Aalto University,
Department of Information and Computer Science

Jori Dubrovin,
Antti Siirtola

T-79.4302 Parallel and Distributed Systems
Examination, 27 May 2013

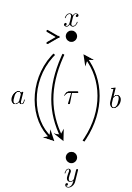
Write down on every answer sheet: the name of the course, the course code, the date, your name, your student id, and your signature.

Calculators are NOT allowed.

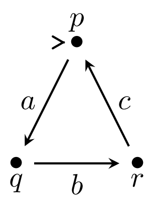
To pass the course, you also need to have passed the quizzes and home assignments in Autumn 2012.

Assignment 1. Consider the parallel composition of the following LTSs $L_i = (\Sigma_i, S_i, s_i^0, \Delta_i)$.

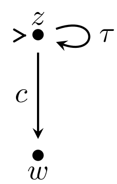
L_1:
$\Sigma_1 = \{a, b\}$



L_2:
$\Sigma_2 = \{a, b, c\}$



L_3:
$\Sigma_3 = \{c\}$



(a) List all pairs of the independent actions of the parallel composition $L_1 \parallel L_2 \parallel L_3$. Write actions as tuples (t_1, t_2, t_3), where each $t_i \in \Delta_i \cup \{-\}$. (2p)

(b) Construct the reachable part of the asynchronous product LTS $L = L_1 \parallel L_2 \parallel L_3$. (2p)

(c) List all reachable states of L that are deadlocks. (2p)

(d) List all reachable states of L in which a livelock exists. (2p)

(e) List all reachable states of L in which a conflict occurs. (2p)

(f) For each reachable state s of L with no conflict, justify why there is no conflict in s. (2p)

Assignment 2. Consider the Kripke structure $M = (S, s^0, R, L)$ with $S = \{s_0, s_1, s_2, s_3, s_4\}$, $s^0 = s_0$, $R = \{(s_0, s_1), (s_1, s_2), (s_2, s_3), (s_3, s_4), (s_4, s_1), (s_4, s_2)\}$, and the function L defined by $L(s_0) = \emptyset$, $L(s_1) = L(s_4) = \{power\}$, $L(s_2) = L(s_3) = \{ready, power\}$. For each of the LTL formulas below, check whether the formula holds in M or not. If the formula holds, give a brief explanation (max 5 lines of text) why the formula holds. If the formula does not hold, give a counterexample execution of M and explain why it violates the formula.

(a) $\mathbf{G}\,(\mathit{ready} \Rightarrow \mathit{power})$ (3p)

(b) $\mathbf{G}\,\mathbf{F}\,\mathit{ready}$ (3p)

(c) $\mathbf{F}\,\mathbf{G}\,\mathit{power}$ (3p)

(d) $\mathbf{G}\,(\mathit{ready} \Rightarrow \mathbf{X}\,\mathbf{X}\,\mathbf{X}\,\mathbf{X}\,\mathit{ready})$ (3p)

More assignments on the second page

Assignment 3.

(a) Briefly explain what is the benefit of bitstate hashing and how it works. (3p)

(b) Briefly explain what is the benefit of partial order reduction and how it works. (3p)

Give the formalisation of the following properties as past safety formulas.

(c) Processes 0 and 1 are never at the same time in the critical section. Use the atomic propositions cs_0 ("process 0 is in the critical section") and cs_1 ("process 1 is in the critical section"). (2p)

(d) If the lock is released, it has been locked in the past. Use the atomic propositions *release* ("the lock is released") and *lock* ("the lock is locked"). (2p)

(e) If the alarm is on, the system has crashed in the past and has not been reset after the latest crash. Use the atomic propositions *alarm* ("the alarm is on"), *crash* ("the system crashes"), and *reset* ("the system is being reset"). (2p)

Assignment 4. Consider the following LTSs over $\Sigma = \{a, b\}$.

L_1: L_2:

(a) Construct a deterministic finite state automaton A' that recognizes the language $\Sigma^* \setminus traces(L_2)$. (2p)

(b) See A' as an LTS L' and compute the asynchronous product LTS $P = L_1 \parallel L'$. Explain how P can be used to argue that $L_1 \leq_{tr} L_2$. (2p)

(c) Does $L_1 \leq_{sim} L_2$ hold? Justify your answer. (2p)

(d) Does $L_2 \leq_{sim} L_1$ hold? Justify your answer. (2p)

(e) From the theory of LTSs, define "bisimulation relation". (2p)

(f) Does $L_1 \sim L_2$ hold? Justify your answer. (2p)