Theory of Hybrid Systems
Prof. Dr. Erika Ábrahám
RWTH Aachen University, Informatik 2 · D-52056 Aachen · GERMANY
http://ths.rwth-aachen.de/

Modeling and Analysis of Hybrid Systems - SS 2015

Series 3

Exercise 1

Consider an elevator that services 4 floors numbered 0 through 3. There is an elevator door at each floor with a call-button and an indicator light that signals whether or not the call-button has been pushed. If the light is on then we say that the corresponding floor is requested. The request is served (and the corresponding light is switched off) when the elevator stays at the given floor and the floor door is open.

Present a set of atomic propositions - try to minimize the number of them - that are needed to describe the following properties of the elevator system as LTL formulae and give the corresponding LTL formulae:

(a) The doors are "safe", i.e., a floor door is never open if the elevator is not staying there.

(b) Any requested floor will eventually be served.

(c) Again and again the elevator stays at floor 0.

(d) If the top floor is requested then the elevator does not stop on any other floor before the top floor is served.

(e) Eventually there will be a last request, i.e., there is a time point after which no floor is requested any more.

Is it also possible to give a CTL formula for each of the properties above?

Exercise 2

A transition system TS is given in Figure 1. Decide whether $TS \models \Phi$ where $\Phi = A\mathcal{G}A\mathcal{F}a$. Please sketch the main steps of the CTL model-checking algorithm. *(Note: To eliminate syntactic sugar, you can use $A\mathcal{F}\varphi \equiv A true\, \mathcal{U}\ \varphi$ and $A\mathcal{G}\varphi \equiv \neg E\mathcal{F}\neg\varphi$.)*

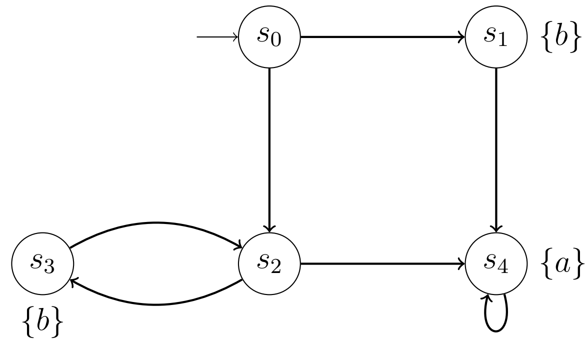
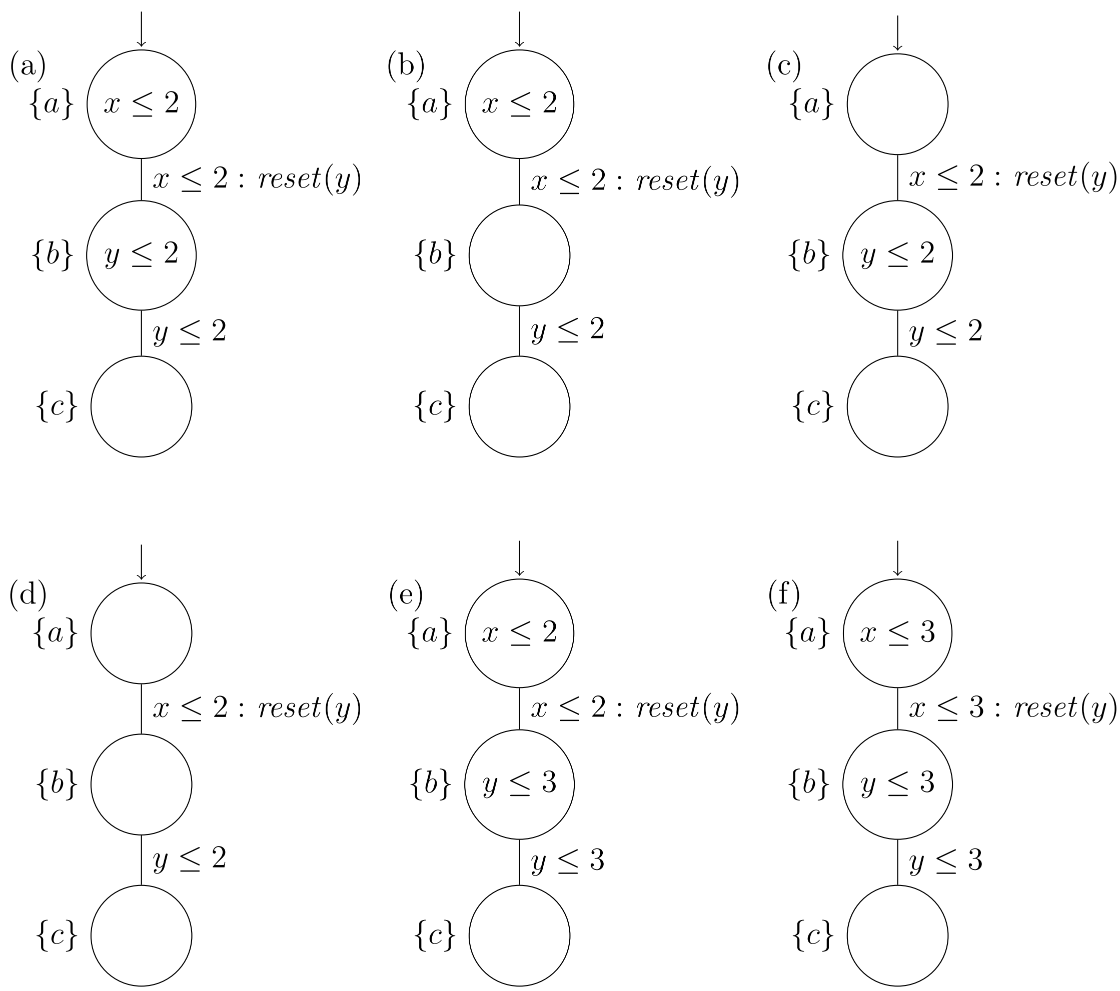


Figure 1: The transition system TS

Exercise 3

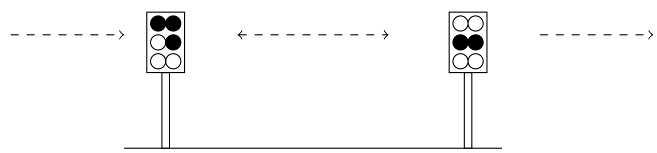
Consider the following six timed automata:



Give for each automaton a TCTL formula that distinguishes it from all other ones. It is only allowed to use the atomic propositions a, b and c and clock constraints.

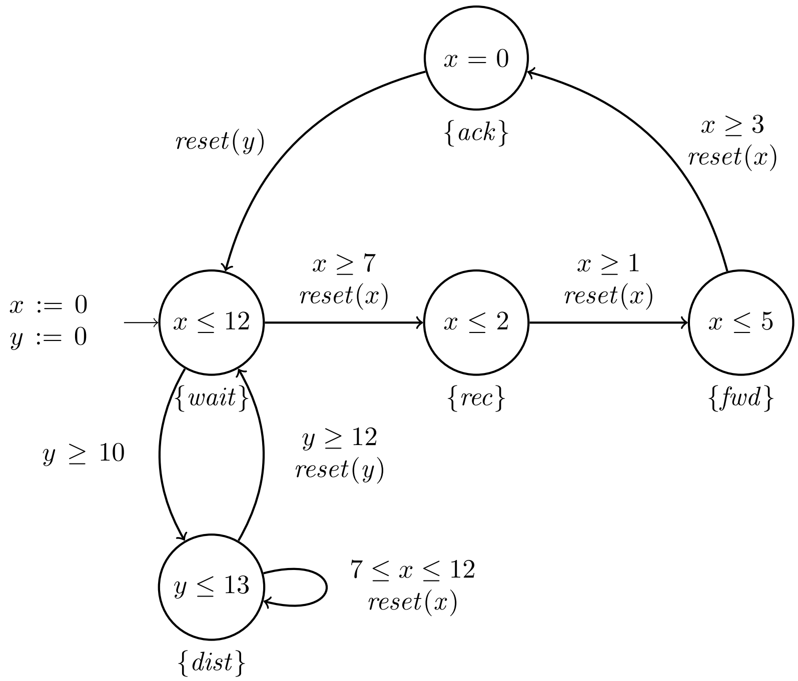
Exercise 4

The "clacks" are a visual telegraph tower system operated by the "Grant Trunk Company" of Ankh-Morpork (cf. Terry Pratchett: "Going postal"). It consists of a network of semaphore towers located about 20 miles from each other spread all over Discworld. Each tower has 6 semaphores which can show either a black panel or a white panel. Each tower is operated by a "clacks operator", whose task it is to watch his predecessing tower and in case there is a message it has to forward the message to the successor tower and after that send back an acknowledgement to the predecessor.



- For each tower, the time till the first incoming message and between two incoming messages from the predecessor is between 7 and 12 minutes.
- As it is very boring to sit and wait for a message, after 10 minutes of concentrated waiting the operator can get distracted, and then he or she is distracted for at least 2 and at most 3 minutes. When the operator is distracted, incoming messages will be lost. When the operator is not distracted, incoming messages will be successfully received.
- The operator needs between 1 and 2 minutes to forward a successfully received message.
- After forwarding, the operator needs another 3 to 5 minutes to send back an acknowledgement to the predecessor.

A timed automaton modelling one clacks-tower is given below, the set of atomic propositions is AP= $\{wait, rec, fwd, ack, dist\}$:



Please give suitable TCTL-formulas, which formalize the following statements:

a) Each successfully received message is acknowledged within 2 minutes. (To assure that the acknowledgment is for the given received message, state that the waiting state is avoided between reception and acknowledgement.)

b) It cannot happen that all messages get lost.

c) It is possible that a message gets lost within the first 10 minutes.

Which of the above formulas holds for the modelled system? Please give reasons for your answer.

Exercise 5

Please give a timed automaton for the following system. You can use as many clocks as you want, but you are restricted to use 4 locations, which are distinguished by the atomic propositions AP= $\{ferry_{left}, ferry_{right}, process\_cargo, travel\}$.

A river can be crossed by taking a ferry which has the following properties:

- Initially the ferry is on the left side of the river ($ferry_{left}$).
- Initially and after each unloading, the ferry waits 1-2 minutes for a new customer ($ferry_{left}/ferry_{right}$).

- Once a customer arrives, the ferry is loaded (*process_cargo*), it crosses the river (*travel*), and it is unloaded (*process_cargo*).
- Loading, crossing and unloading take exactly 10 minutes each.

Hint: You can encode certain properties by a clever usage of different clocks, resets and guards.