EE4-57
EE9-CS2-2

IMPERIAL COLLEGE LONDON

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING
EXAMINATIONS 2014

MSc and EEE/EIE PART IV: MEng and ACGI

Corrected Copy

DISCRETE-EVENT SYSTEMS

Thursday, 22 May 10:00 am

Time allowed: 3:00 hours

There are FOUR questions on this paper.

Answer ALL 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. An office is organized with 3 queues of people waiting to be served. Arrivals at the i-th queue ($i = 1,2,3$) are denoted by a_i, departures from the i-th queue are denoted by d_i.

a) Build a finite deterministic automaton G that models the queues, arrivals and departures, assuming each queue can at most contain 1 person waiting. [4]

b) Assume next that when a new customer needing service from queue 1 arrives and queue 1 is already full while queue 2 or queue 3 is not, then an exception o_1 is generated (meaning 'overflow' 1), and the customer is phisically sent to the next available empty queue. Similarly, if a new customer needing service from queue 2 arrives and queue 2 is already full while queue 3 or queue 1 is not, then an exception o_2 is generated (meaning 'overflow' 2), and the customer is physically sent to the next available empty queue (queue 3 or 1, respectively). Finally, if a new customer needing service from queue 3 arrives and queue 3 is already full while queue 1 or queue 2 is not, then an exception o_3 is generated (meaning 'overflow' 3), and the customer is physically sent to the next available empty queue (queue 1 or 2, respectively). Modify the previous automaton by including exceptions o_1, o_2 and o_3 as possible transitions. Denote the new automaton by G_O. [2]

c) Design an automaton G_L that is meant to act as a labeling device to discriminate between the situation in which o_1 has not occurred and the one in which o_1 has occurred. [2]

d) Compute the parallel composition $G_O||G_L$. [3]

e) Assume that events o_i ($i = 1,2,3$) are partially observable, namely only a generic event o, "overflow" would be generated each time any of o_1, o_2 or o_3 occurs. Replace in $G_O||G_L$ events o_i with a generic event o and denote by G_N the resulting non-deterministic automaton and by f_N the associated transition map; identify the state(s) x of G_N for which $f_N(x,o)$ is a set of cardinality bigger than 1. [2]

f) Build a diagnoser G_D that can decide if events o_1 have occurred (meaning YES / NO / MAYBE) by processing events of type $a_1, a_2, a_3, d_1, d_2, d_3$ and o only. [7]

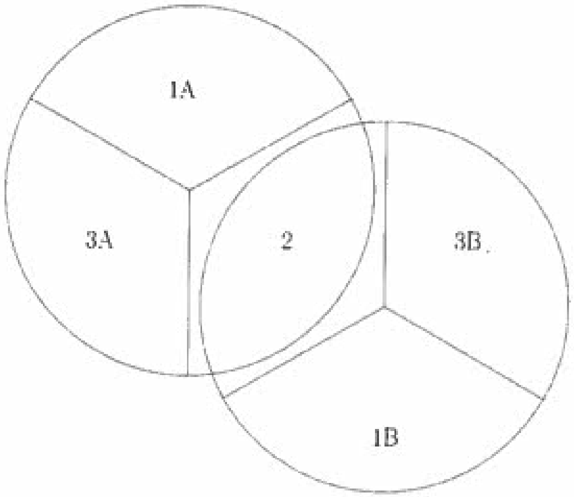

Figure 2.1 Robots A and B operating areas

2. Two robots operate in overlapping regions of space, as shown in Fig. 2.1. Robot A is a high priority robot that rotates clockwise between 3 regions of space (regions 1A,2 and 3A, respectively). At each tick of a clock it either stays in the same position (event t_s) or it moves to the next position in a clockwise direction, event t_c. Robot B is a low priority robot that can move asynchronously between positions $3B$ and $1B$, and $1B$ and $3B$, following events c or a, respectively. When in position $1B$ an event c can trigger transition to a synchronized mode of operation in which, following events t_s or t_c, the robot enters position 2, and then leaves it at the following tick of the clock in favour of position $3B$ (again event t_c or t_s). Back in position $3B$ or $1B$, the robot is free to operate asynchronously, and events t_s or t_c leave its position unaffected.

a) Build a finite deterministic automaton G_A modeling Robot A and its occupation of positions 1, 2, 3, respectively, assuming position 1A as initial state. [4]

b) Build a finite deterministic automaton G_B modeling Robot B, its occupation of positions 1, 2, 3 and the transition to the synchronous mode of operation, assuming position 1B as initial state. [4]

c) Compute the parallel composition $G_A||G_B$. [4]

d) Design an automaton H to implement the specification that robots should not both operate simultaneously in position 2. [2]

e) Is this specification controllable, assuming the set of uncontrollable events to be $E_{uc} = \{t_c, t_s\}$? (justify your answer). [2]

f) Realize a supervisor that implements the supremal controllable sublanguage $\mathscr{L}(H)^{\uparrow C}$. [4]

3. A machine works according to the following set of rules: pieces to be processed are dispatched to the machine, event a, and accumulated in a stack; in order to process the pieces, event p, the machine needs to first pick up two distinct tools, Tool A and Tool B, (events pA and pB respectively); after processing the pieces the machine may decide to release the tools (event rA and rB respectively). Processed pieces can then exit the factory, event x.

a) Build a model of the machine, the arrival of pieces and their departure from the factory as well as the tools A and B employed by means of a marked Petri Net $< N, M_0 >$. [5]

b) Assume that two machines are sharing the same tools, how can you modify the previous model to take into account this situation? [3]

c) How can you modify the previous network in order to take into account that only a finite number of processed pieces can be hosted inside the factory? [2]

d) For the Petri Net in item a), compute the P-invariant vectors; what is their physical meaning? [3]

e) For the Petri Net in item a), sketch the coverability graph. [4]

f) Which places of network N are structurally bounded ? Do they differ from the places which are behaviourally bounded for N, with initial marking M_0 as specified in item a)? [3]

4. In a car park it has been observed that cars arrive with a normal probability distribution of arrival times, at a rate λ_a. On the other hand, cars parked inside leave with a normal probability distribution of departure times with a rate λ_d.

a) Assume next that the car park has a total of 3 available parking spaces and that arrivals of cars when the parking is full do not occur. Build a continuous time Markov Chain to model the time behaviour of the total number of parked cars. (*Hint: pay attention to the rate at which cars leave the car park when occupancy is equal to n*). [6]

b) What is the asymptotic average number of cars in the car park? [4]

c) Assume next that parking lots are numbered 1, 2, 3. Assume that when a car comes it always takes the first available slot (viz. the one with the smallest number). Sketch the transition diagram of a Markov chain modelling the occupancy of all 3 parking slots. (*Hint: 8 states are needed*). [5]

d) Consider now the occupancy of parking slot 1 only. Consider the stochastic process that takes the value TAKEN when parking 1 is occupied and AVAILABLE otherwise. Show, by using the Markov chain in the previous item, that this can be modeled as a Markov chain with 2 states. Find the average occupancy of parking slot 1. [3]

e) Explain why a similar argument cannot be repeated for parking slot 2. [2]