CSL750: Foundations of Automatic Verification

May 7, 2014 **Take-home major exam**

Marks Distribution: 10 + (5+5+10) + (8+8) + (8+(8+4)+8+2*(10+3)) = 100

(To be submitted on neatly stapled A4-sheets by 11:00 AM 12 May 2014)

1. The *modal depth* of a HML formula is defined by structural induction and represents the maximum length chain of modal prefixes that the formula has.

   Let $\sim_n$ be the inductive definition of bisimulation where $s \sim_0 t$ for all states s, t and $s \sim_{n+1} t$ whenever for all actions a, $s \xrightarrow{a} s'$ implies for some t', $t \xrightarrow{a} t'$ and $s' \sim_n t'$ and for all actions b, $t \xrightarrow{b} t''$ implies for some s'', $s \xrightarrow{a} s''$ and $s'' \sim_n t''$.

   Prove that there exists a HML formula of *modal depth* at most n which can distinguish states s and t if $s \not\sim_n t$.

2. Consider Peterson's mutual exclusion problem.

   (a) Extend the solution (with minumum number of changes to the original code − write out your code for 2 processes in your favourite language) to 3 processes and prove the mutual exclusion property. Your design should involve 3 symmetric processes which look identical except for the numbers which represent process ids.

   (b) Does your solution generalise to $n > 3$ processes also. Explain what are the possible drawbacks of extending your solution to $n > 3$ processes.

   (c) Express the properties required for "mutually-exclusive" access to the critical section in LTL. Prove the required properties for your solution for $n = 3$

3. Consider the following scheme of transforming a Labelled transition system (LTS) $\mathcal{L} = \langle S, A, \rightarrow, s_0 \rangle$ to a Decorated transition system (DTS) $\mathcal{D} = \langle S, \rightarrow, s_0, \delta \rangle$ with $AP = \{via_a \mid a \in A\}$ such that $s \xrightarrow{a} s'$ in $\mathcal{L}$ <u>if and only if</u> $via_a \in \delta(s')$

   (a) Show that not every formula of HML on $\mathcal{L}$ is expressible as a CTL formula on $\mathcal{D}$ (you may want to illustrate it by constructing your own LTS $\mathcal{L}$ and obtaining a corresponding DTS $\mathcal{D}$).

   (b) Show that not every CTL formula on $\mathcal{D}$ is expressible as an HML formula in $\mathcal{L}$ (here again you might want to construct your own new LTS and obtain the corresponding DTS).

4. A timed automaton (TA) is said to be *deterministic* if for every location l, for any two outgoing edges from l that are labelled with the same action a, the guards g_1 and g_2 on the two edges do not overlap, i.e. $[\![g_1]\!] \cap [\![g_2]\!] = \emptyset$.

   (a) Show that there exists a TA A,corresponding to which there does *not* exist a deterministic TA B such that $L(A) = L(B)$.

   (b) Consider a TA A with the set of actions $Act_A \subseteq Act$ appearing on the edges of A. The set of clocks of A is defined as $C = \{x_a \mid a \in Act_A\}$. On every edge labelled with an action $a \in Act_A$ in A, only clock x_a is reset (i.e. no other clock is reset on the edge labelled with action a). Show that such a TA can be determinized, i.e. given a non-deterministic TA A, one can construct a deterministic TA B such that $L(A) = L(B)$. Write the steps for determinizing the TA and analyze the time complexity of your procedure.

   (c) Show with an example, that the construction of the zone graph discussed in class may not terminate.

   (d) Write algorithms for checking time abstracted delay bisimulation and time abstracted observational bisimulation using zone graph. Discuss the time complexity of your algorithms.