NATIONAL UNIVERSITY OF SINGAPORE
SCHOOL OF COMPUTING

SPIN/LTL Exercises posted for cs4271 students Semester 2, 2009/2010

(a) Are the two following Linear time Temporal Logic formula equivalent ? If yes, give a proof. If not, construct example traces to show that they are not equivalent.

$$\mathbf{F}(p\mathbf{U}q) \Leftrightarrow \mathbf{F}p \ \mathbf{U} \ \mathbf{F}q$$

You can assume that p and q are atomic propositions.

Answer: The two formulae are equivalent. Consider a trace π satisfying $F(pUq)$. Then by the definition of the F, U operators, there must exist a state in π which satisfies q. Let the first position in π where q is true be k. Then clearly $\pi^k \models pUq$, and hence $\pi \models F(pUq)$. Since $\pi^k \models q$ We see that $\pi \models Fq$; By definition of the until operator $\pi \models \varphi \Rightarrow \pi \models \psi U \varphi$ for any LTL properties φ, ψ. Thus, $\pi \models FpUFq$.

Now, consider any trace π such that $\pi \models FpUFq$. Again it means that there exists $k \geq 0$ such that $\pi^k \models Fq$ which means that there exists $m \geq k \geq 0$ such that $\pi^m \models q$. Then $\pi^m \models pUq$ and hence $\pi \models F(pUq)$.

This concludes the proof of equivalence of the two formulae. In fact we see that any trace with at least one state in which q is true, satisfies both the formulae and vice-versa.

(b) In class, we discussed the nested depth-first search algorithm implemented inside the model checker SPIN. Among other things, this allows us to easily retrieve the counter-example trace from the stack. Suppose we implemented breadth-first search with queues instead for the purpose of model checking. Will the task of counter-example computation become any more difficult? Explain your answer.

Answer: In the nested depth-first search, the counter-example trace can be obtained can simply concatenating the two stacks. This will not be the case for the nested breadth-first search. In order to retrieve the counter-example trace in the nested breadth-first search we need to perform more book-keeping during the search. One possibility is to store a link at each state pointing to a predecessor state; this will allow the counter-example trace to be reconstructed when a violation is detected.

(c) Recall the definition of the Until operator $\mathbf{U}$ in Linear-time temporal logic (LTL). Let us now define a new until operator $\mathbf{U_1}$ as follows:
$M, \pi \models \varphi \mathbf{U_1} \psi \;\equiv\;$ if there exists a $k \geq 0$ such that $M, \pi^k \models \psi$ then for all $0 \leq j < k$ we have $M, \pi^j \models \varphi$

The notation π^k was discussed in class (and also appears in the textbook). Express $\varphi \mathbf{U_1} \psi$ as a Linear-time temporal logic (LTL) formula and give explanation for your answer. You may assume that φ, ψ are arbitrary LTL properties.

Answer: The definition is

$\varphi \mathbf{U}_1 \psi = (\varphi \mathbf{U} \psi) \vee \mathbf{G} \neg \psi$

The only difference between $\mathbf{U}$ and $\mathbf{U}_1$ is that ψ is not required to hold eventually in the definition of $\mathbf{U}_1$. This accounts for the disjunction in the definition of $\mathbf{U}_1$.

(d) Assume p is an atomic proposition. Describe the following property in LTL: "along any path, a state satisfying p occurs at most once". Explain your answer.

Answer:

$G\neg p \vee (\neg p U(p \wedge XG\neg p))$

$G\neg p$ is true when p never occurs.

If p occurs exactly once then the path starting from the state in which p occurs must satisfy $p \wedge XG\neg p$ (i.e. p occurs at the start and never occurs again). This explains the answer.