Master 2 Recherche en Informatique
VTS Module *Due November 2nd 2009*

Timed automata – Exercises

Exercise 1 (Number of regions)
Given a timed automaton $\mathcal{A} = (L, L_0, L_{acc}, \Sigma, \mathcal{X}, E)$ with maximal constant M, justify that the number of standard regions is bounded by

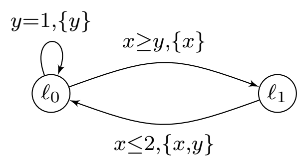
$$2^{|\mathcal{X}|} \cdot |\mathcal{X}|! \cdot (2M+2)^{|\mathcal{X}|}$$

Exercise 2 (Expressivity of diagonal constraints)
Prove that timed automata with diagonal constraints are no more expressive than classical timed automata.
Hint: For every guard $x - y \leq c$, build two copies of the original timed automaton: In the first one, $x - y \leq c$ holds and in the second one $x - y > c$ holds.

- Detail the construction (in particular when to change copy) and its correctness.
- What is the size of the resulting diagonal-free timed automaton?

Exercise 3 (Regions for diagonal constraints)
Give the minimal set of regions compatible with any timed automaton with diagonal constraints which has two clocks and maximal constant M. Justify that the latter regions are equivalence classes of a time-abstract bisimulation between states of the timed automaton.

Exercise 4 (Minimal set of regions)
For the automaton below, give the minimal set of regions which is compatible with constraints, time-elapsing and resets.



Exercise 5 (Expressivity of ε-transitions)
Prove that timed automata with ε-transitions are strictly more expressive than classical timed automata. Hint: you may use the following language: $\mathcal{L}_\varepsilon = \{(a, t_1) \cdots (a, t_n) \mid \forall k,\ t_k \bmod 2 = 0\}$.

Exercise 6 (Updatable timed automata)
The goal of this exercise is to prove that reachability is undecidable for updatable timed automata. Recall that in updatable timed automata, updates of the clocks are of the following form: $x :\bowtie y + c$, with $\bowtie \in \{<, \leq, =, \geq, >\}$.
In order to prove the undecidability result, we reduce the termination problem of a two-counter machine (which is known to be undecidable). Given $\mathcal{M}$ a two-counter machine, we thus build $\mathcal{A}$ a timed automaton with updates such that $\mathcal{M}$ terminates if and only if $\mathcal{A}$ accepts at least one timed word. Hence reachability in $\mathcal{A}$ is at least as hard as termination of $\mathcal{M}$, which is undecidable.
Let c and d be the counters of $\mathcal{M}$. Timed automaton $\mathcal{A}$ contains three clocks: x which encodes the value of c, y which represents the contents of d and an auxilliary clock z.
A computation of $\mathcal{M}$ is encoded by a timed word, and the computation terminates if and only if the timed word is accepted by $\mathcal{A}$. In this exercise we focus on the encoding of the two basics operations on counters: incrementation and decrementation. We prove that they can be simulated by timed automata (1) with decrementing clocks, or (2) with incrementing clocks and diagonal constraints.

Decrementing clocks Prove that updates of the form $x := x - 1$ lead to undecidability. In particular, design modules (sub-automata of the global timed automaton) simulating:

- the increment of a counter; if entering the module, $x = \alpha$, $y = \beta$ and $z = 0$, at the end of the module clocks should satisfy: $x = \alpha + 1$, $y = \beta$ and $z = 0$.
- the decrement of a counter (provided it is positive).

Incrementing clocks Using an additional clock w prove that updates of the form $x := x+1$ lead to undecidability of the reachability problem for updatable timed automata *with diagonal constraints.*

Exercise 7 (Difference bounded matrix)
Give the minimal DBM for the conjunction of all constraints appearing in the automaton of Exercise 4.

Feel free to send an e-mail at `nathalie.bertrand@irisa.fr` if you have questions.