Technische Universiteit **Eindhoven** University of Technology

Student name:

Student number:

Examination cover sheet

(to be completed by the examiner)

Course name: Algorithms for Model Checking | Course code: 2IW55

Date: 23-06-2015

Start time: 13:30 | End time : 16:30

Number of pages: 2

Number of questions: 4

Maximum number of points/distribution of points over questions:100

Method of determining final grade: divide total of points by 10

Answering style: open questions

Exam inspection: With the lecturer

Other remarks:

Instructions for students and invigilators

Permitted examination aids (to be supplied by students):

- ☐ Notebook
- ☐ Calculator
- ☐ Graphic calculator
- ☑ Lecture notes/book
- ☐ One A4 sheet of annotations
- ☐ Dictionar(y)(ies). If yes, please specify:
- ☑ Other: Notes, sheets of annotations and other written material, see also the next page

Important:

- examinees are only permitted to visit the toilets under supervision
- it is not permitted to leave the examination room within 15 minutes of the start and within the final 15 minutes of the examination, unless stated otherwise
- examination scripts (fully completed examination paper, stating name, student number, etc.) must always be handed in
- the house rules must be observed during the examination
- the instructions of examiners and invigilators must be followed
- no pencil cases are permitted on desks
- examinees are not permitted to share examination aids or lend them to each other

During written examinations, the following actions will in any case be deemed to constitute fraud or attempted fraud:

- using another person's proof of identity/campus card (student identity card)
- having a mobile telephone or any other type of media-carrying device on your desk or in your clothes
- using, or attempting to use, unauthorized resources and aids, such as the internet, a mobile telephone, etc.
- using a clicker that does not belong to you
- having any paper at hand other than that provided by TU/e, unless stated otherwise
- visiting the toilet (or going outside) without permission or supervision

Examination Algorithms for Model Checking (2IW55)

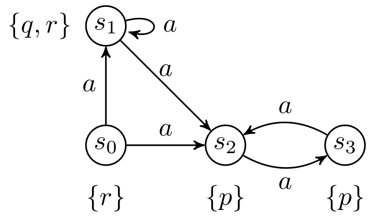
23 June, 2015, 13:30 – 16:30

Important notes:

- The exam consists of four questions.
- Weighting: 1: **25**, 2: **25**, 3: **25**, 4: **25**.
- Carefully read and answer the questions. The book, the course notes and other written material may be used during this examination.
- Use these solutions at your own risk; they may contain typos or even life-threatening flaws.

---

1. Consider the formula ϕ given by $\mathsf{E}\ \mathsf{G}\ \mathsf{E}\ [p\ \mathsf{U}\ q]$, in the context of the following mixed Kripke Structure, where $\{p, q, r\}$ is the set of atomic propositions and a is an action.



(a) Convert formula ϕ to an equivalent formula in the μ-calculus. Compute the nesting depth, the alternation depth and the dependent alternation depth of the formula you obtained. *(10 points)*

Solution: The formula $\mathsf{E}\ \mathsf{G}\ \mathsf{E}\ [p\ \mathsf{U}\ q]$ *translates to:*

$$\nu X.\langle a\rangle X \wedge \mu Y.q \vee (p \wedge \langle a\rangle Y)$$

The nesting depth and the alternation depth of the above formula are both 2; the dependent alternation depth is 1.

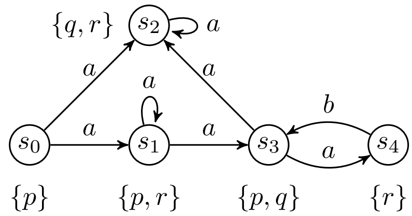
(b) Solve the μ-calculus formula $\nu X.([a]X \wedge \nu Y.\ (q \vee (p \wedge [a]Y)))$ using either the naive model checking algorithm for the μ-calculus, or the Emerson-Lei algorithm. Show the intermediate steps in your computations. *(15 points)*

Solution: We will use the naive algorithm and discuss where Emerson-Lei will improve upon that. The naive algorithm essentially performs two nested fixpoint calculations, which we will detail below; we write X^i and Y^i to indicate X and Y's i-th approximation and Y^i_j to denote the j-th approximation of Y within the i-th approximation of X.

$$\begin{array}{lll}
X^0 & = \{s_0, s_1, s_2, s_3\} & \\
X^1 & = Y^1 & \\
 & \textit{where } Y^1 \textit{ is given by:} & Y^1_0 = \textsf{true} \\
 & & Y^1_1 = \{s_1, s_2, s_3\} \\
 & & Y^1_2 = \{s_1, s_2, s_3\} \\
X^2 & = \{s_0, s_1, s_2, s_3\} \cap Y^2 & \\
 & \textit{where } Y^2 \textit{ is given by:} & Y^2_0 = \textsf{true} \\
 & & Y^2_1 = \{s_1, s_2, s_3\} \\
 & & Y^2_2 = \{s_1, s_2, s_3\} \\
X^3 & = \{s_0, s_1, s_2, s_3\} \cap Y^3 & \\
 & \textit{where } Y^3 \textit{ is given by:} & Y^3_0 = \textsf{true} \\
 & & Y^3_1 = \{s_1, s_2, s_3\} \\
 & & Y^3_2 = \{s_1, s_2, s_3\}
\end{array}$$

So the property holds in all states except for state s_0. The optimisation of Emerson-Lei avoids the recomputations performed to compute Y^2 and Y^3.

2. Consider the mixed Kripke Structure below, where $\{p, q, r\}$ is the set of atomic propositions and a and b are the actions.



Consider the μ-calculus formula ϕ defined as $\nu X.([a]X \wedge \mu Y.(\langle b\rangle \mathsf{true} \vee ([a]Y \wedge \langle a\rangle \mathsf{true})))$. Construct a Boolean equation system that can be used to solve which states of the mixed Kripke structure satisfy ϕ. Solve the resulting Boolean equation system using Gauß Elimination and answer which states satisfy ϕ. *(25 points)*

Solution: We first construct a Boolean equation system by combining the mixed Kripke structure and the formula. This results in the following:

$$
\begin{aligned}
\nu X_0 &= (X_1 \wedge X_2) \wedge Y_0 \\
\nu X_1 &= (X_1 \wedge X_3) \wedge Y_1 \\
\nu X_2 &= X_2 \wedge Y_2 \\
\nu X_3 &= X_2 \wedge X_4 \wedge Y_3 \\
\nu X_4 &= Y_4 \\
\mu Y_0 &= Y_1 \wedge Y_2 \\
\mu Y_1 &= Y_1 \wedge Y_3 \\
\mu Y_2 &= Y_2 \\
\mu Y_3 &= Y_4 \wedge Y_2 \\
\mu Y_4 &= \mathit{true}
\end{aligned}
$$

Solving the above equation system using Gauß Elimination proceeds as follows. Note that we combine the local solving and substitution to the left (up, in this case) and perform some on-the-fly simplification; the equation currently being dealt with by the algorithm is indicated by a $^{(*)}$:

$\nu X_0 = X_1 \wedge X_2 \wedge Y_0$	$X_1 \wedge X_2 \wedge Y_0$	$X_1 \wedge X_2 \wedge Y_0$	$X_1 \wedge X_2 \wedge Y_0$	$X_1 \wedge X_2 \wedge Y_0$	*false*
$\nu X_1 = X_1 \wedge X_3 \wedge Y_1$	$X_1 \wedge X_3 \wedge Y_1$	$X_1 \wedge X_3 \wedge Y_1$	$X_1 \wedge X_3 \wedge Y_1$	*false*	*false*
$\nu X_2 = X_2 \wedge Y_2$	$X_2 \wedge Y_2$	$X_2 \wedge Y_2$	*false*	*false*	*false*
$\nu X_3 = X_2 \wedge X_4 \wedge Y_3$	$X_2 \wedge X_4 \wedge Y_3$	$X_2 \wedge X_4 \wedge Y_2$	*false*	*false*	*false*
$\nu X_4 = Y_4$	*true*	*true*	*true*	*true*	*true*
$\mu Y_0 = Y_1 \wedge Y_2$	$Y_1 \wedge Y_2$	$Y_1 \wedge Y_2$	*false*	*false*	*false*$^{(*)}$
$\mu Y_1 = Y_1 \wedge Y_3$	$Y_1 \wedge Y_3$	$Y_1 \wedge Y_2$	*false*	*false*$^{(*)}$	
$\mu Y_2 = Y_2$	Y_2	Y_2	*false*$^{(*)}$		
$\mu Y_3 = Y_4 \wedge Y_2$	Y_2	$Y_2^{(*)}$			
$\mu Y_4 = \mathit{true}$	*true*$^{(*)}$				

Running the entire procedure just once yields solutions to all variables, except for Y_3. Substituting $Y_2 =$ false to the right (down, in this case, which is permitted since it is a solved equation), solves Y_3. This results in all variables to have solution false, except for $X_4 = Y_4 =$ true. Therefore, ϕ only holds in s_4.

3. This question is composed of four subquestions.

   (a) For *5pt*, give a parity game $G = (V, E, p, (V_\Diamond, V_\Box))$, consisting of 10 vertices and ensure that G is such that:

i. $|V_1| = 1$, $|V_2| = 2$, $|V_3| = 3$ and $|V_4| = 4$, where $V_i \subseteq V$ is the set of vertices with priority i,

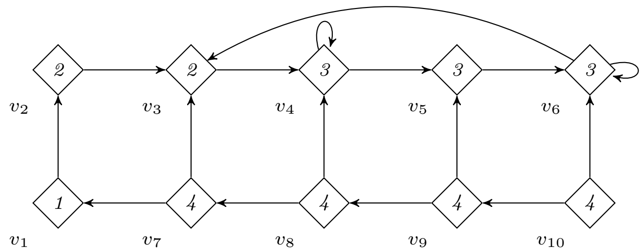
ii. G has at least 16 edges,

iii. all vertices in G are won by player $\Diamond$.

iv. player $\Diamond$ **has no** strategy σ guaranteeing that priority 4 occurs infinitely often on **all** plays consistent with σ (i.e. plays starting in arbitrary vertices),

v. player $\Box$ **has a memoryless** strategy ρ such that for **all** plays π (starting in arbitrary vertices) consistent with ρ, at some point 2 vertices with **odd** priority appear in π, *without* any vertex with priority 2 in between these two vertices.

Solution: There is an easy solution that meets all requirements, which is to construct an even-dominated loop containing all vertices in $|V_3|$ and every other vertex pointing into this loop directly or indirectly. The graph depicted below is an instance:



(b) Prove, for *5pt*, that your game G of question (a) satisfies property (iv).

Solution: observe player $\Box$ does not own any vertices and can therefore not influence the game play; the proof therefore follows if we can show that from any vertex, all paths through the graph have a finite number of vertices with priority 4. But this is obvious, as all paths eventually reach the subgame $\{v_3, v_4, v_5, v_6\}$, from which there is no escape.

(c) For *5pt*, define a memoryless strategy ρ for player $\Box$ and show that the game G of question (a) satisfies property (v).

Solution: since player $\Box$ owns none of the vertices, the domain of the memoryless strategy ρ is empty. Again, every play eventually reaches the subgame $\{v_3, v_4, v_5, v_6\}$, from which there is no escape, and in which all plays must visit at least 3 consecutive vertices with priority 3 without any lower priority in between.

(d) For *10pt*, prove using either Zielonka's recursive algorithm or *Jurdziński's* Small Progress Measures algorithm that your solution meets property (iii). For the recursive algorithm, clearly indicate which subgames are solved in each recursive step. For the Small Progress Measures algorithm, show the intermediate measures.

Solution: we use Zielonka's algorithm, and we only sketch the steps taken by the algorithm. Vertex v_1 has the lowest value in the game, but its $\Box$-attractor is $\{v_1\}$. We must therefore recursively solve the subgame $V \setminus \{v_1\}$. The vertices with the lowest value in the game, viz. $\{v_2, v_3\}$, have an even priority; computing the attractor into $\{v_2, v_3\}$ leads to the entire subgame, which is then won by player $\Diamond$. So the subgame is solved. We must then enter the second recursive call, which is performed on an empty game, as v_1 is attracted into the subgame $V \setminus \{v_1\}$. Hence, the full game is won by player $\Diamond$.

4. Let N be the natural numbers. Let $\mathcal{E}$ be the parameterised Boolean equation system depicted below:

$$\begin{array}{ll} \nu X(n:N) & = \exists k:N.Y(n,k) \\ \nu Y(n:N, k:N) & = ((n=k) \wedge Y(k+1, n+1)) \vee ((n \neq k) \wedge X(k)) \end{array}$$

Compute the solution to $X(0)$, where X is defined by $\mathcal{E}$. Clearly indicate all steps and transformations you use in your computation. *(25 points)*

Solution: Observe that we do not expect that there are parameters that we can eliminate since in Y, all parameters are used. We therefore perform a straightforward symbolic approximation, starting at Y:

$$
\begin{aligned}
Y^0(n,k) &= \mathsf{true} \\
Y^1(n,k) &= ((n = k) \wedge \mathsf{true}) \vee ((n \neq k) \wedge X(k)) \\
&= (n = k) \vee ((n \neq k) \wedge X(k)) \\
Y^2(n,k) &= ((n = k) \wedge ((k+1 = n+1) \vee ((k+1 \neq n+1) \wedge X(n+1)))) \vee ((n \neq k) \wedge X(k)) \\
&= ((n = k) \wedge ((k = n) \vee ((k \neq n) \wedge X(n+1)))) \vee ((n \neq k) \wedge X(k)) \\
&= (n = k) \vee ((n \neq k) \wedge X(k)) \\
&= Y^1(n,k)
\end{aligned}
$$

Next, we substitute the solution to Y in the equation for X:

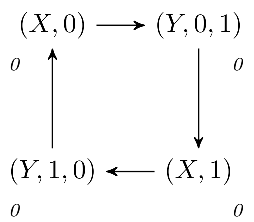
$$\nu X(n : N) = \exists k : N.((n = k \vee ((n \neq k) \wedge X(k))))$$

Solving X again using an approximation yields:

$$
\begin{aligned}
X^0(n) &= \mathsf{true} \\
X^1(n) &= \exists k : N.((n = k \vee n \neq k)) \\
&= \mathsf{true}
\end{aligned}
$$

Hence, $X(n)$ is true *for all values of n; in particular, $X(0)$ is* true.

While not asked for, a proof graph explaining that $X(0)$ is true *is depicted below:*



Note that this proof graph has exactly one infinite path starting in $(X,0)$ and that path is even-dominated. Furthermore, we observe that the proof graph fulfils the true*-dependency graph conditions, as we sketch below:*

- *For vertex $(X,0)$, we observe that*

  $$\llbracket \exists k : N.Y(n,k) \rrbracket (X,0)^{\bullet}_{\mathsf{true}} \varepsilon[n := 0]$$

  is equivalent to $(0,v) \in ((X,0)^{\bullet}_{\mathsf{true}})(Y)$ for some natural number v, which, since $(X,0)^{\bullet} = \{(Y,0,1)\}$, is exactly the case for $v = 1$.
- *For vertex $(X,1)$, we observe that*

  $$\llbracket \exists k : N.Y(n,k) \rrbracket (X,1)^{\bullet}_{\mathsf{true}} \varepsilon[n := 1]$$

  is equivalent to $(1,v) \in ((X,1)^{\bullet}_{\mathsf{true}})(Y)$ for some natural number v, which, since $(X,1)^{\bullet} = \{(Y,1,0)\}$, is exactly the case for $v = 0$.
- *For vertex $(Y,0,1)$, we observe that*

  $$\llbracket ((n = k) \wedge Y(k+1, n+1)) \vee ((n \neq k) \wedge X(k)) \rrbracket (Y,0,1)^{\bullet}_{\mathsf{true}} \varepsilon[n := 0, k := 1]$$

  is equivalent to $1 \in ((Y,0,1)^{\bullet}_{\mathsf{true}})(X)$ which, since $(Y,0,1)^{\bullet} = \{(X,1)\}$, is the case.
- *For vertex $(Y,1,0)$, we observe that*

  $$\llbracket ((n = k) \wedge Y(k+1, n+1)) \vee ((n \neq k) \wedge X(k)) \rrbracket (Y,0,1)^{\bullet}_{\mathsf{true}} \varepsilon[n := 1, k := 0]$$

  is equivalent to $0 \in ((Y,1,0)^{\bullet}_{\mathsf{true}})(X)$ which, since $(Y,1,0)^{\bullet} = \{(X,0)\}$, is the case.

□