15-414 — Bug Catching — Fall 2006

Instructor: Edmund Clarke
Teaching assistant: Himanshu Jain

Assignment 2

Due date: Tuesday, November 14, 2006

1 CTL

This is a problem given in Huth and Ryan's book. Consider the model $\mathcal{M}$ in Figure 1. Check whether $\mathcal{M}, s_0 \models \phi$ and $\mathcal{M}, s_2 \models \phi$ hold for the CTL formula ϕ:

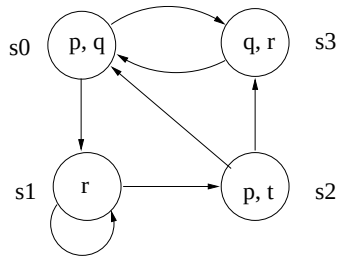

Figure 1: A model with four states

(a) $\mathbf{AF}q$

(b) $\mathbf{AG}(\mathbf{EF}(p \vee r))$

(c) $\mathbf{EX}(\mathbf{EX}r)$

(d) $\mathbf{AG}(\mathbf{AF}q)$

(e) Give the SMV code to check the above properties with the initial state as s_2. Also double check your answers to the previous parts by running a model checker on the SMV code.

2 More CTL

Consider the model $\mathcal{M}$ in Figure 2. Check whether $\mathcal{M}, s_0 \models \phi$ hold for the CTL formula ϕ:

(a) $\mathbf{AG}(\mathbf{AF}c)$

(b) $\mathbf{AG}(\mathbf{AF}(\mathbf{AX}c))$

(c) $\mathbf{AG}(\mathbf{AF}b)$

(d) $\mathbf{AG}(\mathbf{AX}c \vee \mathbf{AX}(\mathbf{AX}(\neg c)))$

(e) $\mathbf{EF}(\mathbf{AG}c)$

You may want to double check your answers by writing the above model in SMV. We do not need to the SMV code.

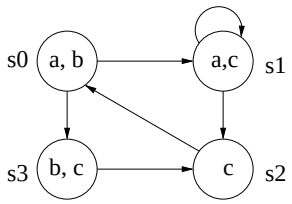

Figure 2: A model with four states

3 Writing specifications in CTL

Express the following properties in CTL. You can use atomic propositions such as $req, ack, restart$ to write the properties.

(a) A request is eventually acknowledged.

(b) A request is acknowledged in atmost three cycles (transitions).

(c) Infinitely many requests occur during the operation of the system.

(d) From any state it is possible to get to a restart state.

(e) From any state it is possible to get to a restart state in exactly two steps.

(f) After p, q is never true.

4 SMV to state machines

In this problem we will use the `INIT, TRANS` construct from SMV language. Consider the SMV program in Figure 3.

```
MODULE main

VAR b1:  boolean;
VAR b2:  boolean;
VAR b3:  boolean;

INIT (b1 & b2 & b3)

TRANS
  (b1 & !b2 & !b3 & next(b1) & !next(b2) & !next(b3)) |
  (b1 & b2 & !b3 & !next(b1) & !next(b2)) |
  (b1 & b2 & b3 & next(b1) & next(b3)) |
  (b1 & !b2 & next(b1) & !next(b2) & next(b3)) |
  (!b1 & !b2 & !next(b1) & !next(b2)) |
  (b1 & b3 & next(b1) & !next(b2) & !next(b3)) |
  (!b1 & b2 & !b3 & !next(b1) & next(b2) & !next(b3)) |
  (!b1 & b2 & b3 & !next(b1) & next(b2) & !next(b3))
```

Figure 3: A SMV program.

(a) Draw the state transition graph represented by the above SMV program? Mark the initial states.

Answer whether the program satisfies the following specifications using the state transition graph.

(b) **AG (b1)**.

(c) **AF (!b2)**.

(d) **AG(EF (b2))**.

(e) **EF(!b1 & b2)**.

Check your answers by using NuSMV or Cadence SMV. The above program is present in a SMV file available seperately.