EE4-57

SOLUTIONS: DISCRETE EVENT SYSTEMS
MASTER IN CONTROL

1. Exercise

a) The Buffer occupancy when packets arrive and depart according to the rules prescribed can be modeled by the automaton shown in Fig. 1.1.

b) The non-deterministic automaton G_N is obtained simply replacing events a_2 and a_4 by a in G. See Fig. 1.2.

c) Given a current level of occupancy x, the map $f_N(x,a)$ returns $\{x+2,x+4\} \cap \{0,1,2,\ldots,6\}$. This is indeed an interval of the same parity of x. Hence, for any even interval I, $f(I,a) = \bigcup_{x\in I} f_N(x,a)$ is an interval of the same parity of I. Instead $f_N(x,d_3) = \{x-3\}$, provided $x \geq 3$ (and is undefined otherwise). Notice that $x-3$ is an integer of parity opposite to that of x. Hence, $f_N(I,d_3) = \bigcup_{x\in I} f_N(x,d_3)$ which is an interval of parity opposite to that of I.

d) The observer automaton G_O can be built as in Fig. 1.3.

e) It is not possible to enter the state $\{4\}$ through a d_3 event as this would imply an initial occupancy of 7. The only possibility of entering such a state is therefore through an a event. However, two only possibilities arise in this respect, initial state with occupancy 0 or 2 (or both simultaneously). In both cases $\{4\}$ will not be the only state as the uncertainty associated to a events would respectively lead to states $\{2,4\}$ or $\{4,6\}$.

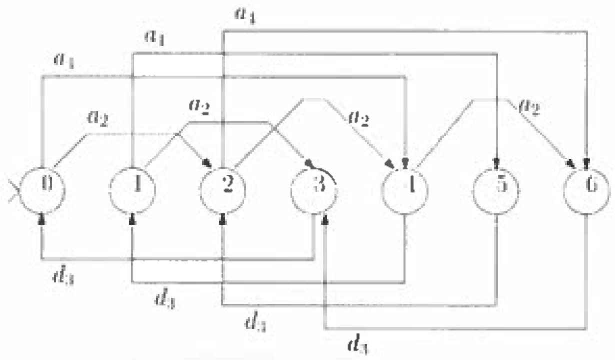

Figure 1.1 The automaton G

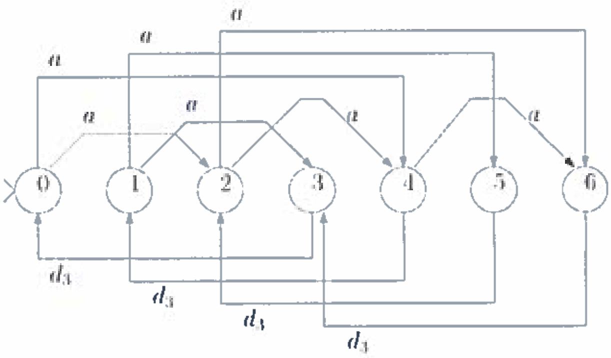

Figure 1.2 The automaton G_N

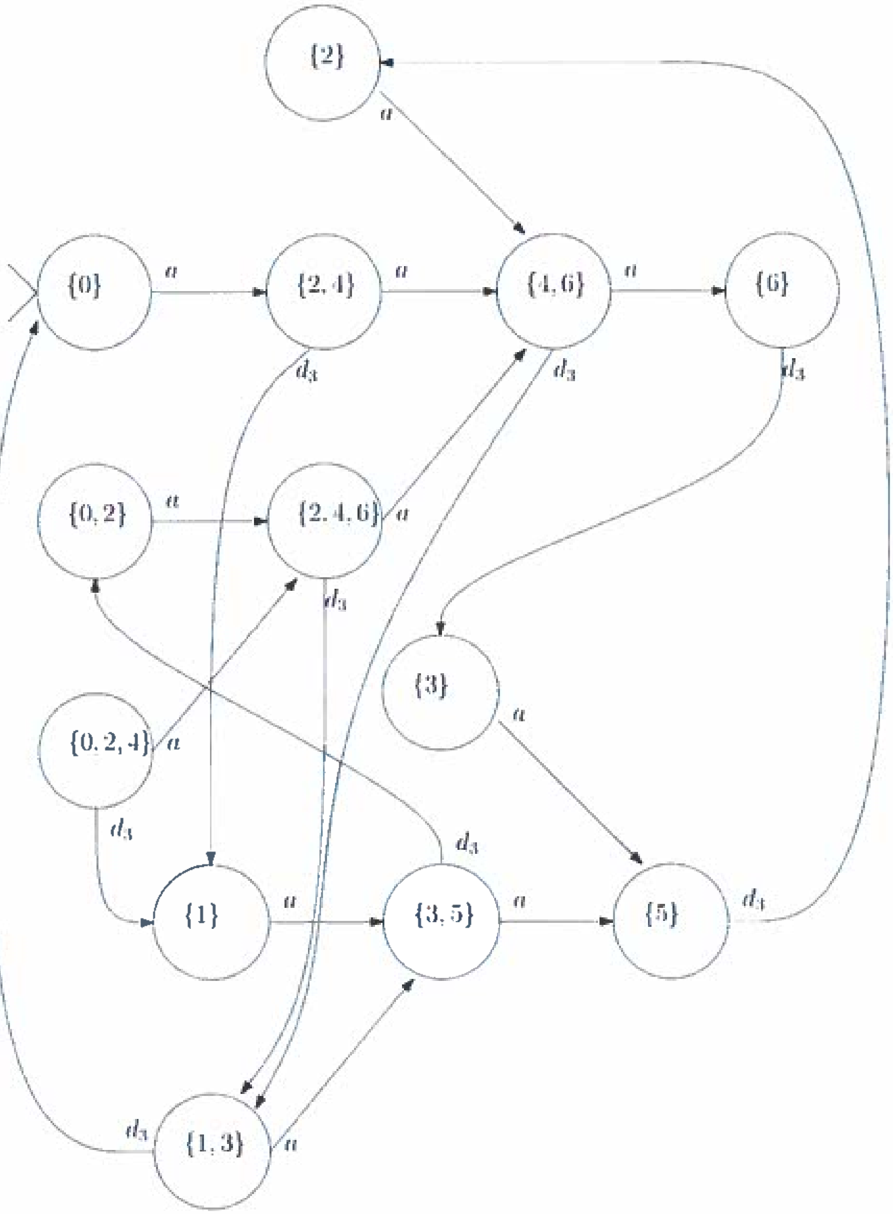

Figure 1.3 The automaton G_O

x_1	V				
x_2	V				
x_3	V				
x_4	V				
x_5		V	V	V	V
	x_0	x_1	x_2	x_3	x_4

Figure 2.1 Flagged terminal vs. non terminal states

x_1	V				
x_2	V	V			
x_3	V	V			
x_4	V		V	V	
x_5		V	V	V	V
	x_0	x_1	x_2	x_3	x_4

Figure 2.2 Table of equivalence classes

2. Exercise

a) Flagging up pairs of states which are terminal and not terminal yields the table shown in Fig. 2.1. Next, we flag pairs $\{x_a,x_b\}$ for which $\Gamma(x_a) \neq \Gamma(x_b)$. This yields the table in Fig. 2.2. Moreover, for each unflagged pair, every enabled event e still leads to an unflagged pair. Hence, the table in Fig. 2.2 also shows all the equivalent (unflagged) state pairs. The equivalent classes are $\{x_0,x_5\},\{x_1,x_4\},\{x_2,x_3\}$.

b) The automaton $G_{\min}$ has three states: $\{x_0,x_5\},\{x_1,x_4\},\{x_2,x_3\}$. Its transition diagram is shown in Fig. 2.3.

c) In order to only allow a and c events to alternate between each other we may build a specification automaton G_K as in Fig. 2.4

d) To verify if K is controllable we build the parallel composition $G_{\min}||G_K$. This

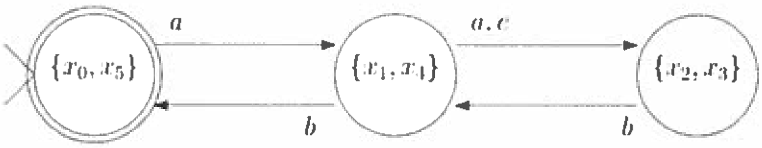

Figure 2.3 The transition diagram of $G_{\min}$

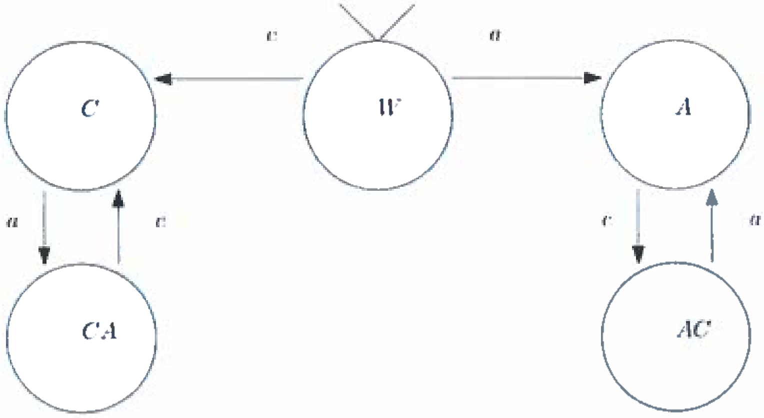

Figure 2.4 The transition diagram of G_K

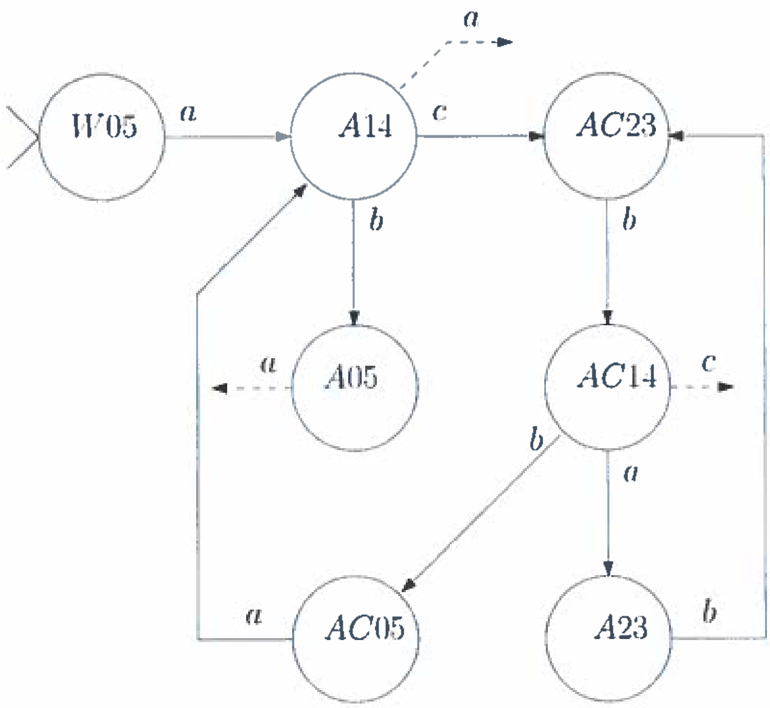

Figure 2.5 The transition diagram of $G_K || G_{min}$

yields the automaton shown in Fig. 2.6 Notice that in state $AC14$ the event c is disabled. Since c is an uncontrollable event the specification K is not controllable.

e) Removing AC14 from the previous automaton, and taking the Trim of what is left yields the automaton in Fig. 2.6. Notice that only a and b events are disabled, both controllable. Hence, this automaton implements a supervisor that generates $K^{\uparrow C}$. Notice that this is a blocking supervisor.

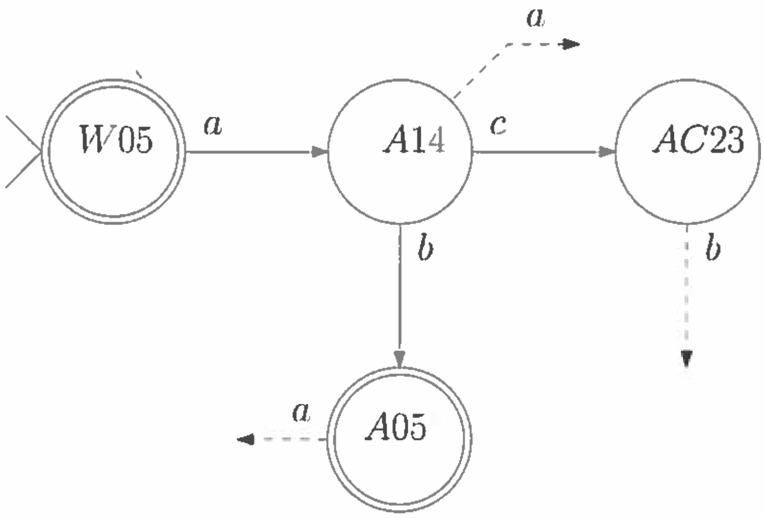

Figure 2.6 The transition diagram of the supervisor implementing the maximal controllable sublanguage

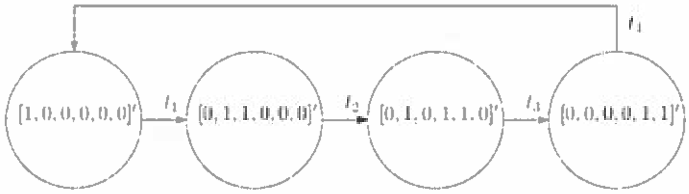

Figure 3.1 The reachable set $\mathscr{R}(M_0)$ and associated transition diagram

3. Exercise

a) The incidence matrix C is computed as follows:

$$C = \begin{bmatrix} -1 & 0 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix}.$$

b) There are 3 independent P-semiflows of minimal support: $[1,1,0,0,0,1]$,$[1,0,1,0,1,0]$ and $[1,0,1,1,0,1]$.

c) Notice that the union of the supports of the P-semiflows is $\{p_1, p_2, \ldots, p_6\}$, hence the network N is structurally bounded.

d) The transition diagram associated to the reachable set is shown in Fig. 3.1.

e) The structure of the concurrent composition is shown in Fig. 3.2.

f) The Petri net model does not distinguish which automaton has executed the transition, and in particular whether a token has been "processed" by the first or second automaton. In this respect its state space is at most of dimension 10 rather than 16. Therefore, it won't be isomorphic to the presvious parallel composition (10 being the number of unordered pairs of 4 elements).

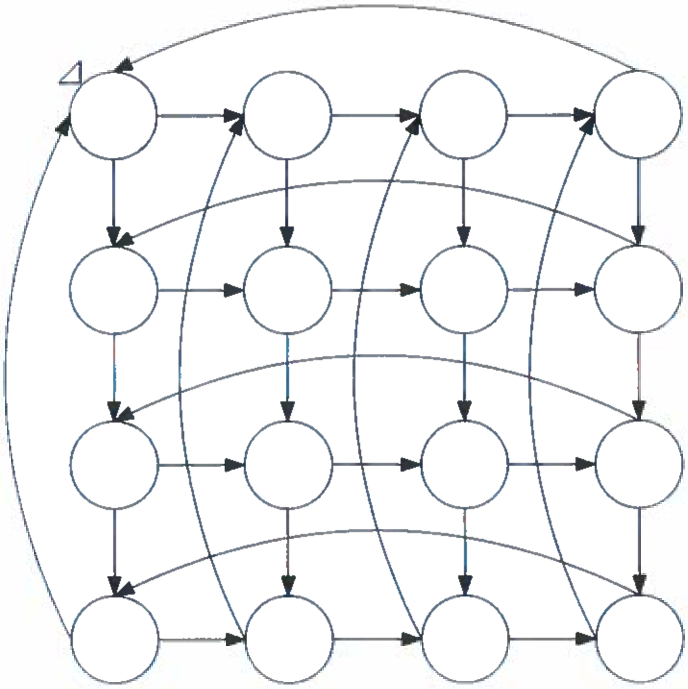

Figure 3.2 Structure of concurrent composition

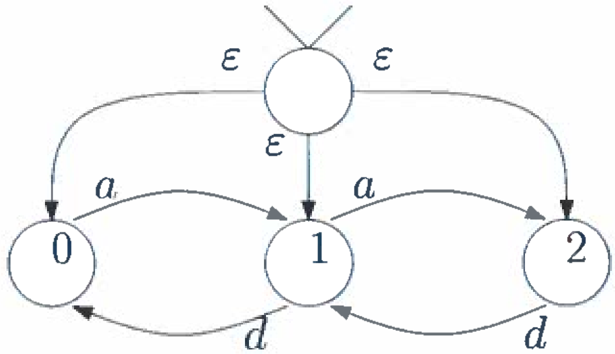

Figure 4.1 The non-deterministic automaton G_N

4. Exercise

a) In order to model an uncertain initial condition we may let ε events bring us from an auxiliary initial state into all other states. This is done in the Automaton G_N shown in the Fig. 4.1.

b) The observer automaton of G_N is shown in Fig. 4.2.

c) The transition diagram is the same of the observer automaton. Ordering states as $\{0,1,2\},\{1,2\},\{0,1\},\{2\},\{1\},\{0\}$ and considering that all state transitions occur with rate λ, the associated Markov chain fulfills the following equations:

$$\dot{\pi} = \pi Q, \quad Q = \begin{bmatrix} -2\lambda & \lambda & \lambda & 0 & 0 & 0 \\ 0 & -2\lambda & \lambda & \lambda & 0 & 0 \\ 0 & \lambda & -2\lambda & 0 & 0 & \lambda \\ 0 & 0 & 0 & -\lambda & \lambda & 0 \\ 0 & 0 & 0 & \lambda & -2\lambda & \lambda \\ 0 & 0 & 0 & 0 & \lambda & -\lambda \end{bmatrix}$$

d) The transition diagram has a unique absorbing strongly connected component.

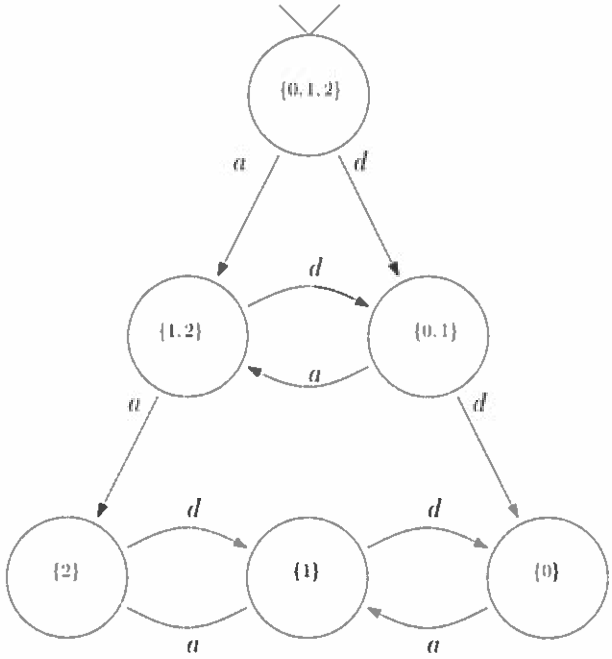

Figure 4.2 The observer of G_N

Moreover, each transition has a positive probability of occurring whenever enabled. Hence, with probability one the state will be eventually confined in the absorbing component. Notice that the absorbing component includes the states $\{2\}$, $\{1\}$ and $\{0\}$ only, where there is no uncertainty on the current state estimate.

e) Computing the average time for a correct estimate amounts to computing the average absorption time of the component $\{2\},\{1\},\{0\}$. This is done more easily first noticing that re-grouping the states as $x_a := \{0,1,2\}$, $x_b := \{\{0,1\},\{1,2\}\}$, and $x_c := \{\{0\},\{1\},\{2\}\}$ yields a reduced Markov chain (with 3 states only) of equations:

$$\dot{\pi}_a = -2\lambda\pi_a \qquad \dot{\pi}_b = 2\lambda\pi_a - \lambda\pi_b \qquad \dot{\pi}_c = \lambda\pi_b.$$

The average time for transition from state a to state b is $1/2\lambda$. While from transition from x_b to x_c is $1/\lambda$. Overall the average absorption time is:

$$\tau := \frac{1}{2\lambda} + \frac{1}{\lambda} = \frac{3}{2\lambda}.$$

f) This happens as long as the Automaton is minimal and each event has a strictly positive probability of occurring. In this case, there are no 'language' equivalent states and this indeed allows to discriminate exactly the current state on the basis of the sequence of events occurred.