TECHNICAL UNIVERSITY OF DENMARK

Written examination, May 29, 2020

Course: Modelling and Analysis of Real-Time Systems Course no. 02224

Aids allowed: All

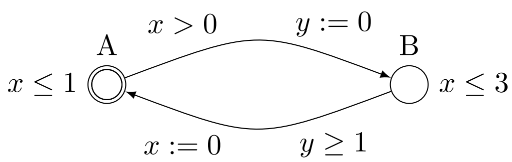
Exam duration: 2 hours

Weighting: PROBLEM 1: approx. 25 % PROBLEM 3: approx. 40 %
PROBLEM 2: approx. 35 %

You are encouraged to briefly justify your answers. If you happen to be in doubt about the precise meaning of a question, you should write down how you choose to understand it.

PROBLEM 1 (approx. 25 %)

In this problem we consider the following timed automaton P with two locations: A and B.



The timed automaton P

Question 1.1:

(a) Give two states of the transition system underlying the timed automaton P.

(b) Give two transitions of the transition system underlying the timed automaton P, where one transition should be a discrete transition and the other a progress of time.

Question 1.2:

A region automaton is a finite automaton constructed on the basis of a timed automaton.

(a) A *clock region* may be of one of the following types: A *point*, a bounded *line segment*, an unbounded *half line (ray)*, a *bounded area*, or an *unbounded area*. Present an example of each region type for P.

(b) How many clock regions are there of the different types and how many in total?

(c) Present two states of the region automaton for P.

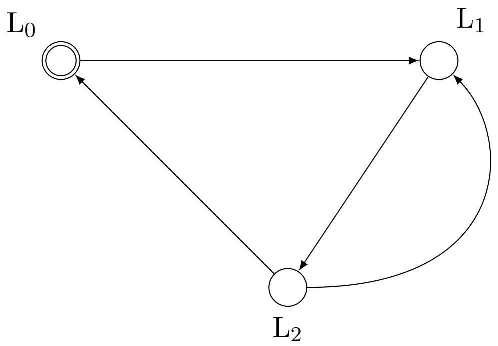
(d) Provide three consecutive transitions of the region automaton for P, where at least one transition should be a discrete step. [Transitions are consecutive if the end state of one transition is the start state of the next one.]

Question 1.3:

Let the automaton P' be like P with the A-invariant $x \leq 1$ removed. Explain why P' may not be considered a sound model.

PROBLEM 2 (approx. 35 %)

Consider the following timed automaton Q with three locations L_0, L_1 and L_2.



The timed automaton Q

Question 2.1: Extend this automaton Q so that

- a stay in L_1 lasts for at most 2 seconds (use a clock `x` to express this property),
- a stay in location L_2 lasts for at least 3 seconds and at most 5 seconds (the clock `x` should be reused to express this property), and
- if L_0 is left, then L_0 must be reentered between 15 and 20 seconds later (use a clock `y` to express this property).

Question 2.2: For each of the following Uppaal queries, determine whether it is satisfied by your extension of Q. The answers must be accompanied with brief justifications.

(a) `E<> Q.L2`

(b) `A[] (Q.L2 imply y > 0)`

(c) `Q.L2 --> Q.L1`

(d) `A[] (y > 17 imply (Q.L0 or Q.L2 ) )`

Question 2.3: Formulate each of the following properties as Uppaal queries:

(a) It is possible that L_2 is never reached.

(b) A stay in L_1 will always lead to L_2.

(c) It is possible that the value of `y` will never exceed 15.

(d) In L_2, the value of `y` cannot exceed the value of `x` by more than 17.

Question 2.4:

(a) For your extension of Q, determine the minimum and maximum number of times the transition from L_2 to L_1 may be taken from the moment where L_0 is left till L_0 is reentered again.

(b) How would you use Uppaal to verify these minimum and maximum numbers?
Hint: Further extension of Q is allowed.

PROBLEM 3 (approx. 40 %)

The questions in this problem can be solved independently of each other.

A real-time system to be run on a single-processor computer has three periodic tasks a, b and c with the following parameters:

	T	C
a	8	3
b	20	7
c	40	3

The deadline of each task is equal to its period. Initially, the tasks are assumed to be independent.

Question 3.1:

(a) Calculate the load of each task and the total load of the system.

(b) Discuss whether schedulability of the task set using fixed-priority scheduling (FPS) can be guaranteed based on the load figures.

In the given system, the set of task is to be scheduled by a preemptive fixed-priority scheduler (FPS) using rate monotonic priority assignment.

Question 3.2:

Calculate the response time for each of the three tasks. Show all intermediate response time estimates.

Now, a protected, shared resource M is to be used by all three tasks. Each task is always going to use the resource M for exactly 1 of its computation time units C. Furthermore, the usage of M will take place at the end of the computation for each of the three tasks.

Recall that the *blocking time* B_t of a task t is the maximum time the task t may experience being suspended while *lower priority tasks* execute.

Question 3.3:

(a) Explain why the blocking times of tasks b and c are 1 and 0 respectively.

(b) Determine the blocking time B_{a} of task a and justify your answer.

[Notice that blocking of task a may be subject to *priority inversion.*]

(c) Explain why the task set using M is not schedulable for the given priority assignment.

(d) Show that the task set using M may become schedulable by choosing another priority assignment:

State the new priority assignment, determine the blocking time for each task and calculate the response time for each task taking its blocking time into account.