Final

E4.57
ISE4.63
CS2.2

IMPERIAL COLLEGE LONDON

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING
EXAMINATIONS 2010

MSc and EEE/ISE PART IV: MEng and ACGI

DISCRETE-EVENT SYSTEMS

Friday, 7 May 10:00 am

Time allowed: 3:00 hours

There are SIX questions on this paper.

Answer FOUR questions.

All questions carry equal marks

Any special instructions for invigilators and information for candidates are on page 1.

Examiners responsible First Marker(s) : D. Angeli

Second Marker(s) : E.C. Kerrigan

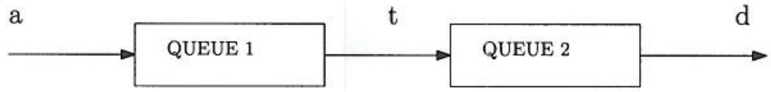

Figure 1.1 Queues in series

1. An office is organized with two queues that are in series with one another, namely departure events from the first queue are also arrival events for the second one. Let a, t, d denote respectively: arrival events at the first queue, transit events between queue 1 and queue 2, departure events from queue 2 (see Fig. 1.1).

   a) Design Finite Deterministic Automata G_1 and G_2 that model the two queues when considered separately; define the initial and terminal states to be the empty queue and assume that a maximum of 3 people are allowed in each queue. [6]

   b) Design a Finite Deterministic Automaton G that models the office as a whole. How is $L_m(G)$ related to $L_m(G_1)$ and $L_m(G_2)$? [6]

   c) Let $F = \{a, d\}$ be a subalphabet of $E := \{a, t, d\}$. Design a Finite Deterministic Automaton G_P such that $L_m(G_p) = P_F(L_m(G))$ (by $P_F(L)$ we denote projection of language L over the subalphabet F). Explain in your words the result obtained. (Hint: replace t events in G by ε events; this is a non-deterministic automaton....). [8]

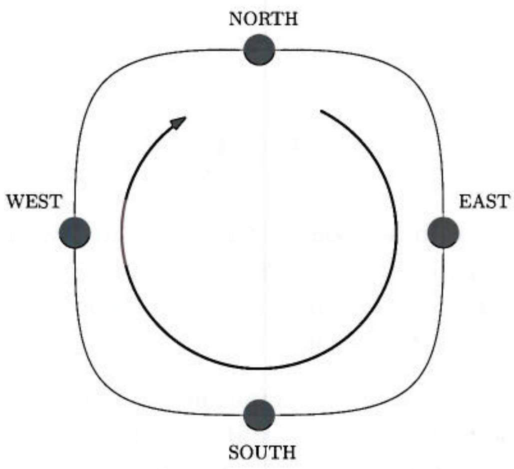

Figure 2.1 Metro line and stations

2. The town of Utopia has an underground system with a single circular line over which trains only run clockwise. There are 4 stations along the line, namely: North, East, South and West (see Fig. 2.1)

a) Assuming that the transit between two stations can be modeled as an atomic event, build a finite deterministic automaton that keeps track of the location of a single train along the circular line (use the alphabet $E = \{e_1, e_2, e_3, e_4\}$ for transitions between North to East, East to South, South to West and West to North respectively). [3]

b) To provide an efficient service, 2 trains (denoted by A and B) can simultaneously circulate in the line. Obviously, each train has a separate set of events, associated to it, namely $E_A = \{a_1, a_2, a_3, a_4\}$ and $E_B = \{b_1, b_2, b_3, b_4\}$. Assuming that train A is in station North and train B in station South at the initial time, build a unique deterministic automaton that keeps track of the position of the 2 trains. [5]

c) For safety reasons it is necessary to avoid that 2 trains are simultaneously at the same station. Assuming all events are controllable, design a supervisor that achieves this specification. [6]

d) Let K be the set of strings $w \in (E_A \cup E_B)^\star$ such that if last event of a type is a_i, only events of type b_j with $j \neq i$ are allowed. Moreover, if no event of type a has occured, then only b_1, b_2, b_3 are legal. Design an automaton G_K which implements the above specification. (Hint: an automaton with 4 states is enough. Each one stands for "a_i event is the last **a** event that has occured", $i = 1,2,3,4$). [3]

e) Assuming that a_i and b_i are controllable if i is odd, and uncontrollable otherwise. Let the initial state x_0 be such that train A is in station North and train B in station South. Is the specification K controllable with respect to $L(G)$? [3]

3. A robot (Robot 1) works on a cycle of 2 operations, using 2 different tools, namely tool A and tool B. Similarly, Robot 2 has a cycle of two operations using tools B and C. Each operation takes a variable amount of time and the probability distribution of the durations is given by a decreasing exponential with rates λ_A, λ_B and λ_C for operations involving tools A, B and C respectively.

a) Model each of the robots (separately) as a continuous-time Markov chain that keeps track of the operation currently being performed. What is the average duration of the cycles of Robot 1 and Robot 2 ? [6]

b) Assume next that only one tool of type B is available for the robots to use. The following policy is adopted: as soon as Robot i is using tool B Robot $(3-i)$ suspends his operation until tool B is released. Build a continuous-time Markov chain model that keeps track of both robots supervised according to the previous policy. [7]

c) What is the average duration of the cycles of Robots 1 and 2 when operated according to the policy described in paragraph b) ? [4]

d) What fraction of time is tool B being employed ? Justify your response. [3]

4. Consider a network of chemical reactions with the following structure:

$$\begin{aligned} A+B &\rightarrow C \\ C &\rightarrow A+B \\ C &\rightarrow B+D \\ D &\rightarrow A \end{aligned}$$

a) Build a Petri Net N that mimics the chemical network, by associating a place to each chemical compound and a transition to each reaction. In particular, tokens within each place represent the number of molecules of the corresponding chemical compound. [4]

b) Compute the incidence matrix of N [4]

c) Compute the P-semiflows of the network N. What is their "chemical" or "physical interpretation" ? Is the network structurally conservative ? [4]

d) Compute the T-semiflows of the network N. Can you think of a "chemical interpretation" of them ? [3]

e) Consider the initial marking $M_0 = [n_A, n_B, n_C, n_D]' = [2, 2, 0, 0]'$. Compute the reachable set $\mathscr{R}(M_0)$ and the associated transition graph. [3]

f) A marked Petri net $\{N, M_0\}$ is said *reversible* if for each $M \in \mathscr{R}(M_0)$ there exists $w \in T^\star$ such that $M[w\rangle M_0$. Is N reversible for M_0 as specified ? [2]

5. A factory for the production of coloured glasses has its terminal production stage organized as follows: normally a rough glass is first processed by machine A, which smoothens the piece, then by machine B, which is responsible for coloring it. If the result is satisfactory, the piece is shipped away, otherwise it is sent back to machine A. Both machines can have at most 2 pieces waiting to be processed.

a) Design finite deterministic automata G_A and G_B for Machines A and B, respectively; use the following event symbols: a for arrivals of new pieces at Machine A, f for arrivals of faulty pieces from Machine B, t for pieces in transit from Machine A to Machine B, d for departures of satisfactory glasses from Machine B. [6]

b) Compute $G = G_A || G_B$, viz. the concurrent composition of the Automata. [6]

c) Assuming that only a and t events are observable, design a finite deterministic automaton for the diagnosis of f events. [8]

6. The transmission protocol over a dedicated serial line is structured as follows:

- bits are transmitted one at a time in packets of 4;
- The fifth bit transmitted is 0 if the number of 1s in the 4-bit packet is even, 1 if the number is odd.

a) Build a finite deterministic automaton G_R that models a receiver operating as follows: the receiver starts in a state in which no bits have been transmitted. Then it counts the numbers of ones and zeros received (as well as the total number of bits), up to a maximum of 4 bits; if the fifth bit received agrees with the parity rule it goes back to the initial state, otherwise it enters an error state. The initial state is also the only terminal state. [7]

b) Is the constructed automaton G_R minimal ? Can you derive a minimal automaton G_M performing the same task ? [7]

c) Assuming that the probability of receiving a wrong bit is α (for any transmitted bit, including the parity bit), and that the transmission errors are independent of one another, can you determine the probability of entering the error state for a single packet of 4 bits + the parity bit being transmitted ? (Hint: how many wrongly transmitted bits does it take to enter the error state ? Use a discrete time Markov chain to evaluate the probability of having an even or an odd number of transmission errors over k transmitted bits). [6]

E4.57
ISE 4.63
CS2.2

SOLUTIONS ~~COURSEWORK~~: DISCRETE EVENT SYSTEMS MASTER IN CONTROL

2010

1. Exercise

a) Automata G_1 and G_2 have event sets $E_1 = \{a,t\}$ and $E_2 = \{t,d\}$ respectively. Each of them has 4 states, $X_1 = X_2 = \{x_0,x_1,x_2,x_3\}$, where x_i corresponds to i clients in the queue. The graphical representations of G_1 and G_2 are shown in Fig. 1.1.

b) In order to compute G it is enough to take the concurrent composition of G_1 and G_2, $G = G_1||G_2$. This yields the automaton shown in Fig. 1.2. Clearly $L_m(G) = L_m(G_1)||L_m(G_2)$.

c) Replacing t in G by ε events yields a Finite Nondeterministic Automaton, G_N which marks the language $P_F(L_m(G))$ (indeed any word w is marked by G iff it is also marked as $P_F(w)$ by G_N). Hence, we may transform G_N to an equivalent deterministic automaton $G_P := Obs(G_N)$. The resulting automaton is displayed in Fig. 1.3. As it can be noticed this is the automaton of a queue with arrivals and departures events and maximum number of clients queueing equal to 6 (that is the sum of the capacities of queues 1 and 2).

2. Exercise

a) A single train can be modeled by 4 states; each state represent a station. We denote $X = \{N,E,S,W\}$, which stand for North, East, South and West respectively. The alphabet E is defined as $E = \{a_1,a_2,a_3,a_4\}$. Each event denotes the transition from one station to the next in a clockwise direction as shown in Fig. 2.1.

b) Let G_A and G_B denote the automata associated to train A and train B respectively. Denote by $E_A = \{a_1,a_2,a_3,a_4\}$ and $E_B = \{b_1,b_2,b_3,b_4\}$. The automaton describing the two trains running simultaneously is obtained by taking the concurrent composition $G := G_A||G_B$. This yields the automaton in Fig. 2.2.

c) The forbidden states are those with train A and B at the same station, namely: $\{NN,EE,SS,WW\}$. A supervisor G_S is obtained by removing those states from

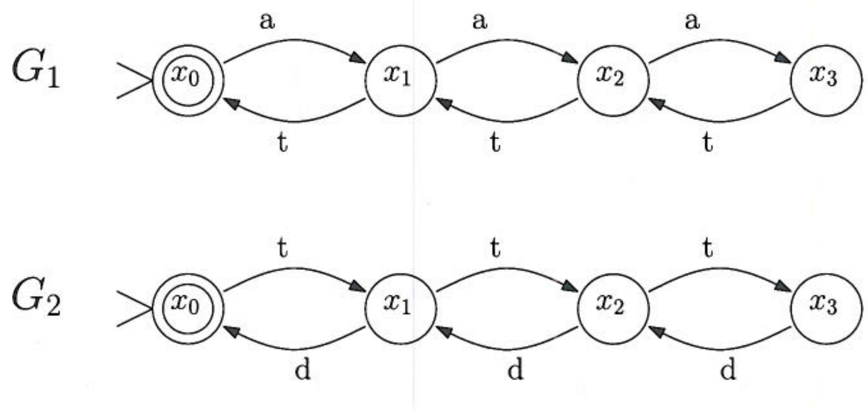

Figure 1.1 Automata G_1 and G_2 (Exercise 1)

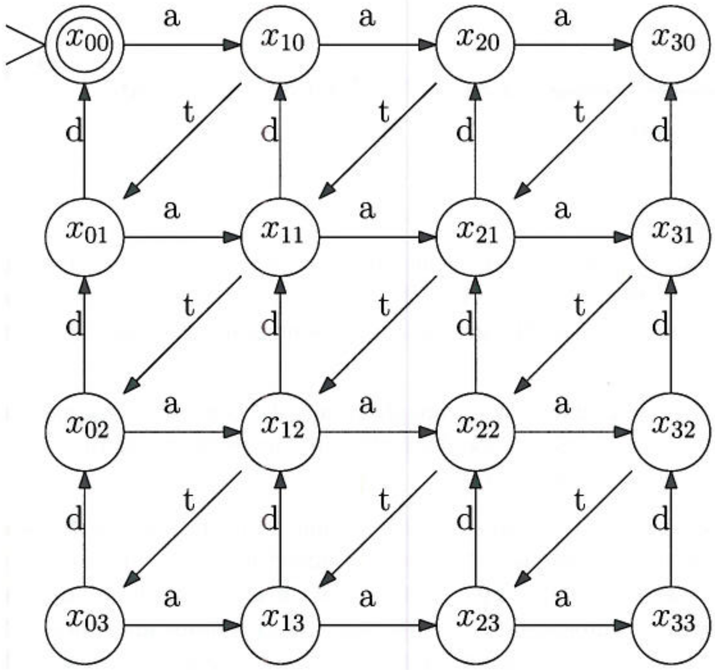

Figure 1.2 Automaton $G = G_1||G_2$ (Exercise 1)

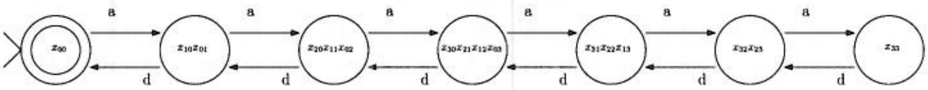

Figure 1.3 Automaton G_P (Exercise 1)

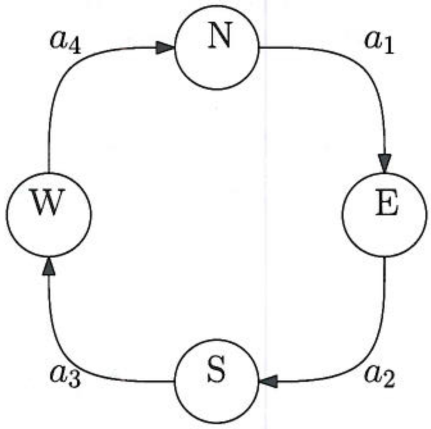

Figure 2.1 Automaton modeling a single train (Exercise 2)

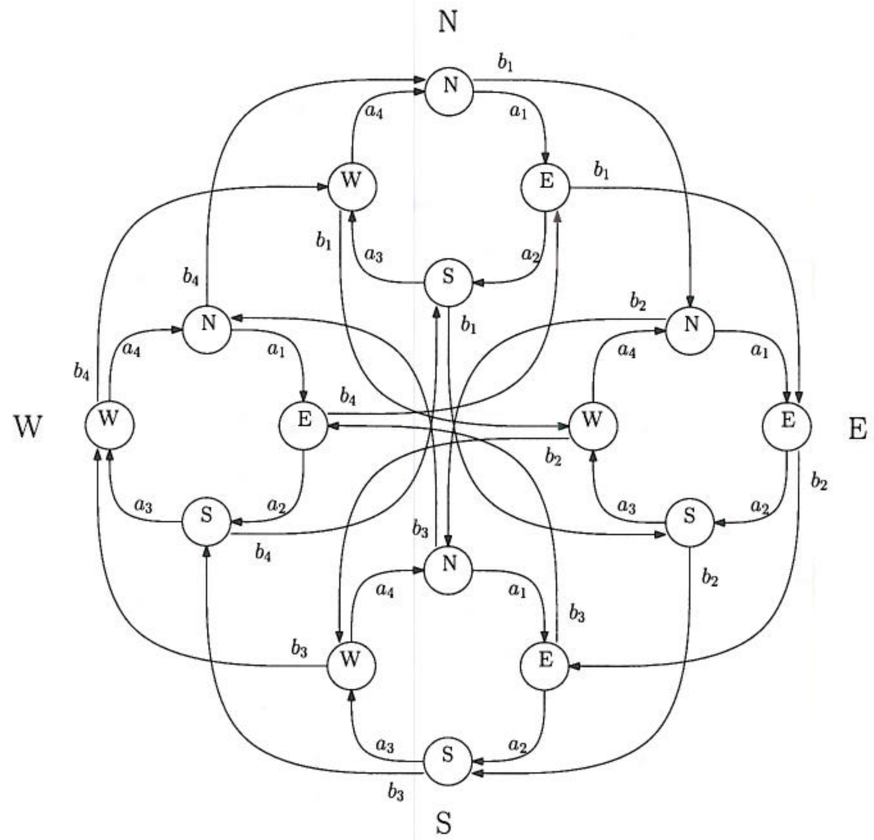

Figure 2.2 Automaton modeling two trains (Exercise 2)

G, as well as the arcs incident to them and letting the set of enabled events $f(w) := \mathscr{A}_{G_S}(\delta^\star(x_0, w))$ with $\mathscr{A}_{G_S}(x)$ denoting the set of active events in state x (for the automaton G_S) and $\delta^\star$ being the transition map of automaton G_S. This is possible since all events are controllable. See Fig. 2.3. Clearly, the initial state x_0 needs to be one of the allowed states, for instance $x_0 = NS$.

d) The automaton G_K is shown in Fig. 2.4.

e) The specification K is uncontrollable. Indeed, denoting $E_{uc} = \{a_2, a_4, b_2, b_4\}$, the string b_3b_4 is not in K, but it is in KE_{uc} and in $L(G)$.

3. Exercise

a) Robot 1 evolves between 2 states, namely state A and state B with probability rates of transition as specified in Fig. 3.1. Similarly for Robot 2, which evolves between state B and C. The average cycle lengths of the 2 robots are simply given by the sum of the average permanence in each of the two states, namely $\frac{1}{\lambda_A} + \frac{1}{\lambda_B}$ for Robot 1 and $\frac{1}{\lambda_B} + \frac{1}{\lambda_C}$ for Robot 2.

b) The Markov chain modeling the two robots working in parallel and sharing tool B is shown in Fig. 3.2. State AB stands for Robot 1 using tool A and Robot 2 using tool B. Analog notation is adopted for the other states. Notice that the state BB is indeed not present. Equations for the above Markov chain read as follow:

$$\begin{aligned}
\dot{\pi}_1 &= -\lambda_B \pi_1 + \lambda_C \pi_2 \\
\dot{\pi}_2 &= \lambda_B \pi_1 - (\lambda_C + \lambda_A)\pi_2 + \lambda_B \pi_3 \\
\dot{\pi}_3 &= \lambda_A \pi_2 - \lambda_B \pi_3
\end{aligned}$$

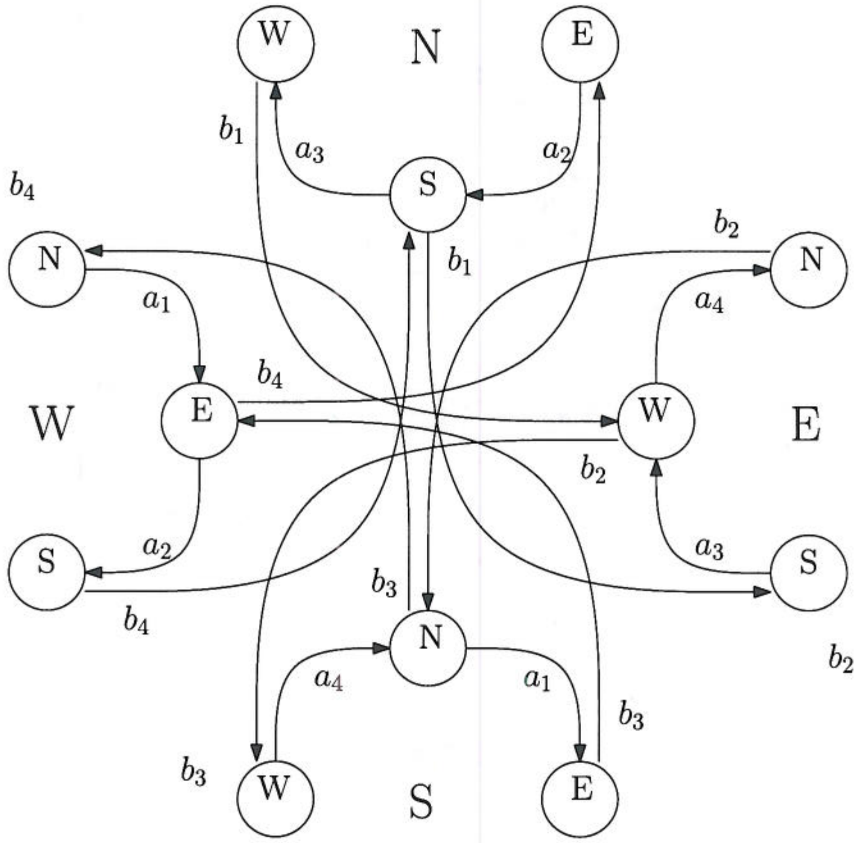

Figure 2.3 Supervisor for the 2 trains system (Exercise 2)

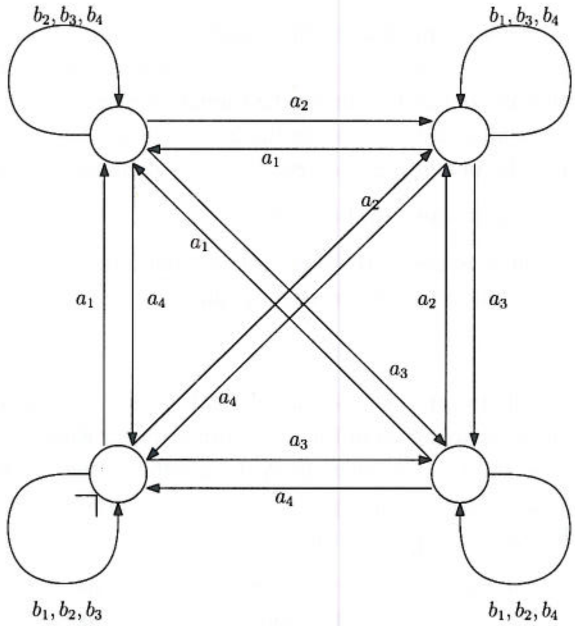

Figure 2.4 Automaton G_K (Exercise 2)

Figure 3.1 Graphical representation of Markov chains for Robot 1 and 2 (Exercise 3)

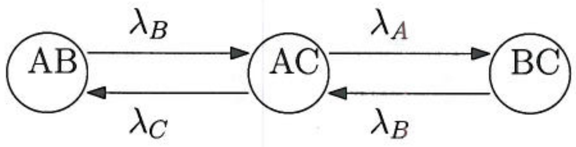

Figure 3.2 Robots sharing tool B (Exercise 3)

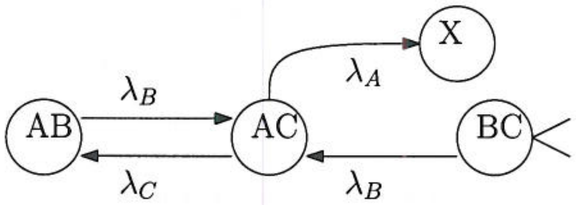

Figure 3.3 Robot 1 cycle as an absorption time of modified Markov chain (Exercise 3)

c) In order to compute the average cycle length of machine A, we may compute the absorption time of state X in the Markov chain shown in Fig. 3.3, when initial state is BC. This is given by

$$\frac{1}{\lambda_B} + \frac{\lambda_C + \lambda_B}{\lambda_B \lambda_A}$$

d) Notice that the Markov chain in Fig. 3.2 is ergodic (being irreducible the associated graph). Therefore, the fraction of time in which tool B is used is exactly the sum of the steady state probabilities of states AB and BC. Computation yields:

$$\frac{\lambda_A + \lambda_C}{\lambda_A + \lambda_B + \lambda_C}$$

4. Exercise

a) The Petri Net modeling the reaction network described is obtained by connecting to each transition the chemical compounds to the left-hand side of the reaction (also called reactants) as input places and the chemical compounds to the right-hand side (also called products) as output places. The result is shown in Fig. 4.1.

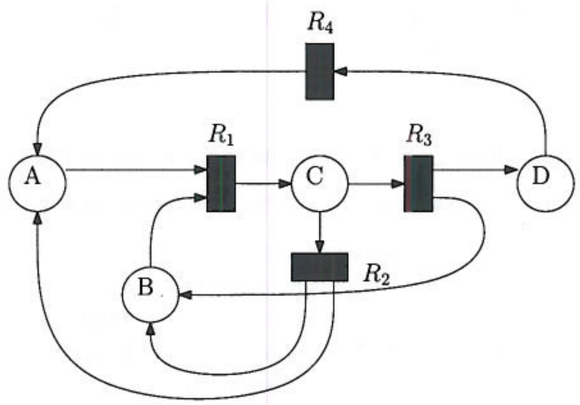

Figure 4.1 Petri Net associated to the given chemical network (Exercise 4)

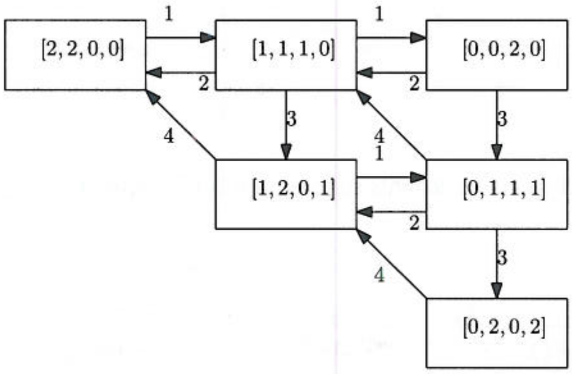

Figure 4.2 Reachable set, transition diagram (Exercise 4)

b) Its incidence matrix C, ordering reactions as listed in the Exercise, and places in alphabetical order, reads:

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

c) The minimal P-semiflows of minimal support of N are given by:

$$[1,0,1,1],[0,1,1,0]$$

These are the so called conserved moietys of the chemical reaction. In particular, denoting n_X the number of molecule of X available at any given time, we have that $n_A + n_C + n_D$ is constant in time, as is $n_B + n_C$. The network is structurally conservative as $[1,1,2,1]$ is a P-semiflow whose support is $\{A,B,C,D\}$.

d) The T-semiflows of N are given by:

$$[1,1,0,0]',[1,0,1,1]'$$

These can be interpreted as reaction rates which keep chemical concentrations unaltered.

e) The reachable set is finite and shown in Fig. 4.2.

f) The network N,M_0 is reversible, since the transition diagram of the Reachable set is an irreducible graph.

5. Exercise

a) The automata modelling Machines A and B are shown in Fig. 5.1.

b) The concurrent composition $G_A||G_B$ is the automaton shown in Fig. 5.2.

c) In order to implement a diagnoser, let first compose G with the automaton shown in Fig. 5.3. This yields the Automaton G_D in Fig. 5.4. A non-deterministic automaton G_N is obtained from G_D simply by replacing d and f events by ε events. Finally, the diagnoser can be designed by computing the observer automaton of G_N. This is shown in Fig. 5.5.

6. Exercise

a) The automaton modelling the receiver is illustrated in Fig. 6.1.

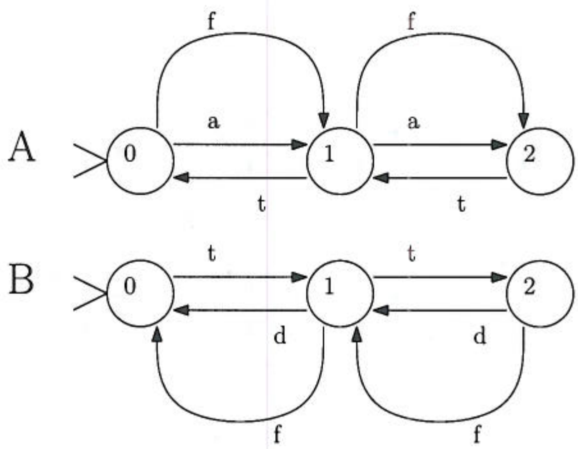

Figure 5.1 Automata G_A and G_B (Exercise 5)

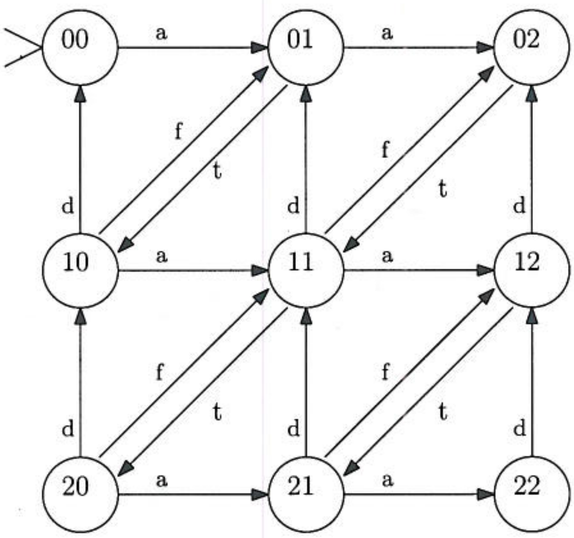

Figure 5.2 Automaton G (Exercise 5)

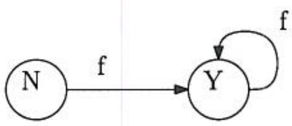

Figure 5.3 Auxiliary automaton for diagnosis (Exercise 5)

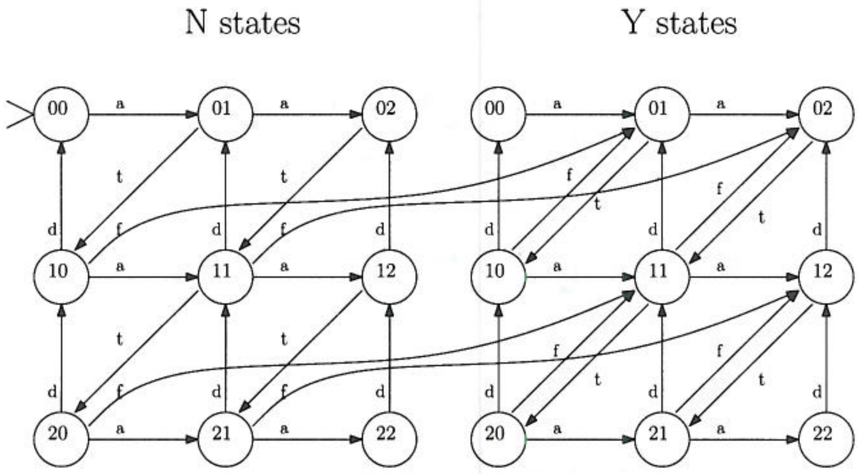

Figure 5.4 Composition of auxiliary automaton with G (Exercise 5)

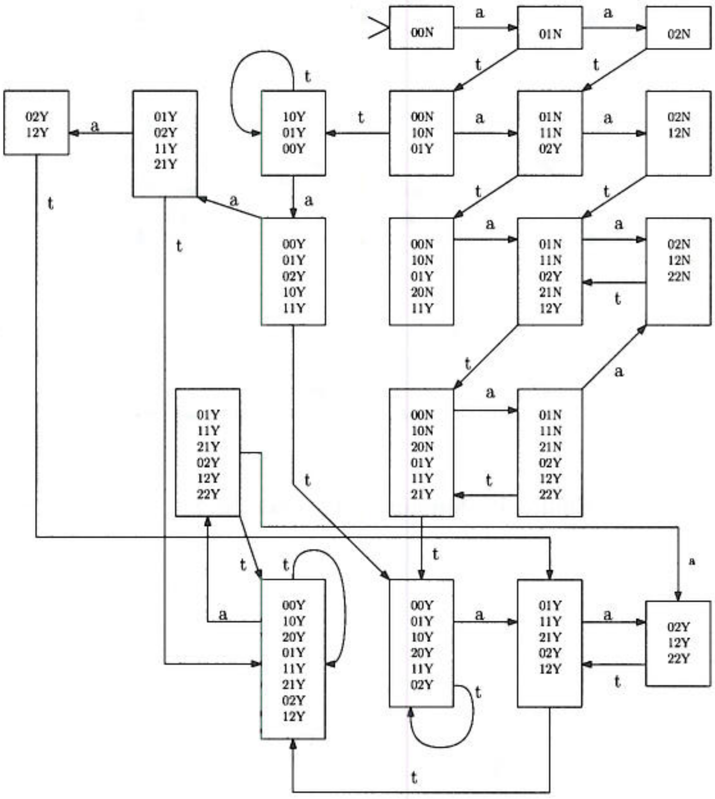

Figure 5.5 Diagnoser automaton (Exercise 5)

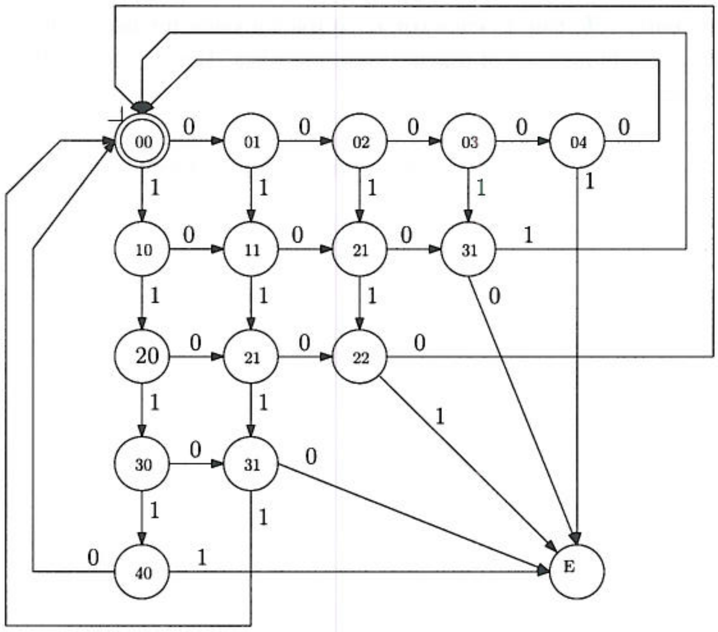

Figure 6.1 Receiver automaton G_R (Exercise 6)

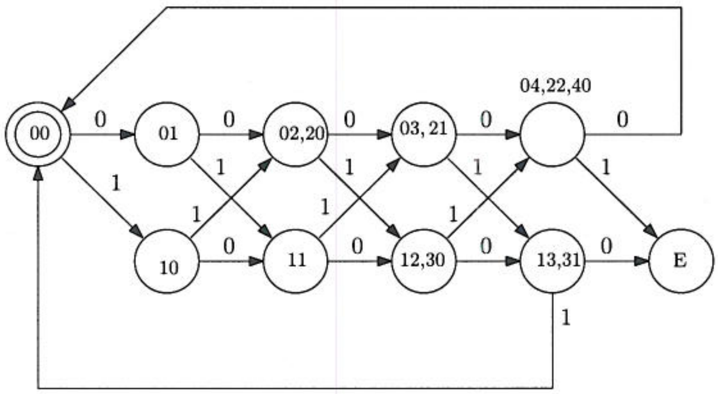

Figure 6.2 Minimal receiver automaton G_M (Exercise 6)

b) The minimal automaton G_M which marks and generates the same language is shown in Fig. 6.2. Notice that the minimal automaton only keeps track of the total number of bit received and the parity of the number of ones received (rather than their exact number).

c) The error state is entered whenever an odd number of transmission errors occur over the 5 bits sequence. Hence, a simple discrete-time Markov chain can be used in order to keep track of the probability of entering the error state after 5 transmitted bits. In particular, denoting P the matrix:

$$P = \begin{bmatrix} 1-\alpha & \alpha \\ \alpha & 1-\alpha \end{bmatrix}$$

we may compute the probability of making an even or an odd number of transmission errors after k bits transmitted by evaluating

$$P^k \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$

the first entry referring to even (or zero) transmission mistakes and the second referring to odd number of transmission mistakes. Hence, the sought probability is given by

$$[0,1]\,P^5 \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 5\alpha - 20\alpha^2 + 40\alpha^3 - 40\alpha^4 + 16\alpha^5$$