IMPERIAL COLLEGE LONDON

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING
EXAMINATIONS 2011

MSc and EEE/ISE PART IV: MEng and ACGI

DISCRETE-EVENT SYSTEMS

Friday, 13 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

1. A small factory operates with 2 robots, R_1 and R_2. Each robot is dedicated to the production of a specific part of an artifact, and may use one of two tools which are available, T_1 and T_2. Let u_{ij} denote the event Robot R_j uses tool T_i; similarly, let r_{ij} denote the event Robot R_j releases tool T_i.

a) Model tools T_i, $i = \{1,2\}$ by means of finite deterministic automata. These should have states which also take into account the possibility of a tool being asked by more than one robot (underflow), or being released by more than one robot (overflow). [3]

b) Robot R_1 has an operation cycle which entails a sequence of 2 stages, the first using tool T_1 and the second using tool T_2; after them the event p_1 may take place and the cycle restarts. Similarly, robot R_2 has an operation cycle which entails a sequence of two lavorations, the first using tool T_2 and the second using tool T_1. Following these the event p_2 may take place and the cycle may restart. Build finite deterministic automata R_1 and R_2 which model the operation cycles of the two robots. [3]

c) Build the automaton R_{12} corresponding to the concurrent composition $R_1||R_2$ (make sure that its states are displayed graphically in a suitably organized grid). [4]

d) Build next the automaton $O = R_{12}||T_1||T_2$. This represents the open-loop behaviour of our factory (Hint: try to use a similar grid to that of R_{12}). [4]

e) Assume next that the controllable events are $E_c = \{u_{11}, u_{12}, u_{21}, u_{22}\}$ while the rest are uncontrollable. Build an automaton H which models the following specification: each tool can only be used by one robot and be released only after it has been in use. [3]

f) Build the concurrent composition $O||H$ and see if the specification $\mathscr{L}(H)$ is controllable with respect to $\mathscr{L}(O)$ and the set of uncontrollable events $E_{uc} = \{r_{11}, r_{12}, r_{21}, r_{22}, p_1, p_2\}$. [3]

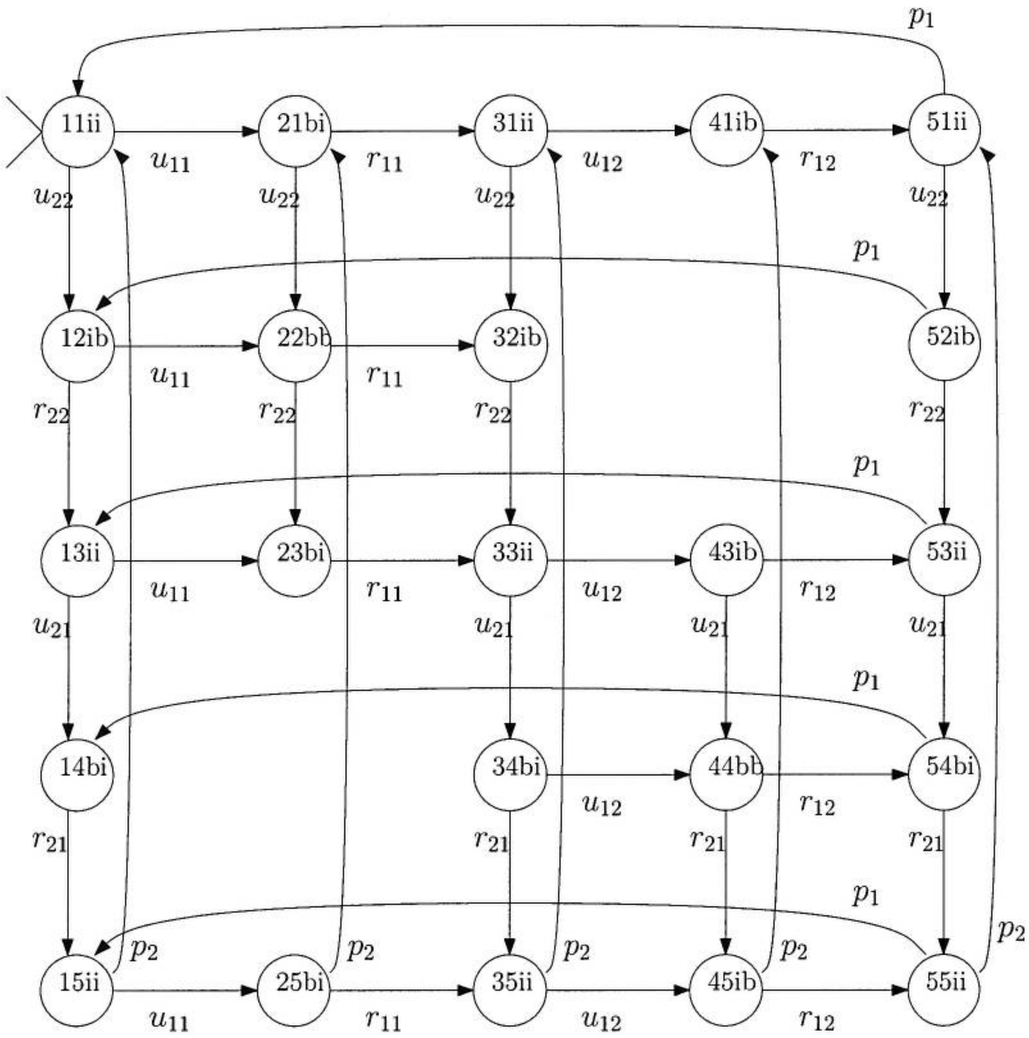

Figure 2.1 Supervised factory G

2. A factory as in Exercise 1 is operated according to the Automaton G shown in Fig. 2.1. Assume that a new stage of production is added, in order to assemble pieces of type p_1 and p_2 in a final product. The buffer between robots R_1 and R_2 and the final assembling machine is limited to at most one piece either of type p_1 or of type p_2. In other words, whenever the buffer is empty and event p_i occurs (arrival of piece p_i) it must be the case that p_{3-i} occurs before p_i can occur a second time; furthermore, the occurence of p_{3-i} leaves the buffer empty again (as assembling of the two pieces can occur).

a) Model this specification as an automaton H with events p_1 (arrival of piece p_1) and p_2 (arrival of piece p_2). [4]

b) Let K be defined as follows: $K = \mathscr{L}(G)||\mathscr{L}(H)$. Is K controllable with respect to $\mathscr{L}(G)$ assuming $E_c = \{u_{11}, u_{12}, u_{21}, u_{22}\}$. Justify your answer. [6]

c) Sketch the automaton which generates the supremal controllable sublanguage $K^{\uparrow C}$. [10]

3. The train system of DES-Town is organized as a singular circular line which joins 3 stations, **E**ast, **W**est and **S**outh, respectively. Two distinct trains are running along the line. Each station is equipped with 2 platforms. Train A, which is cheaper and normally slower, always uses Platform 1, and an event a is produced each time the train moves from one station to the next. Train B, instead, may decide when entering a station to do it by visiting Platform 2 provided it was in Platform 1 at the previous station (in which case an event x is produced) or to visit Platform 1 (in which case an event b is produced).

a) Build finite deterministic automata G_A and G_B that keep track of Train A as it moves along the line, and of Train B as it moves along the line and possibly switches to a different platform. Assume that Train A starts at station West Platform 1, and Train B in station West Platform 2. [5]

b) Build the parallel composition $G_A || G_B$. [5]

c) Assume next that we would like to design a supervisor which forbids two trains to land in the same platform at the same station. Which is the minimal set of events to make this specification controllable with respect to the language $\mathscr{L}(G_A || G_B)$? [5]

d) Assume that a supervisor has been designed to fulfill the previous specification and let x events be unobservable. Design an observer for the non-deterministic automaton obtained by replacing x events by ε events. [5]

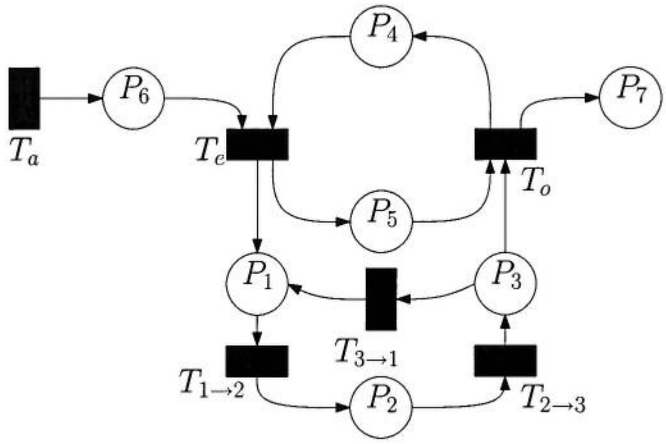

Figure 4.1 Petri Net N

4. A museum is organized over 3 rooms. Entrance is from Room 1, while Exit is from Room 3. It is possible to visit the rooms only in sequence, Room 1, 2 and 3, but also to go back from Room 3 to Room 1. This can be modelled via the Petri Net N shown in Fig. 4.1. In particular, places P_1, P_2 and P_3 represent the number of visitors in rooms 1, 2 and 3, respectively. People queueing at the entrance of the museum are accounted for in place P_6, while visitors who have already left the museum are in place P_7.

a) Let transitions be ordered as follows: T_a, T_e, $T_{1\to2}$, $T_{2\to3}$, $T_{3\to1}$, T_o. Compute the incidence matrix of the Petri Net N; [4]

b) What do transitions T_e, T_o and $T_{i\to j}$ model ? [2]

c) What is the use of places P_4 and P_5, assuming an initial marking $M_0 = [0,0,0,n,0,0,0]'$? [4]

d) Compute the P-invariant vectors of N; could the result be expected ? [4]

e) Are there P-decreasing vectors in N (besides the P-invariants) ? Could this be expected ? [2]

f) Remove Transition T_a from the net; compute the reachable set from initial marking: $[0,0,0,1,0,2,0]'$. [4]

5. In order to go from K to W (Kensington to Wenstminster) a commuter needs to take two buses. Either bus A and then bus B, or bus C and then bus D. The buses take different roads to reach the destination, so it is impossible to take bus A and then D or bus C and then B. Assume that buses come at random times according to clocks with exponential probability distributions of rates λ_A, λ_B, λ_C and λ_D, respectively and that journey time between the stops is negligible compared to the waiting time.

a) Model the travel of the commuter by means of a continuous time Markov chain. [3]

b) Find the average travel time if the commuter always takes the first bus coming (in particular between A and C). [5]

c) Under what conditions on the λ parameters, this policy is quicker (on average) than always waiting for bus A and then B ? For which values of λs it is quicker than always waiting for bus C and then D ? [4]

d) Assume that also on the way back the buses take the same road and come with the same rates, respectively λ_A, λ_B, λ_C and λ_D. Assume that a controller jumps in and out always at the 4 stops considered in this question and always takes the first bus coming (regardless of its direction). How should the Markov chain be modified to model this behaviour ? [2]

e) Is this an ergodic Markov chain ? Why ? [2]

f) What is the average percent of time that the controller spends in the 'Westminster' stop assuming $\lambda_A = \lambda_B = 1$ and $\lambda_C = \lambda_D = 2$? [4]

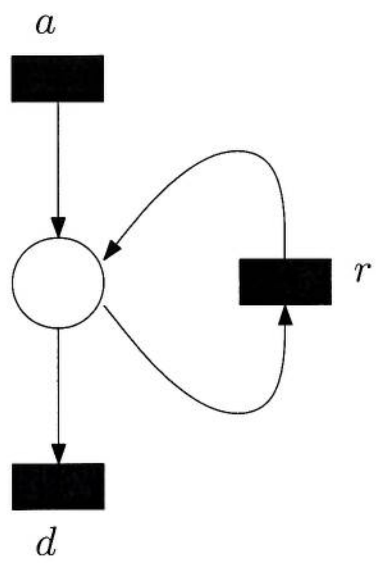

Figure 6.1 Petri Net N

6. Consider the Petri Net N shown in Fig. 6.1, with transitions $T = \{a, d, r\}$. Let its initial marking be $M_0 = [0]$. The generated language is defined as follows:

$$\mathscr{L} = \{w \in T^\star : M_0[w >\}.$$

a) Sketch in a graph the reachable set as well as the transitions between different reachable markings; [4]

b) Describe in set-theoretic terms (more specific than what used for the definition) the generated language $\mathscr{L}$; [4]

c) Can this language be generated by a Finite Deterministic Automaton ? (justify your response); [4]

d) Assume that the automaton sketched in item a) is a Deterministic Timed automaton with constant life-times given respectively as: $\theta_a = 3$, $\theta_d = 2$, $\theta_r = 7$. Sketch the time evolution of the automaton assuming clocks are updated according to the so called 'activation memory' algorithm. [4]

e) Sketch the time evolution of the automaton assuming the so called 'total memory' algorithm for the update of clocks. [4]

SOLUTIONS COURSEWORK: DISCRETE EVENT SYSTEMS 2011
MASTER IN CONTROL

1. Exercise

a) According to the definition of events u_{11},u_{12}, r_{11} and r_{12}, the automaton T_1 has the structure shown in Fig. 1.1. The 4 states $\{i, b, u, o\}$ correspond to **i**dle, **b**usy, **u**nderflow and **o**verflow conditions. A similar automaton models T_2.

b) The operation cycle of robots R_1 and R_2 is shown in Fig. 1.2.

c) The parallel composition of robots R_1 and R_2 can be displayed as a grid of 5×5 states, as shown in Fig. 1.3.

d) The automaton O resulting from the parallel composition $O = R_{12}||T_1||T_2$ has fewer states than R_{12}, and can be represented by using a similar grid. This is shown in Fig. 1.4.

e) The specification H can be implemented by modeling T_1 and T_2 without overflows and underflows states (and taking their parallel composition). See Fig. 1.5.

f) The closed-loop behaviour is represented in Fig. 1.6. Notice that only controllable events have been disabled by H. Hence the specification $\mathscr{L}(H)$ is controllable with respect to $\mathscr{L}(O)$.

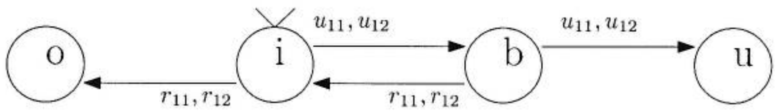

Figure 1.1 Finite Deterministic Automaton T_1

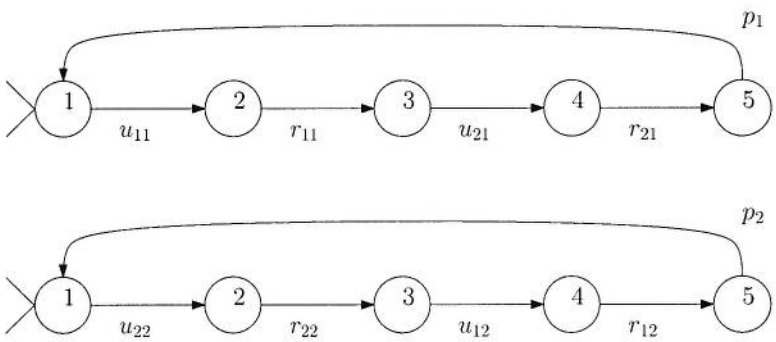

Figure 1.2 Finite Deterministic Automata for R_1 and R_2

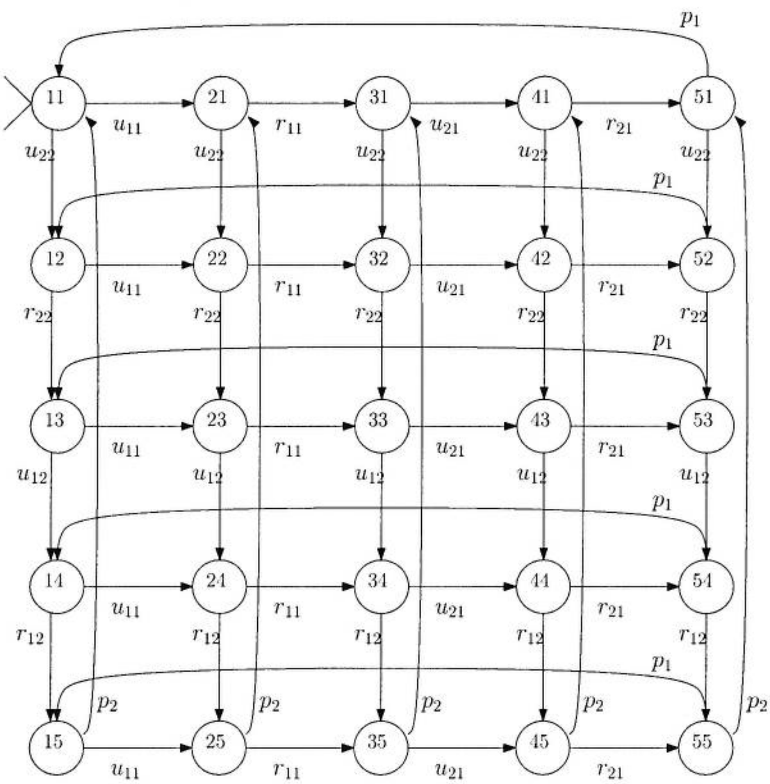

Figure 1.3 Finite Deterministic Automaton for $R_1||R_2$

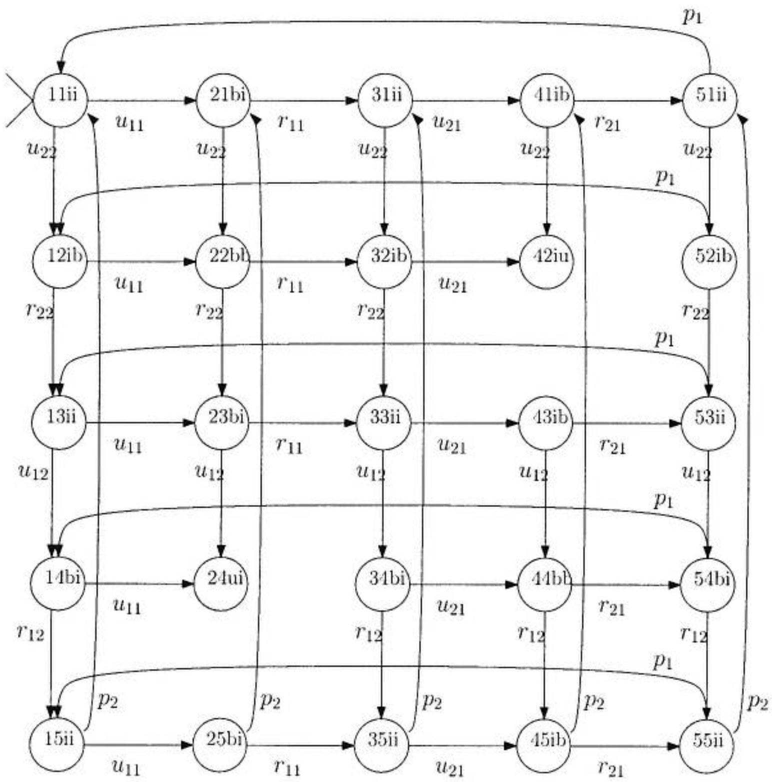

Figure 1.4 Finite Deterministic Automaton for $R_{12}||T_1||T_2$

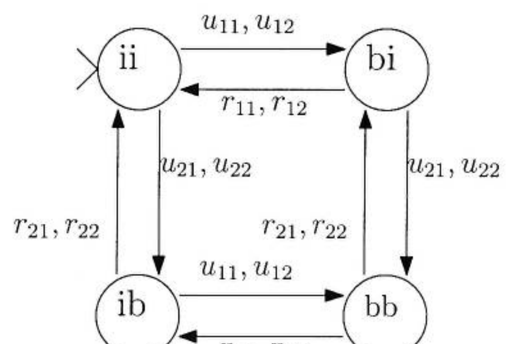

Figure 1.5 Finite Deterministic Automaton H

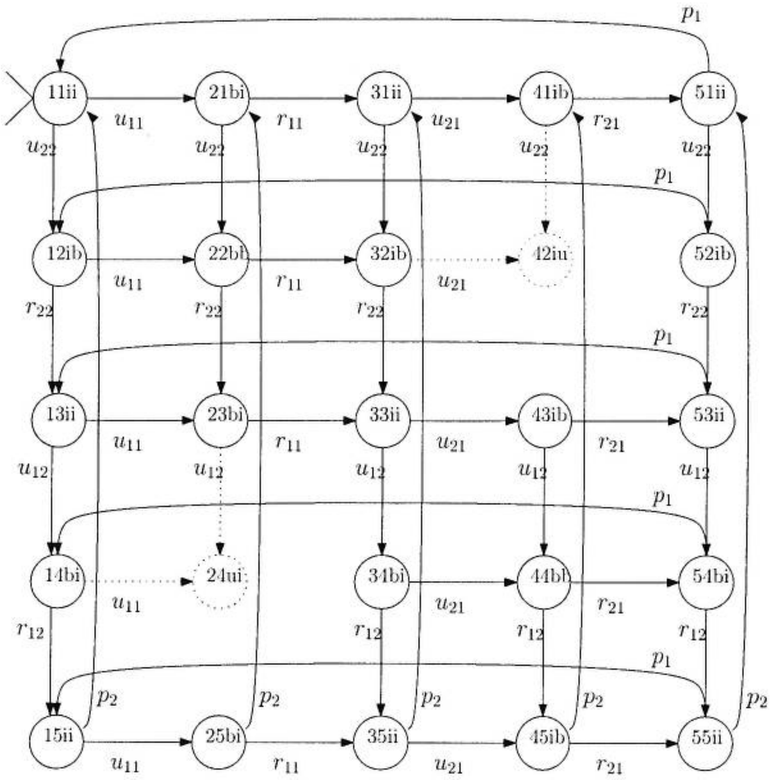

Figure 1.6 Finite Deterministic Automaton for $O||H$

2. Exercise

a) The automaton H is shown in Fig. 2.1.

b) The parallel composition $G||H$ yields the automaton shown in Fig. 2.2. Then, it is possible to verify that $\mathscr{L}(H)$ is uncontrollable with respect to E_{uc}. Indeed p_1 events are disabled in the rightmost states in Fig. 2.2. Similarly p_2 events are disabled in the bottom states of the same Figure. As these events were enabled in the corresponding states of G, and being p_1 and p_2 uncontrollable events, we may conclude that the language $\mathscr{L}(H)$ is uncontrollable.

c) In order to obtain the automaton that generates the supremal controllable sub-language we may refine the automaton shown in Fig. 2.2 by taking out states in which uncontrollable events are disabled together with their ingoing and out-

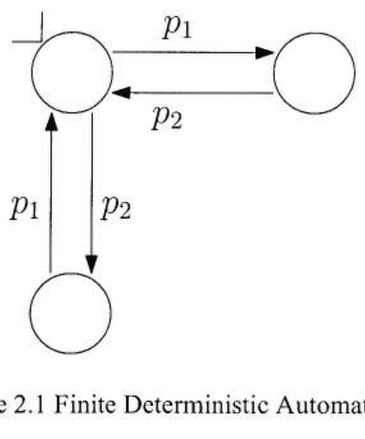

Figure 2.1 Finite Deterministic Automaton H

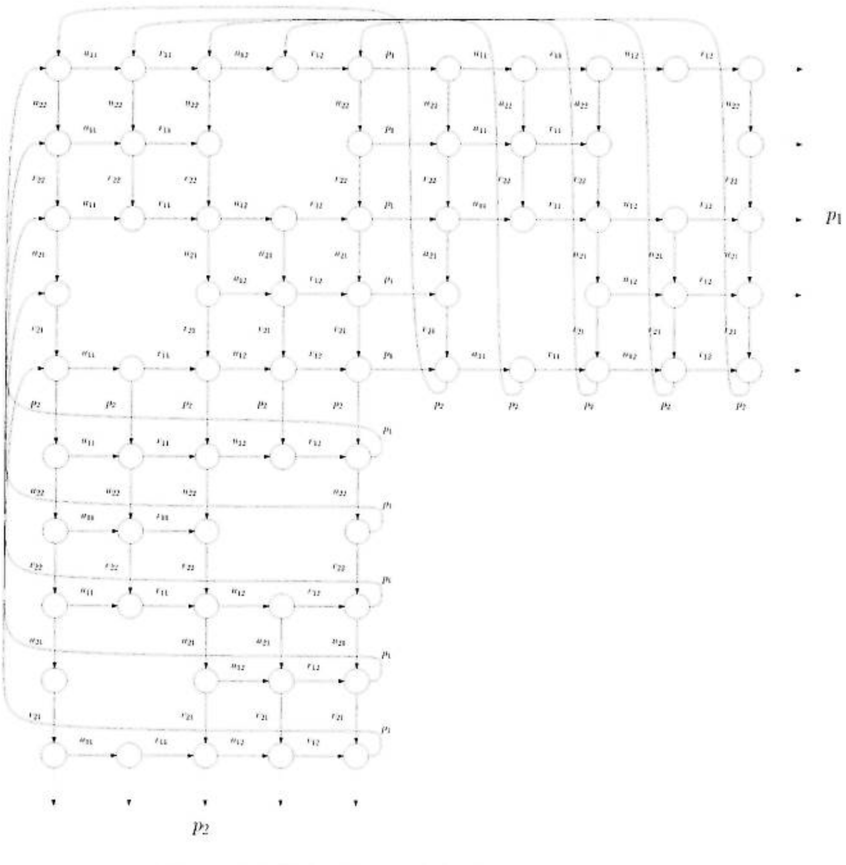

Figure 2.2 Finite Deterministic Automaton $G||H$

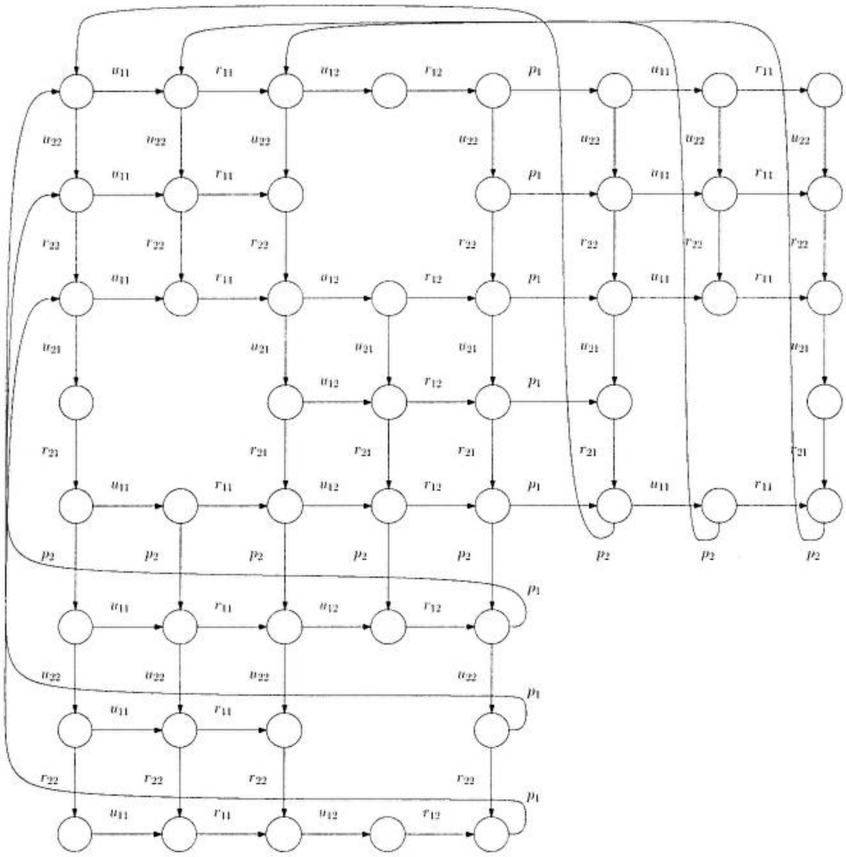

Figure 2.3 Finite Deterministic Automaton generating $K^{\uparrow C}$

going edges. Repeating the whole procedure until no more states are removed yields the desired automaton, which is shown in Fig. 2.3.

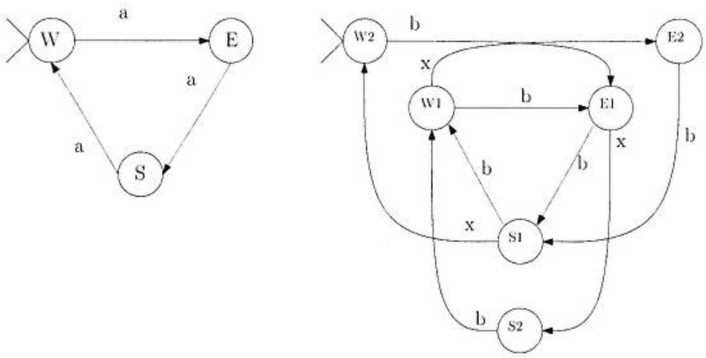

Figure 3.1 Finite Deterministic Automata G_A and G_B

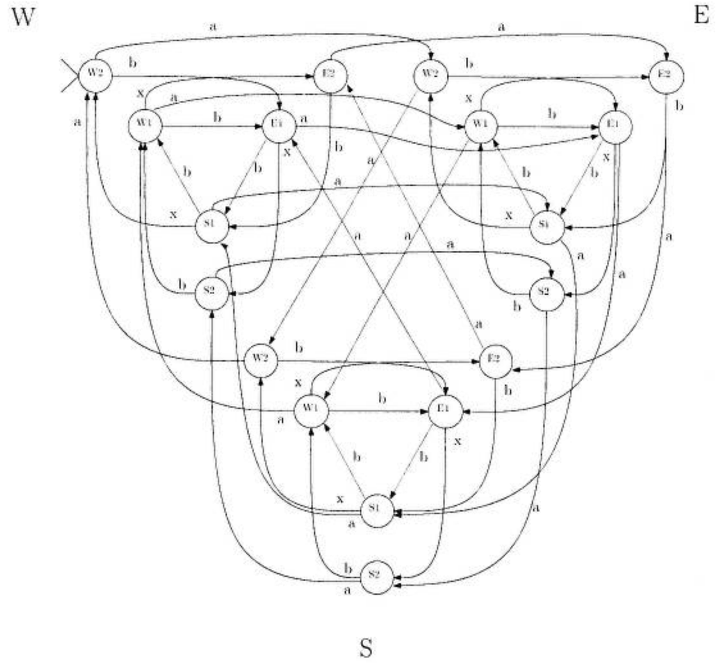

Figure 3.2 Finite Deterministic Automaton $G_A||G_B$

3. Exercise

a) The automata G_A and G_B are shown in Fig. 3.1.

b) The concurrent composition of G_A and G_B is shown in Fig. 3.2.

c) The supervised railway has the closed-loop language $\mathscr{L}(S/(G_A||G_B))$ generated by the automaton in Fig. 3.3. Notice that forbidden states $\{WW_1, EE_1, SS_1\}$ have been taken out from the Automaton $G_A||G_B$, as well as incoming and outgoing edges. The events that need to be controllable in order for this to be an admissible supervisory control action are the labels of ingoing arcs at removed states, that is a and b.

d) The requested observer is shown in Fig. 3.4.

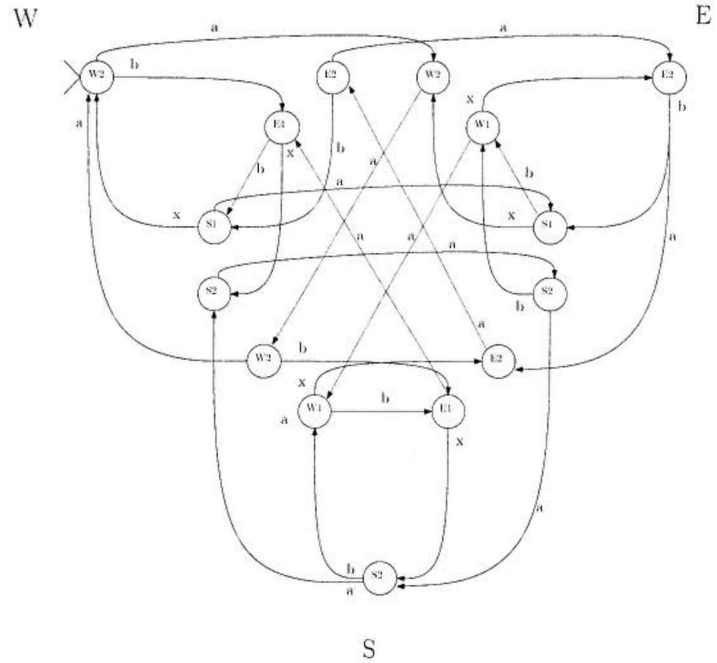

Figure 3.3 Supervisor of $G_A||G_B$

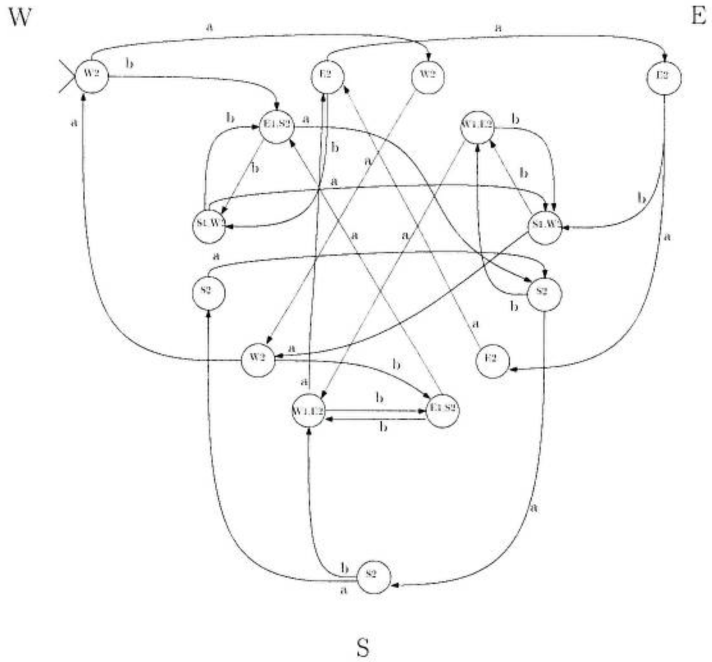

Figure 3.4 Observer of supervised $G_A||G_B$

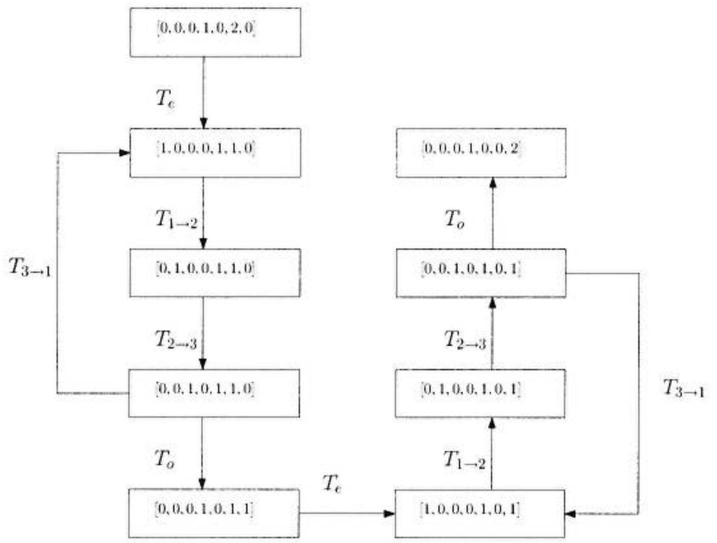

Figure 4.1 Reachable set and transition graph

4. Exercise

a) The incidence matrix is the following:

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

b) Transition T_e occurs each time a visitor enters the museum, T_o each time a visitor leaves the museum, whereas $T_{i\to j}$ occurs each time a visitor moves from room i to room j;

c) Place P_4 can be used to impose a maximum number of visitors inside the museum at any given time. In particular, with the prescribed initial marking n, n is the total maximum number of visitors allowed inside the Museum. For any marking M belonging to $\mathscr{R}(M_0)$, $M(P_4)$ is the number of places still available inside the Museum. $M(P_5)$ is the number of visitors currently inside the museum.

d) There are two P-invariant vector, $[0,0,0,1,1,0,0]$ and $[1,1,1,1,0,0,0]$. This corresponds to the fact that the number of visitors currently inside the museum plus the number of places available inside the museum is constant and equal to the maximum number of visitors allowed.

e) There are no P-decreasing vectors.

f) The directed graph associated to the reachable set is shown in Fig. 4.1

5. Exercise

a) The graph associated to the requested Markov chain is shown in Fig. 5.1. The

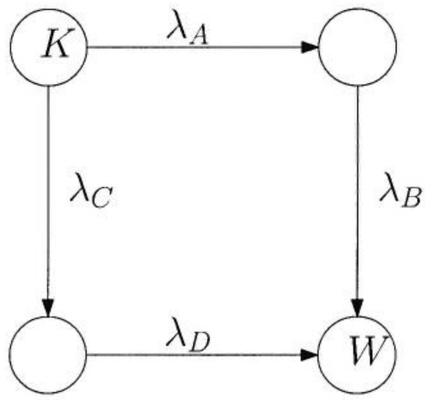

Figure 5.1 Markov chain modeling buses from Kensington to Wenstmister

equations describing the evolution of probabilities are reported below:

$$
\begin{aligned}
\dot{\pi}_1 &= -(\lambda_A + \lambda_C)\pi_1 \\
\dot{\pi}_2 &= \lambda_A \pi_1 - \lambda_B \pi_2 \\
\dot{\pi}_3 &= \lambda_C \pi_1 - \lambda_D \pi_3 \\
\dot{\pi}_4 &= \lambda_B \pi_2 + \lambda_D \pi_3
\end{aligned}
$$

b) Notice that W is an absorbing state for the Markov chain. The average trip duration is the so called average absorption time. To this end we compute the laplace transform of $\dot{\pi}_4(t)$ when $\pi(0) = [1,0,0,0]'$, as follows:

$$
\mathscr{L}(\dot{\pi}_4) = [0, \lambda_B, \lambda_D] \begin{bmatrix} s + (\lambda_A + \lambda_C) & 0 & 0 \\ -\lambda_A & s + \lambda_B & 0 \\ -\lambda_C & 0 & s + \lambda_D \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}
$$

$$
= \frac{\lambda_A \lambda_B (s + \lambda_D) + \lambda_C \lambda_D (s + \lambda_B)}{(s + \lambda_A + \lambda_C)(s + \lambda_B)(s + \lambda_D)}
$$

The average absorption time can be computed as:

$$
\frac{d}{ds} - \mathscr{L}(\dot{\pi}_4)\bigg|_{s=0} = \frac{\lambda_A + \lambda_B + \lambda_C}{(\lambda_A + \lambda_C)^2 \lambda_B^2} + \frac{\lambda_A + \lambda_C + \lambda_D}{(\lambda_A + \lambda_C)^2 \lambda_D^2}
$$

c) The average trip time waiting for bus A and B (regardless of C arrivals) is given by:

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

Hence, taking the first coming bus is quicker in average than taking bus A and B provided that:

$$
\frac{\lambda_A + \lambda_B + \lambda_C}{(\lambda_A + \lambda_C)^2 \lambda_B^2} + \frac{\lambda_A + \lambda_C + \lambda_D}{(\lambda_A + \lambda_C)^2 \lambda_D^2} < \frac{1}{\lambda_A} + \frac{1}{\lambda_B}
$$

Similarly, taking the first coming bus is quicker in average than taking bus C and D provided that:

$$
\frac{\lambda_A + \lambda_B + \lambda_C}{(\lambda_A + \lambda_C)^2 \lambda_B^2} + \frac{\lambda_A + \lambda_C + \lambda_D}{(\lambda_A + \lambda_C)^2 \lambda_D^2} < \frac{1}{\lambda_C} + \frac{1}{\lambda_D}
$$

d) In order to take into account buses traveling in both directions the Markov chain can be modified as shown in Fig. 5.2.

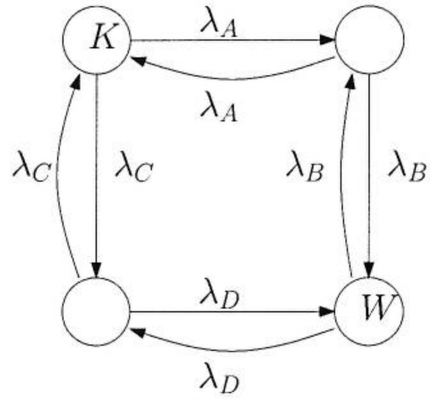

Figure 5.2 Markov chain modeling buses traveling in both directions

e) The chain is ergodic as the associated graph is strongly connected (hence it contains a unique absorbing component).

f) In order to compute the average amount of time spent in stop W it is enough to compute the steady state probability distribution. Its 4-th entry is the fraction of time spent in W. Notice that, since the matrix Q is symmetric, $\pi_\infty = [1/4, 1/4, 1/4, 1/4]$ regardless of parameters λ. Hence the controller spends 25% of time in W.

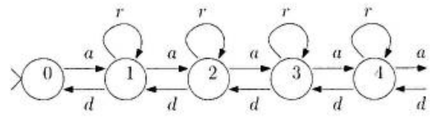

Figure 6.1 Reachable set and transition graph

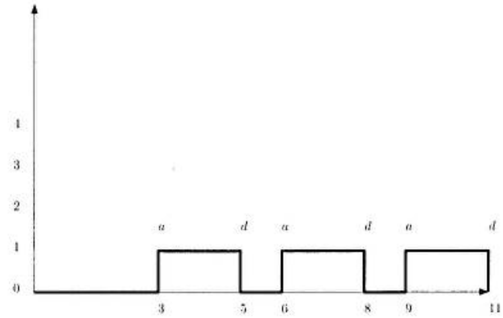

Figure 6.2 Time evolution with activation memory

6. Exercise

a) The transition diagram of the Petri Net N is as represented in Fig. 6.1.

b) The generated language is given by:

$$\mathscr{L} = \{w \in T^\star : \forall v \leq w, |v|_a \geq |v|_d \text{ and } \forall vr \leq w, |v|_a \geq |v|_d + 1\}$$

where $v \leq w$ denotes v is a prefix of w and $|v|_e$ denotes the number of events e in string v.

c) Let $\mathscr{L}(n)$ denote the generated language from initial marking n. It is obvious that $d^n \in \mathscr{L}(n)$ and $d^k \notin \mathscr{L}^{n-1}$ for all positive integers n and all $k < n$. Hence $\mathscr{L}(n)$ are all distinct from each other and no finite deterministic automaton can be found which generates the language $\mathscr{L}$.

d) The time evolution of the marking of Petri Net N relative to the 'activation memory' update is shown in Fig. 6.2;

e) The time evolution of the marking of Petri Net N relative to the 'total memory' update is shown in Fig. 6.3;

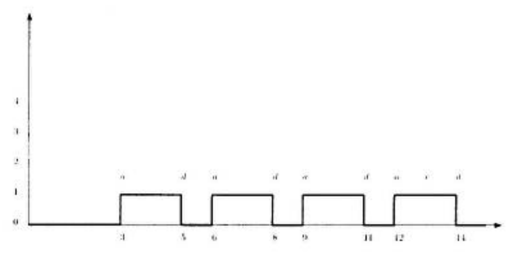

Figure 6.3 Time evolution with total memory