Homework 1

Ahmed Rezine

April 5, 2022

Submit your solutions in the form of two files: a text document (possibly with pictures for the computation trees and automata) and a SPIN file for problem 4. The SPIN file should be well documented and clear. Make sure you separate the transitions of the frogs (a process for each frog) from the property whose negation captures success.

Problem 1

Assume CTL^*. Express each of the following using (boolean combinations) of $true$, the path formulas `f` and `g`, the temporal binary operator $\mathbf{U}$ and the path quantifer $\mathbf{E}$:

1. $(\mathbf{F}\ \mathtt{f})\ =\ \ldots$
2. $(\mathbf{G}\ \mathtt{f})\ =\ \ldots$
3. $(\mathbf{A}\ \mathtt{f})\ =\ \ldots$
4. $(\mathtt{f}\ \mathbf{R}\ \mathtt{g}) =\ \ldots$

Problem 2

Assume the atomic propositions `blk` (for block), `esc` (for escape), `tai` (for taint), `ted` (for tainted) `run` (for run) and `res` (for reset). Consider the sequence of CTL formulas:

1. $\mathbf{AX}(\mathtt{blk})$
2. $\mathbf{EG}(\mathtt{esc})$
3. $\mathbf{AG}(\mathtt{tai} \implies \mathbf{AF}(\mathtt{ted}))$
4. $\mathbf{AG}(\mathtt{run} \implies \mathbf{EF}(\mathtt{res}))$

For each furmula, give the first levels of a computational tree (similar to those describing formluas $M, s_0 \models \mathbf{EF}\ g$ and $M, s_0 \models \mathbf{AF}\ g$ in the slides) where the corresponding atomic propositions appear at least once. For instance, the computational tree associated to formula (1) should have at least one node satisfying `blk`.

Problem 3

Assume atomic propositions `En` (for Enabled) and `Ex` (for Executed). Consider the following two LTL formulas:

1. $\mathbf{GF}(\neg\mathtt{En} \vee \mathtt{Ex})$

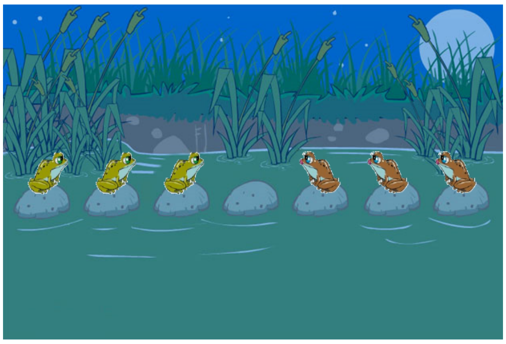
2. $\mathbf{GF}(\texttt{En}) \implies \mathbf{GF}(\texttt{Ex})$

For each formula:

1. Give an infinite word on $\{\texttt{En}, \texttt{Ex}\}$ that violates the formula
2. Give a Büchi automaton that accepts all negations on $\{\texttt{En}, \texttt{Ex}\}$, of the formula.

Problem 4

Model the frogs' puzzle in Spin[1] where an even number "n" of frogs are placed on "n+1" rocks forming a line from left to right. At any given moment, there can be at most one frog on each rock. Half of the frogs sitt on the left rocks and face the rocks to the right. The other half sitts on the right rocks and face the left rocks (see picture below).
The objective is to formalize a model and a property whose counter-example, found by SPIN, is a sequence of moves that allows all frogs to switch side. Frogs jump one step to the rock in front of them if it is empty, or to the rock after that if the rock in front of them is occupied. Your model should ge parameterized by the number "n" of frogs to allow for experimenting with different dimensions of the problem. Send your mordel together with the size of the largest number of frogs that could be handled with a time-out of 5 minutes.



[1] `https://data.bangtech.com/algorithm/switch_frogs_to_the_opposite_side.htm`