Outline

Cookies
vampire --mode casc SET014-3.p
If-then-else and Let-in

A partial correctness statement:

\[
\{ \forall X (p(X) \Rightarrow X \geq 0) \} \\
\{ \forall X (q(X) > 0) \} \\
\{ p(a) \} \\
\text{if } (r(a)) \{ \\
    a := a + 1 \\
\} \\
\text{else } \{ \\
    a := a + q(a). \\
\} \\
\{ a > 0 \}
\]
If-then-else and Let-in

A partial correctness statement:

\[
\{ \forall X (p(X) \implies X \geq 0) \} \\
\{ \forall X (q(X) > 0) \} \\
\{p(a)\} \\
\text{if } (r(a)) \{ \\
\quad a := a+1 \\
\} \\
\text{else } \{ \\
\quad a := a + q(a). \\
\} \\
\{a > 0\}
\]

The next state function for \(a\):

\[
a' = \\
\quad \text{if } r(a) \\
\quad \text{then let } a=a+1 \text{ in } a \\
\quad \text{else let } a=a+q(a) \text{ in } a
\]
If-then-else and Let-in

A partial correctness statement:

{∀X(p(X) => X ≥ 0)}
{∀X(q(X) > 0)}
{p(a)}
if (r(a)) {
  a := a+1
}
else {
  a := a + q(a).
}
{a > 0}

The next state function for a:

a' =
  if r(a)
  then let a=a+1 in a
  else let a=a+q(a) in a

In Vampire:

tff(1,type,p : $int > $o).
tff(2,type,q : $int > $int).
tff(3,type,r : $int > $o).
tff(4,type,a : $int).

tff(5,hypothesis,! [X:$int] : (p(X) => $greatereq(X,0))).
tff(6,hypothesis,! [X:$int] : ($greatereq(q(X),0))).
tff(7,hypothesis,p(a)).

tff(8,hypothesis,

  a0 = $itet(r(a),
      $lettt(a,$sum(a,1),a),
      $lettt(a,$sum(a,q(a)),a))
  )).

tff(9,conjecture,$greater(a0,0)).
Consequence Elimination

Given a large set of formulas, find out which formulas are consequences of other formulas in the set. For example, used for pruning a set of automatically found loop invariants.
Consequence Elimination

Given a large set of formulas, find out which formulas are consequences of other formulas in the set. For example, used for pruning a set of automatically found loop invariants.

fof(ax1, axiom,a => b).
fof(ax2, axiom,b => c).
fof(ax3, axiom,c => a).

fof(c1, claim, a | d).
fof(c2, claim, b | d).
fof(c3, claim, c | d).

vampire --mode consequence_elimination consequence.tptp
Grounding

- Can transform first-order problems into propositional ones (complete for EPR);
- SAT solver can be used after grounding.

vampire --mode grounding
Can be (and was being) used as a clausifier.

vampire --mode clausify simple.tptp

All Vampire preprocessing options can be used.
Very Large Theories

Implement Sine selection for selecting axioms “relevant” to the conjecture.
vampire --mode program_analysis

(Yet unavailable).