EECS 219C: Computer-Aided Verification **Sanjit A. Seshia**

Homework 3: Symbolic Model Checking, Simulation Checking

Assigned: February 28, 2007 *Due in class: March 14, 2007*

Note: See the webpage for rules on collaboration.

1. **Fixpoint Characterization of AFp** (15 points)
   Prove that the set of states satisfying $\mathbf{AF}p$ is the least fixpoint of the function τ given by $\tau(Z) = p \vee \mathbf{AX}Z$.

2. **Simulation and SAT** (20 points)
   Let $M = (S, S_0, R, L)$ and $M' = (S', S'_0, R', L')$ be two Kripke structures, where the set of atomic propositions is the same for both. In this question we will see how the problem of checking whether M' simulates M can be formulated as a SAT problem. The construction of a SAT problem will use an "explicit-state" approach, enumerating states in S and S'. If the constructed SAT problem is satisfiable, then we conclude that M' simulates M and the satisfying assignment indicates the simulation relation. If not, then M' does not simulate M.

   The following questions will walk you through the construction:

   (a) (3 points)
   For each pair of states (s, s') where $s \in S$ and $s' \in S'$, introduce a Boolean variable $x_{s,s'}$. If $x_{s,s'}$ is assigned 1, it will mean that the pair (s, s') is related by the simulation relation, otherwise not.
   How many such Boolean variables do we have?

   (b) (7 points)
   Suppose that two states s and s' have the same label: $L(s) = L'(s')$. Further, assume that there is $t \in S$ such that $R(s, t)$. Let $\{t'_1, t'_2, \ldots, t'_k\}$ be the k successors of state s' (i.e. $R'(s', t'_j)$ holds for $1 \leq j \leq k$) such that $L(t) = L'(t'_j)$ for each j.
   Express the following constraint as a CNF clause:

   > If (s, s') is related by the simulation relation, then there is some successor t'_j of s', $1 \leq j \leq k$, that is related to t by the simulation relation.

   If $k = 0$ (i.e., there are no successors of s' satisfying the labeling condition), then what is the form of the CNF clause?
   How many such CNF clauses can we possibly have?

   (c) (5 points)
   For each initial state $s_0 \in S_0$, write down a CNF clause expressing the following constraint:

There exists $s'_0 \in S'_0$ such that (s_0, s'_0) are related by the simulation relation.

(Use k to denote the cardinality of S'_0 for this question.)

(d) (5 points)
The overall SAT problem comprises the CNF clauses generated in steps (b) and (c) above. What can you say about the structure of this SAT problem? Is it a special case of SAT? If so, what is the asymptotic running time of an efficient algorithm to solve it (expressed in terms of the sizes of the Kripke structures)?

Note: The above simulation checking algorithm is one of the algorithms underlying a recent software model checker for C programs.

3. **Symmetry Reduction** (15 points)
List all automorphisms for the Kripke structure in Figure 1 and the corresponding orbits.

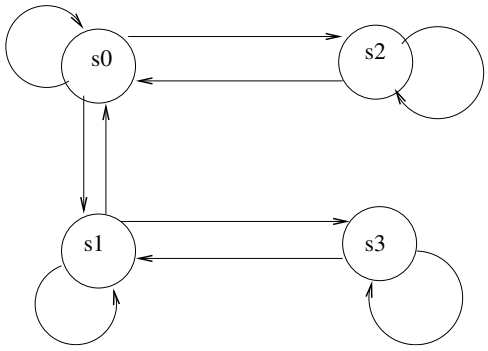

Figure 1: Kripke structure for Symmetry question

4. **A Bridge Traffic Controller in NuSMV** (50 points)[1]
In this question you have to use the NuSMV model checker to design and verify a traffic light controller for a bridge that has only one lane (for both directions).

The controller works as follows:

- At each end of the bridge, there is a traffic light that can be either red or green. This can be modeled with a signal TA, TB $\in \{\text{red}, \text{green}\}$ for the two ends A and B of the bridge.
- At each end, there are also sensors that detect the presence and number of cars that are waiting to cross. Assume that there are at most 10 cars at each end at any time. These can be modeled with counters WA, WB for the two ends of the bridge.

[1] Adapted from a problem formulated by D. Kröning

- At any time, a car travelling from A to B is either waiting at A, or has passed across the bridge. The presence of a car waiting at A is modeled with WA. Passage across the bridge only occurs when the traffic light TA is green; it is instantaneous and is modeled by a decrement to WA. Only one car passes across the bridge in one "clock tick". There is no need to keep track of a car after it leaves the bridge. (Similarly for cars that travel B to A.)
- Once a car starts waiting at one end to use the bridge, it continues to wait until it passes onto the bridge.
- Both lights start out red with no cars at either end.
- If there is at least one car waiting at end A and none at end B, then TA turns green. (Similarly for the symmetric case at end B). If both lights are red and cars arrive simultaneously at both ends, then non-deterministically one of TA and TB is chosen to turn green first.
- If light TA is green, and there are cars waiting at B, the number of cars allowed to pass the bridge from A to B is limited to 5 from that time point; when that bound is reached or no cars remain at A, TA turns red and TB turns green. If there are no cars at end B, then TA continues to stay green until a car arrives at B, from which time point the bound of 5 is imposed on the A to B direction. (Symmetrically for end B.)

The desired properties of the controller are:

(a) TA and TB are never green at the same time.

(b) If a car is waiting to pass the bridge at an end with a red traffic light, that traffic light eventually turns green.

Your task is to do the following:

(a) (20 points) Model the above controller in NuSMV with an environment that models the arrival and passage of cars.

(b) (20 points) Formulate the above desired properties in LTL and verify that your model satisfies them.

(c) (10 points) Is there a number of clock ticks N such that the following re-formulation of the second property is true on your model:

> If a car is waiting to pass the bridge at an end with a red traffic light, that traffic light turns green at most N clock ticks later.

If not, state why not. If so, state the value of N, formulate the property in LTL, and verify the property on your model.

Run NuSMV with BDD-based symbolic model checking. Indicate any non-default options you used and the exact command used for each of your runs.

Include any written answers with the rest of your homework, and also e-mail the file with your NuSMV model to `bbrady @ eecs` by the due date.