalInductions* module contains an induction rule for proving a result about all natural numbers by proving it first for 0. So we define $R(i)$ as follows so that $R(0)$ is the assertion for paths of length 1.

$\langle 2 \rangle$ DEFINE $R(i) \triangleq$

$$\begin{array}{l} \forall\, m \in S,\, q \in \mathit{Nodes} : \\ \quad (\exists\, p \in \mathit{Seq}(\mathit{Nodes}) : \land\, \mathit{IsPathFromTo}(p,\, m,\, q) \\ \qquad\qquad\qquad\qquad\quad \land\, \mathit{Len}(p) = i + 1) \\ \qquad \Rightarrow (q \in S \cup \mathit{ReachableFrom}(T)) \end{array}$$

$\langle 2 \rangle 1.\ \forall\, i \in \mathit{Nat} : R(i)$

Level $\langle 3 \rangle$ is the obvious decomposition for an induction proof.

$\langle 3 \rangle 1.\ R(0)$

TLAPS has no problem proving this.

$\langle 4 \rangle$ SUFFICES ASSUME NEW $m \in S$, NEW $q \in \mathit{Nodes}$,
NEW $p \in \mathit{Seq}(\mathit{Nodes})$,
$\land\, \mathit{IsPathFromTo}(p,\, m,\, q)$
$\land\, \mathit{Len}(p) = 0 + 1$
PROVE $q \in S \cup \mathit{ReachableFrom}(T)$

OBVIOUS

$\langle 4 \rangle$ QED

BY DEF *IsPathFromTo*

$\langle 3 \rangle 2.$ ASSUME NEW $i \in \mathit{Nat}$, $R(i)$
PROVE $R(i+1)$

The proof of $R(i+1)$ is decomposed as usual.

$\langle 4 \rangle$ SUFFICES ASSUME NEW $m \in S$, NEW $q \in \mathit{Nodes}$,
NEW $p \in \mathit{Seq}(\mathit{Nodes})$,
$\land\, \mathit{IsPathFromTo}(p,\, m,\, q)$
$\land\, \mathit{Len}(p) = (i + 1) + 1$
PROVE $q \in S \cup \mathit{ReachableFrom}(T)$

BY DEF R

Since m is in S and $p[2]$ is in $\mathit{Succ}[m]$, the lemma's hypothesis implies that $p[2]$ is in $S \cup T$. The proof that q is in $S \cup \mathit{ReachableFrom}(T)$ is split into the two cases $p[2] \in S$ and $p[2] \in T$. If $p[2]$ is in S, then the result follows from the induction hypothesis, since $\mathit{Tail}(p)$ is a path of length $\mathit{Len}(p) - 1$ from an element of S to q . If $p[2]$ is in T, then $\mathit{Tail}(p)$ is a path from an element of T to q, so q is in $\mathit{ReachableFrom}(T)$.

Step $\langle 4 \rangle 1$ asserts some simple facts that I found were needed to get *TLAPS* to prove the first case. I then found they helped *TLAPS* prove the second case too, so I moved them before the case split.

⟨4⟩1. ∧ $Tail(p) \in Seq(Nodes)$
∧ $IsPathFromTo(Tail(p),\ p[2],\ q)$
∧ $Len(Tail(p)) = i + 1$
BY DEF $IsPathFromTo$

This step isn't necessary because *TLAPS* can figure out that the two cases are exhaustive from the usable facts and the definition of *PathFromTo*, but I think it makes the proof easier to read.

⟨4⟩2. $p[2] \in S \cup T$
BY DEF $IsPathFromTo$

TLAPS easily proves the two cases. However, it needs to be told to split the proof into cases because it's not good at figuring out by itself when to use a case split.

⟨4⟩3.CASE $p[2] \in S$
BY ⟨3⟩2, ⟨4⟩1, ⟨4⟩3

⟨4⟩4.CASE $p[2] \in T$
BY ⟨4⟩1, ⟨4⟩4 DEF $ReachableFrom$, $ExistsPath$

⟨4⟩5. QED
BY ⟨4⟩2, ⟨4⟩3, ⟨4⟩4

⟨3⟩ HIDE DEF R

⟨3⟩3. QED
BY ⟨3⟩1, ⟨3⟩2, $NatInduction$, Isa

Proving $q \in S \cup ReachableFrom(T)$ from ⟨2⟩1 is straightforward.

⟨2⟩2. PICK $m \in S,\ p \in Seq(Nodes) :$
$IsPathFromTo(p,\ m,\ n)$
BY DEF $ReachableFrom$, $ExistsPath$

We have to tell *TLAPS* to apply ⟨2⟩1 with $i = Len(p) - 1$.

⟨2⟩3. $R(Len(p) - 1) \Rightarrow n \in S \cup ReachableFrom(T)$
BY ⟨2⟩2 DEF $IsPathFromTo$

Hiding the definition of R makes it easier for *TLAPS* to prove the result.

⟨2⟩ HIDE DEF R

The definition of *IsPathFromTo* is needed for *TLAPS* to deduce $Len(p) > 0$, so $Len(p) - 1$ is in Nat.

⟨2⟩4. QED
BY ⟨2⟩1, ⟨2⟩2, ⟨2⟩3 DEF $IsPathFromTo$

⟨1⟩3. QED
BY ⟨1⟩1, ⟨1⟩2

The proof of this lemma is straightforward.

LEMMA $Reachable2 \triangleq$
$\forall\, S \in$ SUBSET $Nodes : \forall\, n \in S :$
$\wedge\ ReachableFrom(S) = ReachableFrom(S \cup Succ[n])$
$\wedge\ n \in ReachableFrom(S)$

⟨1⟩ SUFFICES ASSUME NEW $S \in$ SUBSET $Nodes$,
NEW $n \in S$
PROVE $\wedge\ ReachableFrom(S) = ReachableFrom(S \cup Succ[n])$

$\wedge\ n \in \mathit{ReachableFrom}(S)$

OBVIOUS

$\langle 1\rangle 1.\ \mathit{ReachableFrom}(S) = \mathit{ReachableFrom}(S \cup \mathit{Succ}[n])$

We decompose the proof of equality of two sets to proving the two subset relations.

$\langle 2\rangle 1.\ \mathit{ReachableFrom}(S) \subseteq \mathit{ReachableFrom}(S \cup \mathit{Succ}[n])$

This subset relation is trivial because $S \subseteq T$ obviously implies $\mathit{ReachableFrom}(S) \subseteq \mathit{Reachable}(T)$

BY DEF $\mathit{ReachableFrom}$

$\langle 2\rangle 2.\ \mathit{ReachableFrom}(S \cup \mathit{Succ}[n]) \subseteq \mathit{ReachableFrom}(S)$

We reduce the proof $U \subseteq V$ to proving that $u \in V$ for every u in U.

$\langle 3\rangle$ SUFFICES $\mathit{ReachableFrom}(\mathit{Succ}[n]) \subseteq \mathit{ReachableFrom}(S)$
BY DEF $\mathit{ReachableFrom}$

$\langle 3\rangle$ SUFFICES ASSUME NEW $m \in \mathit{Succ}[n]$, NEW $o \in \mathit{Nodes}$, $\mathit{ExistsPath}(m,\ o)$
PROVE $\mathit{ExistsPath}(n,\ o)$
BY DEF $\mathit{ReachableFrom}$

$\langle 3\rangle 1.$ PICK $p \in \mathit{Seq}(\mathit{Nodes}) : \mathit{IsPathFromTo}(p,\ m,\ o)$
BY DEF $\mathit{ExistsPath}$

$\langle 3\rangle$ DEFINE $q \triangleq \langle n\rangle \circ p$

$\langle 3\rangle 2.\ (q \in \mathit{Seq}(\mathit{Nodes})) \wedge \mathit{IsPathFromTo}(q,\ n,\ o)$
BY $\langle 3\rangle 1$, $\mathit{SuccAssump}$ DEF $\mathit{IsPathFromTo}$

$\langle 3\rangle 3.$ QED
BY $\langle 3\rangle 2$ DEF $\mathit{ExistsPath}$

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

Here's where we need $\mathit{Reachable}0$.

$\langle 1\rangle 2.\ n \in \mathit{ReachableFrom}(S)$
BY $\mathit{Reachable}0$

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

This lemma is quite obvious.

LEMMA $\mathit{Reachable}3 \triangleq \mathit{ReachableFrom}(\{\}) = \{\}$
BY DEF $\mathit{ExistsPath}$, $\mathit{ReachableFrom}$

* Modification History
* Last modified Sat *Apr* 13 18:07:57 *PDT* 2019 by *lamport*
* Created *Thu Apr* 11 18:19:10 *PDT* 2019 by *lamport*