

## Tutorial for Cyber-Physical Systems - Hybrid Models

### Exercise Sheet 8

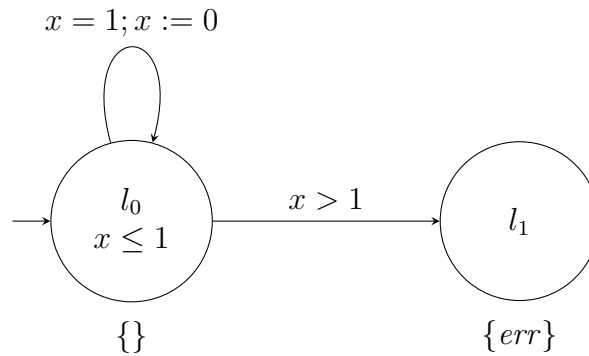


Figure 1: A timed automaton.

#### Exercise 1: Timed automata and programs 1

Consider the timed automaton  $\mathcal{T}_1$  from Figure 1 with clock variable  $x$ , i.e.,  $\mathcal{C} = \{x\}$ .

- (a) Translate  $\mathcal{T}_1$  to an equivalent program  $\mathcal{P}_1$ , i.e., with the same executions/paths.
- (b) Compute the reachable states  $\varphi_{\text{reach}}$  of  $\mathcal{P}_1$  by iteration of *post*.

## Exercise 2: Timed automata and programs 2

Consider the timed automaton  $\mathcal{T}_2$  which is obtained from  $\mathcal{T}_1$  (see Figure 1) by adding another clock variable  $y$ , i.e.,  $\mathcal{C} = \{x, y\}$ . Note that  $y$  is never read in any guard or invariant. Still, the state space changes (recall that a state is a pair  $(\ell, \nu)$  with  $\nu : \mathcal{C} \rightarrow \mathbb{R}$ ).

You may wonder why adding an unused clock should affect the reachability of a state. In fact, it does not (in some sense). However, the algorithms behave differently.

- (a) Translate  $\mathcal{T}_2$  to an equivalent program  $\mathcal{P}_2$ .
- (b) What are the reachable states  $\varphi_{\text{reach}}$  of  $\mathcal{P}_2$ ?  
Solve this exercise intuitively, i.e., do not apply a formal algorithm. Alternatively, you may reason about the reachable states  $Reach$  of  $\mathcal{T}_2$ .
- (c) What happens when you try to compute the reachable states  $\varphi_{\text{reach}}$  of  $\mathcal{P}_2$  by iteration of  $post$ ?
- (d) Find a suitable set of predicates  $Preds$  such that the predicate abstraction (iteration of  $post^\#$ ) can prove safety (specified by the TCTL formula  $\mathbf{AG}\neg err$ ) of  $\mathcal{P}_2$ .  
Provide the abstract reachability graph that you obtain.

*Hint:* Consider *some* of the predicates which are used to define the regions for the region transition system (*RTS*) construction.