Prof. Dr. Andreas Podelski
Dominik Klumpp
Frank Schüssele

Hand in until December 22nd, 2021
23:59 via ILIAS
Discussion: January 10th/11th, 2022

Tutorial for Cyber-Physical Systems - Discrete Models
Exercise Sheet 9

Exercise 1: Liveness Properties 5 Points

The goal of this task is to learn how to recognize liveness properties.

Consider following properties over the set $AP = \{a, b\}$ of atomic propositions.

(P_1) Always (at any point of time) a or b holds.

(P_2) Every time a holds there will be eventually a point of time where b holds.

(P_3) a holds exactly once.

(P_4) b holds only finitely often.

(P_5) False

For each property P_i check if it is a liveness property.

- If so, show that no prefix is *"bad"*: Explain how any finite prefix $A_0A_1 \ldots A_n$ can be continued to form a trace $A_0A_1 \ldots A_nA_{n+1} \ldots$ that satisfies the property.
- Otherwise give any finite prefix that cannot be continued to form a trace that satisfies the property (i.e., a bad prefix).

Example: For the property "eventually a", any finite prefix $A_0A_1 \ldots A_n$ can be extended with $\{a\}^\omega$, such that $A_0A_1 \ldots A_n\{a\}^\omega \models$ "eventually a".

Exercise 2: Safety-Liveness Decomposition 5 Points

The goal of this exercise is to understand the relation between any LT property and safety and liveness properties, by applying the decomposition theorem.

According to the decomposition theorem, any LT property P can be decomposed into a safety property P_{safe} and a liveness property P_{live}, such that the property P is equal to their intersection, i.e.,

$$P = P_{safe} \cap P_{live} \quad .$$

Apply the construction in the proof of the decomposition theorem to find the decomposition for the following properties with $AP = \{a, b\}$. In particular, for each property, give its closure. Use set notation to express P_{safe} and P_{live}.

(P_1) Every a is immediately followed by b.

(P_2) The atomic proposition a holds infinitely often.

(P_3) At exactly 3 points of time, a holds.

(P_4) a holds initially and infinitely often.

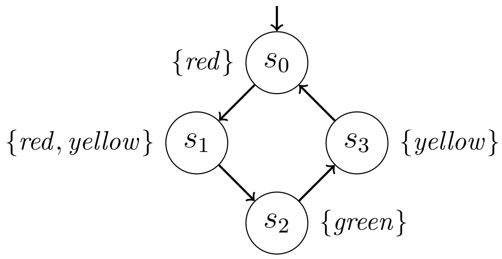
(P_5) True

Hint: Some tasks may require very little work.

Exercise 3: Model Checking 10 Points

In this exercise, we arrive at the goal towards which we have worked since the beginning of the semester: For a cyber-physical system (given as a transition system) and desired correctness properties, we are able to determine if the system satisfies these properties.

The following transition system T models the behaviour of a traffic light.



(a) Draw an NFA $\mathcal{A}_T$ over the alphabet $\Sigma = 2^{AP}$ with $AP = \{red, yellow, green\}$ such that $\mathcal{A}_T$ accepts exactly the finite prefixes of $Traces(T)$, i.e., $\mathcal{L}(\mathcal{A}_T) = pref(Traces(T))$.

Note: You can construct $\mathcal{A}_T$ as you prefer, you do not necessarily need to follow the construction introduced in the lecture (it still has to accept the correct language of course).

(b) Consider the following safety properties:

(P_1) *"It is always the case that if the green light is on, then the red light will be off in the next step."*

(P_2) *"It is always the case that if the red light is on, then the green light will be off in the next step."*

Give automata for the bad prefixes of these properties, i.e., draw NFAs $\mathcal{A}_{P_1}$ and $\mathcal{A}_{P_2}$ that accept exactly the bad prefixes of the property P_1 respectively P_2.

Note: You can (but you don't have to) draw the automata in symbolic notation, i.e., with propositional formulas as edge labels.

(c) Draw the intersection NFAs of $\mathcal{A}_T$ with $\mathcal{A}_{P_1}$ respectively $\mathcal{A}_{P_2}$. For both intersection NFAs, check if the accepted language is empty (i.e. no accepting state can be reached) to determine if T satisfies the respective property.