Discrete Event Systems

Course code: SSY165

Examination 2014-10-28

Time: 8:30-12:30, Location: M-building

Teacher: Bengt Lennartson, phone 3722

The examination includes 25 points, where grade three requires 10 points, grade four 15 points and grade five 20 points.

The result of this examination will be announced latest on Tuesday *November 11* on the notice board of the division, at the entrance in the south east corner on floor 5 of the E-building. *Inspection* of the grading is done on Tuesday *November 11* and Wednesday *November 12* at 12:30-13:00.

Allowed aids at the examination:

- Standard mathematical tables such as Beta, see also formulas in the end of this examination.
- Pocket calculator.

Good luck!

Department of Signals and Systems
Division of Automatic Control, Automation and Mechatronics
Chalmers University of Technology

1

Show the following set implication

$$A \subseteq C \Rightarrow A \cup (B \cap C) = (A \cup B) \cap C$$

Advice: Show first by predicate logics that $A \subseteq C \Leftrightarrow A \cap C = A$, and then show the remaining part by set expressions.

(4 p)

2

Prove that

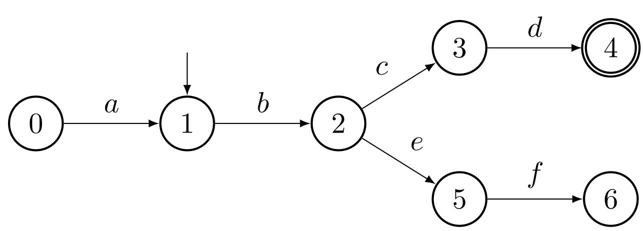
$$\exists x[P(x)] \rightarrow Q \Leftrightarrow \forall x[P(x) \rightarrow Q]$$

by assuming a universal set Ω with a finite number of arbitrary elements

$$\Omega = \{a_1, a_2, \cdots, a_n\}$$

(3 p)

3



a) Formulate explicit predicates $P_r(x)$ and $P_c(x)$ for the reachable and coreachable (backward reachable) states of the automaton above, where x is the state variable taking explicit values according to the given automaton.

(1 p)

b) Formulate a general statement P_b for an arbitrary automaton based on the reachable and coreachable predicates $P_r(x)$ and $P_c(x)$, which is true if and only if the arbitrary automaton has any *blocking states*. Verify that this statement is correct for the example automaton.

(2 p)

c) Formulate in the same way as in task b) a general statement P_{nb}, including the conditional connective $\rightarrow$, specifying that an arbitrary automaton is *nonblocking*. Show that the negation of this statement is equivalent to the blocking state specification in task b).

(2 p)

4

Consider a plant P with the language

$$L(P) = \overline{a(c + db(bc + cd)) + dc}$$

and a specification Sp given by the marked language

$$L_m(Sp) = a(c + dbbc) + dc$$

Assume that the events c and d are uncontrollable, while a and b are controllable.

a) Formulate automata for the languages $L(P)$ and $L_m(Sp)$ with minimal number of states.

(1 p)

b) Generate a controllable and nonblocking supervisor, by the fix point algorithm presented in the lecture notes. Show the resulting automaton after each Backward_Reachability computation.

(3 p)

5

a) Generate a minimal automaton G for the language

$$L = fac^* + afbc^*$$

(1 p)

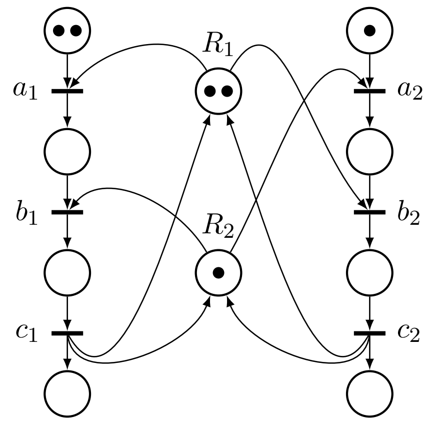
b) Show that this automaton is diagnosable when the event f is an unobservable fault event, while the rest of the events are observable.

(2 p)

c) Generate a diagnoser for this system

(1 p)

6



a) Generate the reachability graph for the Petri net above, and identify any blocking states.

(3 p)

b) Modify the Petri net to avoid the blocking states, either by adding arcs and optionally places, or extra guards at some transitions based on the tokens in the net. Motivate that your solution is maximally permissive, i.e. your solution only removes the blocking states but no nonblocking states.

(2 p)

Solution to the exam 2014-10-28 in Discrete Event Systems

BL141101

1. $A \subseteq C \Leftrightarrow A \cap C = A$ since corresponding predicates show that

$$\forall x(((x \in A \wedge x \in C) \rightarrow x \in A) \wedge (x \in A \rightarrow (x \in A \wedge x \in C)) \Leftrightarrow$$

$$\forall x((x \notin A \vee x \notin C \vee x \in A) \wedge (x \notin A \vee (x \in A \wedge x \in C)) \Leftrightarrow$$

$$\forall x((\underbrace{x \notin A \vee x \in A}_{T} \vee x \notin C) \wedge (\underbrace{x \notin A \vee x \in A}_{T}) \wedge (x \notin A \vee x \in C)) \Leftrightarrow$$

$$\forall x((\underbrace{T \wedge x \notin C}_{T}) \wedge T \wedge (x \in A \rightarrow x \in C)) \Leftrightarrow \forall (x \in A \rightarrow x \in C)$$

$$(A \cup B) \cap C = (A \cap C) \cup (B \cap C) \quad A \cap C = A \Rightarrow$$

$$(A \cup B) \cap C = A \cup (B \cap C)$$

2.

$$\exists x [P(x)] \rightarrow Q \Leftrightarrow (P(a_1) \vee P(a_2) \vee \ldots \vee P(a_n)) \rightarrow Q \Leftrightarrow$$

$$(\neg P(a_1) \wedge \neg P(a_2) \wedge \ldots \wedge \neg P(a_n)) \vee Q \Leftrightarrow$$

$$(\neg P(a_1) \vee Q) \wedge (\neg P(a_2) \vee Q) \wedge \ldots \wedge (\neg P(a_n) \vee Q) \Leftrightarrow$$

$$\forall x [\neg P(x) \vee Q] \Leftrightarrow \forall x [P(x) \rightarrow Q]$$

3.

a) $P_r(x): 1 \leq x \leq 6$

$P_c(x): 0 \leq x \leq 4$

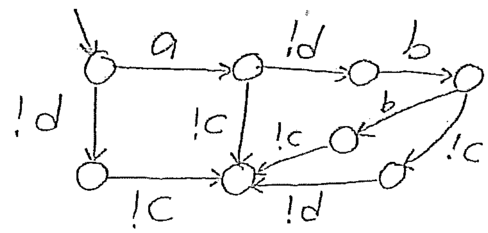
b) $P_b \triangleq \exists x [P_r(x) \wedge \neg P_c(x)] \Leftrightarrow$

$$\Leftrightarrow \exists x [(1 \leq x \leq 6) \wedge (x \geq 5)] \Leftrightarrow \exists x [x = 5 \vee x = 6] = \text{True}$$

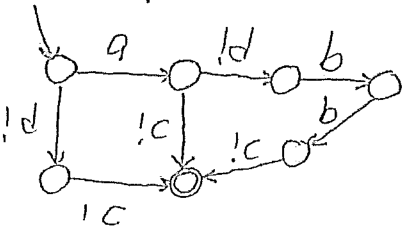
c) $P_{nb} \triangleq \forall x [P_r(x) \rightarrow P_c(x)] \Leftrightarrow \forall x [\neg P_r(x) \vee P_c(x)]$

$$\Leftrightarrow \neg \exists x [P_r(x) \wedge \neg P_c(x)] \Leftrightarrow \neg P_b$$

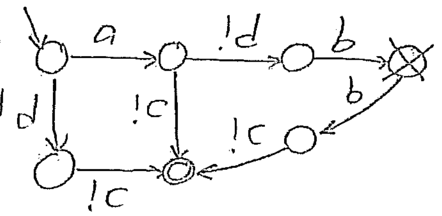
4. P



$S_p = P \| S_p$

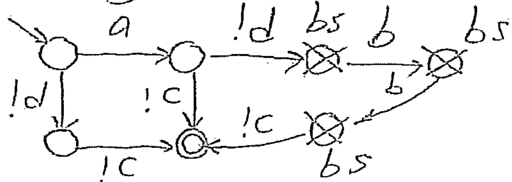


$S_0 = P \| S_p$

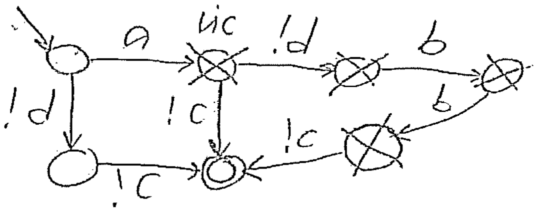


Uncontrollable state, since P can make !c but not S_0..

Blocking and forbidden states (bs)

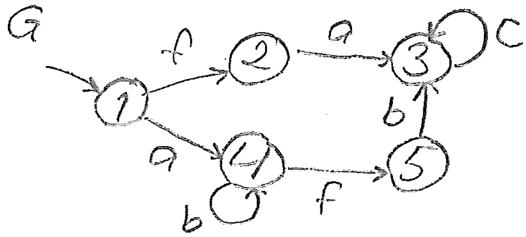


Extended uncontrollable state (uc)



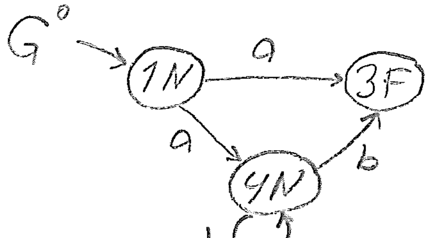
5. G

a)

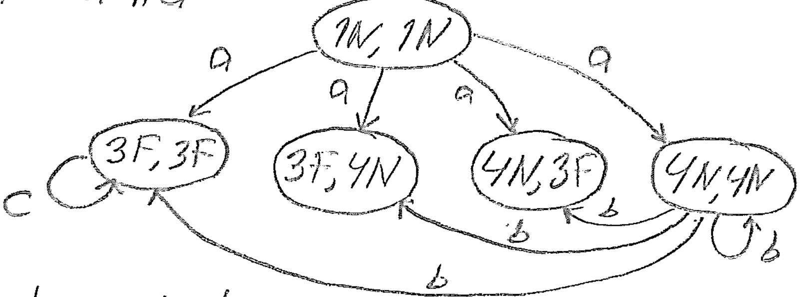


$\Sigma_o = \{a, b, c\}$ $\Sigma_{uo} = \{f\}$

b) G^o

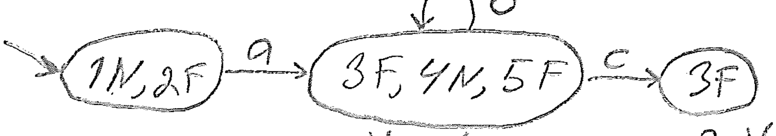


$G^d = G^o \| G^o$



Diagnosable since G^d does not include loops with only mixed F,N or N,F states.

c) G_{diag}



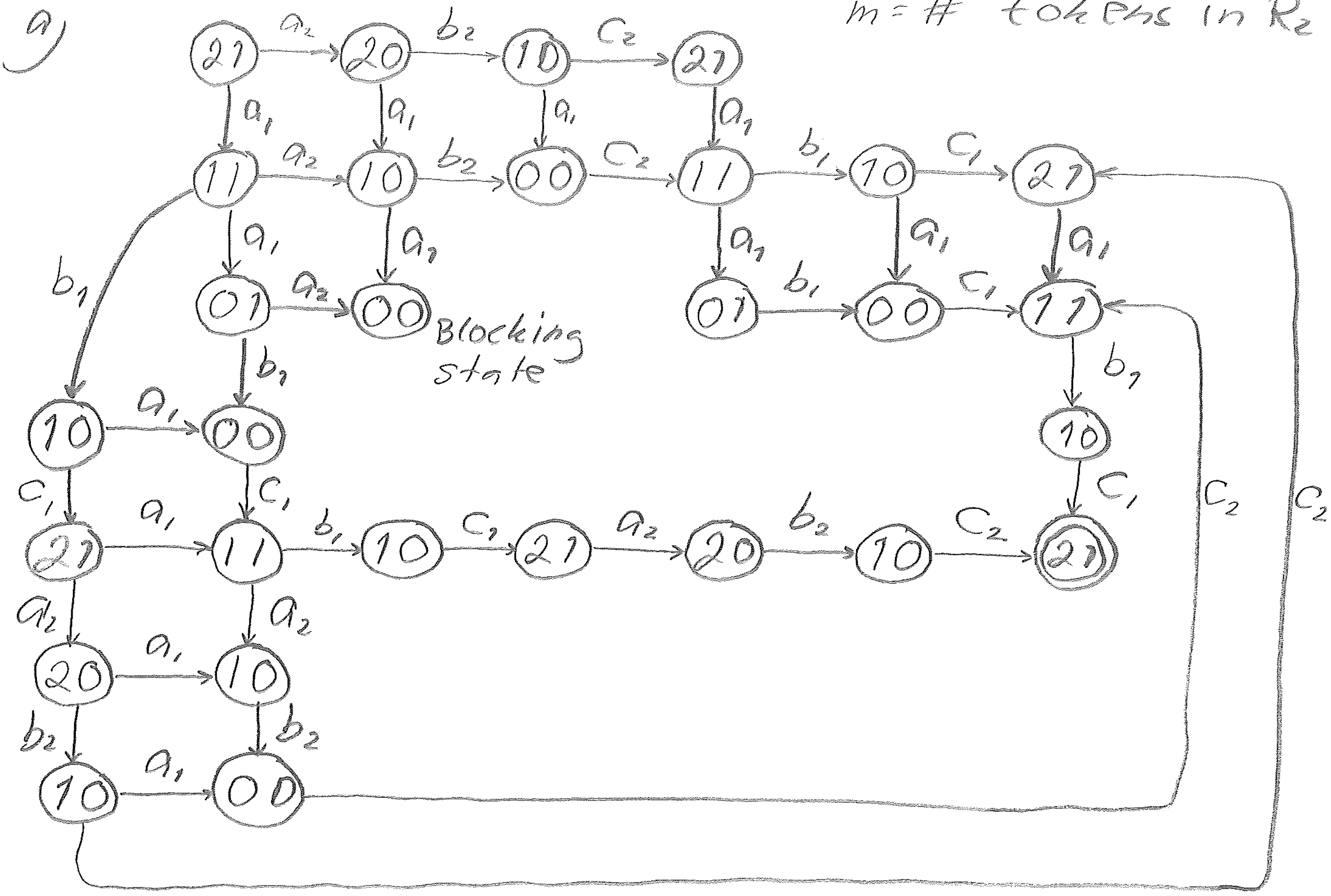
combined failure non failure state

failure state

6.

a)

(nm) n = # tokens in R_1
m = # tokens in R_2



b) The sequences $a_1 a_1 a_2$, $a_1 a_2 a_1$, $a_2 a_1 a_1$ are not allowed. They are avoided by the extra control place C