Model Checking WS 2015: Assignment 2

Institute for Formal Models and Verification, JKU Linz

Due 05.11.2015

Exercise 7

Let x and y be *4-bit signed integer* variables with Java semantics, i.e. two's complement representation and modular arithmetic with underflow/overflow. Let $a \leftrightarrow$ `((x % 2) == 0)` and $b \leftrightarrow$ `(x < y)` be predicates. a and b together define 4 possible abstract states. Draw an abstract transition system for predicates a and b and actions $\alpha :=$ `x++` and $\beta :=$ `y = y - 2`. How do α and β influence the values of a and b? Make sure to consider all possible combinations of abstract states and transitions. For each transition you draw between abstract states, give *concrete values* for variables x and y as a "witness".

Exercise 8

Using predicates a and b and actions α and β from the abstract transition system in Exercise 7, construct an abstract program statement by statement for the program fragment given below. Replace relational expressions by the corresponding predicate. Make sure that *all possible* transitions between abstract states (see Exercise 9) by executing $\alpha :=$ `x++` and $\beta :=$ `y = y - 2` are *fully* encoded in the abstract program. Your program should have only two boolean variables a and b.

```
assert (x < y);
lock();
while (x < y) {
  if ((x % 2) == 0) x++; else y = y - 2;
  if (x >= y && (x % 2) != 0)
    unlock;
}
```

Exercise 9

Justify your answers to the following questions.

a) Explain why the empty set $\lesssim := \{\}$ is a simulation over any arbitrary LTS $L = (S, I, \Sigma, T)$.

b) Given an LTS L and two simulations $\lesssim_1$ and $\lesssim_2$ over L. Prove that $\lesssim_1 \cup \lesssim_2$ is a simulation over L as well.

c) Draw all different LTS $L = (S, I, \Sigma, T)$ with the restrictions that $I = S = \{1, 2\}$, $\Sigma = \{a\}$ and further $(1, a, 2) \notin T$ and $(2, a, 2) \notin T$.

d) Given the relation $\lesssim := S \times S$. For which LTS of part c) is $\lesssim$ a simulation?

Exercise 10

Let F_1 and F_2 be the following formulas over the theory of uninterpreted functions and equality:

$$F_1 := \qquad s = g(v,x) \wedge t = g(x,v) \wedge x = y \wedge v = w \wedge u = z \wedge y = f(z) \wedge w = f(u) \wedge s \neq t$$

$$F_2 := \qquad s = g(y,u) \wedge t = g(z,x) \wedge x = y \wedge v = w \wedge u = z \wedge y = f(z) \wedge w = f(u) \wedge s \neq t$$

Check satisfiability of F_1, F_2 by applying the congruence closure algorithm (SMT slides 12/13).

Exercise 11

Given the following LRA formula:

$$(x \leq 3 \vee \neg(2 \cdot x > 6)) \wedge (2 \cdot x > 6 \vee \neg(x \leq 3))$$

Check its satisfiability by applying propositional abstraction and refinement using Lemmas on demand (SMT slides 17/18). (Use simple reasoning to simulate both a SAT solver as well as a theory solver that can check whether conjunctions of literals are spurious solutions.)

Exercise 12

Given the following input in SMT format. Write it down as a logical formula using infix notation. Is it satisfiable? Justify your answer.

```
(set-logic QF_BV)
(declare-fun x () (_ BitVec 16))
(declare-fun k () (_ BitVec 16))
(assert (= x (bvshl (_ bv1 16) k)))
(assert (distinct (bvand x (bvsub x (_ bv1 16))) (_ bv0 16)))
(check-sat)
(exit)
```