EE4-57

SOLUTIONS: DISCRETE EVENT SYSTEMS
MASTER IN CONTROL

1. Exercise

a) The Robot moving on the platform can be modeled according to the automaton shown in Fig. 1.1.

b) The labeling automaton has two states, $X = \{N,Y\}$, event set $E = \{d\}$, initial state N and transition function $f : X \times E \to X$ defined as follows: $f(N,d) = Y$ and $f(Y,d) = Y$.

c) The concurrent composition of G_L and G is shown in Fig. 1.2.

d) Replacing d events by ε events we obtain a non-deterministic automaton G_N. The diagnoser can be designed by computing the observer automaton $Obs(G_N)$. The result is shown in Fig. 1.3.

e) Let the states $\{0Y\}$ and $\{AY,BY,CY,DY\}$ be the terminal states in the diagnoser automaton. Then, it is possible to realize that the following states are equivalent to each other: $\{0Y\} \equiv \{AY,BY,CY,DY\}$ and also $\{0N\} \equiv \{0N,YN\}$. As a result, the minimal diagnoser automaton is shown in Fig. 1.4.

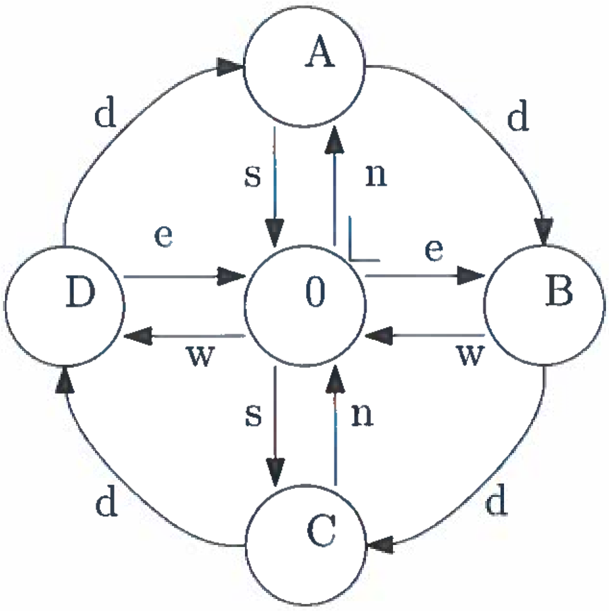

Figure 1.1 The automaton G

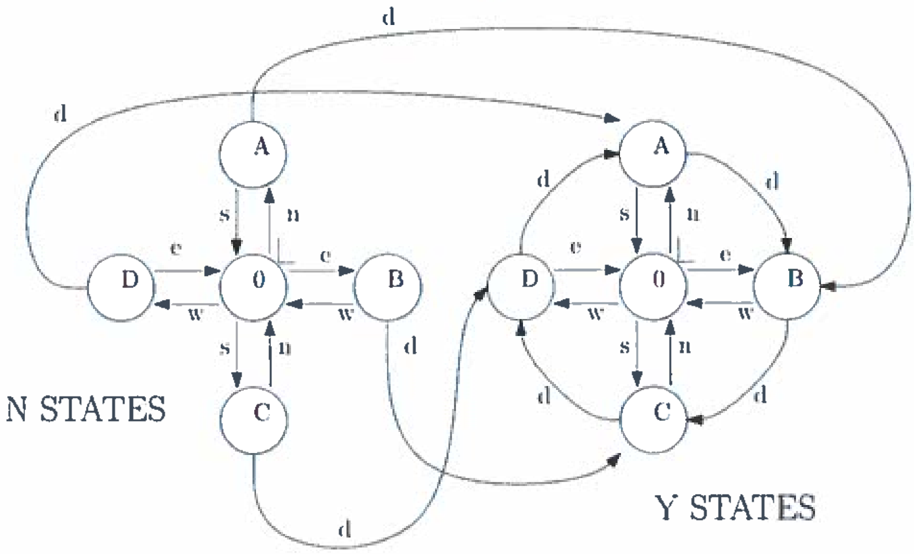

Figure 1.2 The automaton $G||G_L$

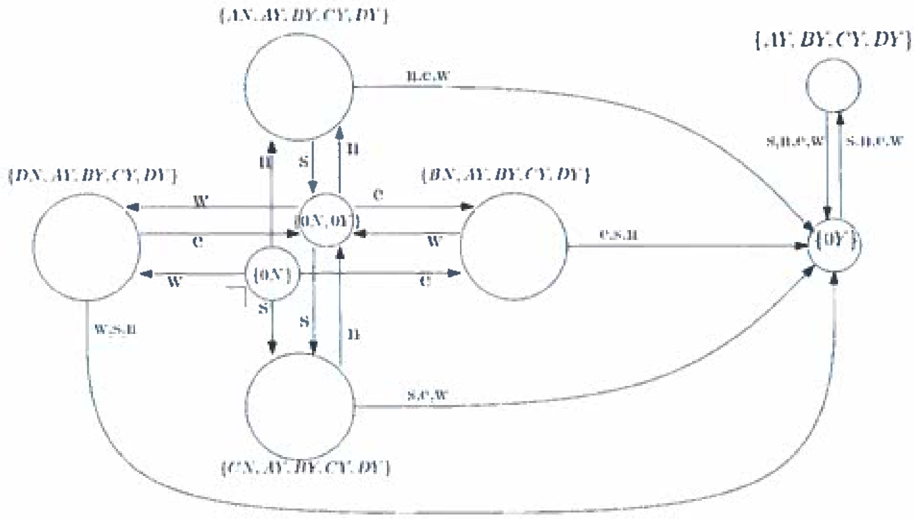

Figure 1.3 The diagnoser automaton

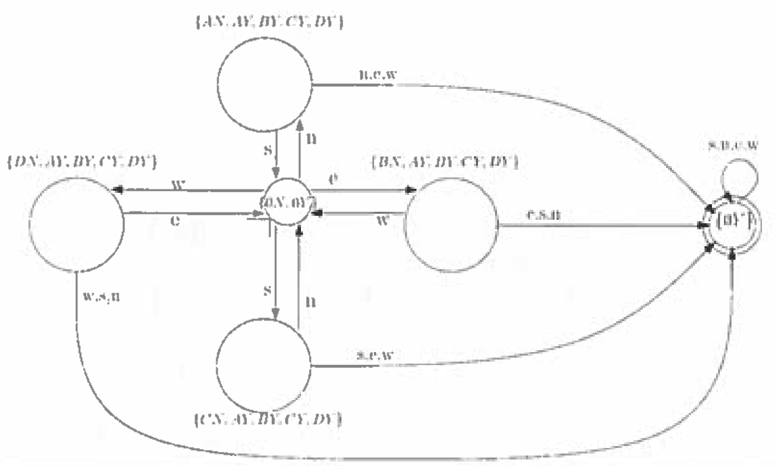

Figure 1.4 Minimal diagnoser automaton

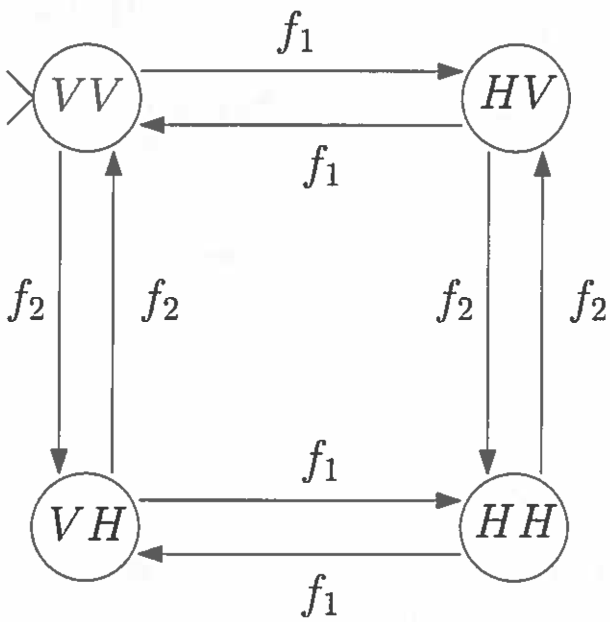

Figure 2.1 Automaton F

2. Exercise

a) Each automaton has two states, therefore $X = \{H, V\}$, standing for horizontal and vertical respectively. The associated alphabets are $E_i = \{f_i\}$, $i = 1, 2$, and initial state is V in both cases. The transition function $\tilde{f}_i$ is simply given as

$$\tilde{f}_i(H, f_i) = V \qquad \tilde{f}_i(V, f_i) = H.$$

b) The simultaneous evolution of the two flippers can be achieved by taking the concurrent composition of the two automata, $F = F_1 || F_2$. The resulting automaton is sketched in Fig. 2.1.

c) The requested specification can be achieved by state-splitting, that is by having two states of HH type in order to discriminate what the last event has been. Therefore, the specification can be represented by the automaton G_{sp} shown in Fig. 2.2.

d) To decide controllability of the specification G_{sp} notice that the parallel composition $F || G_{sp}$ is an automaton isomorphic to G_{sp} and the modified labels shown

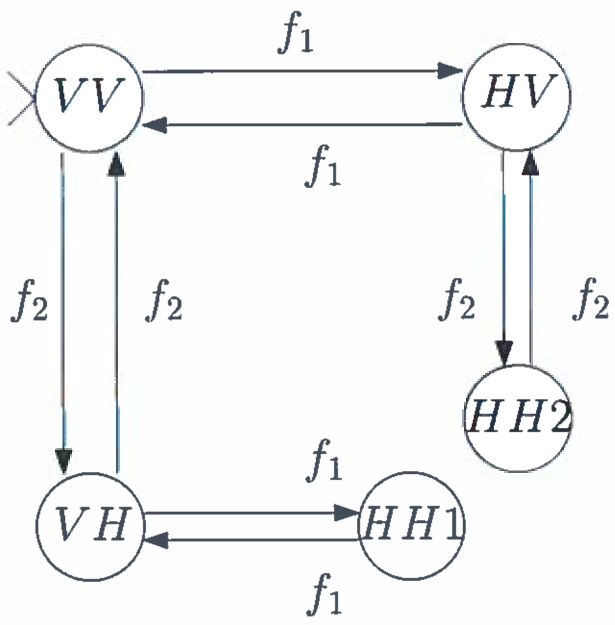

Figure 2.2 Automaton G_{sp}

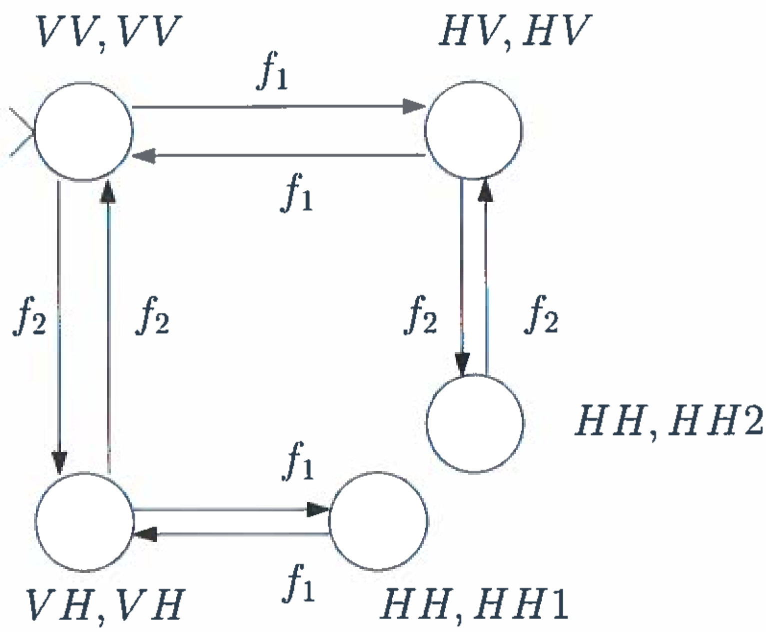

Figure 2.3 Automaton $F||G_{sp}$

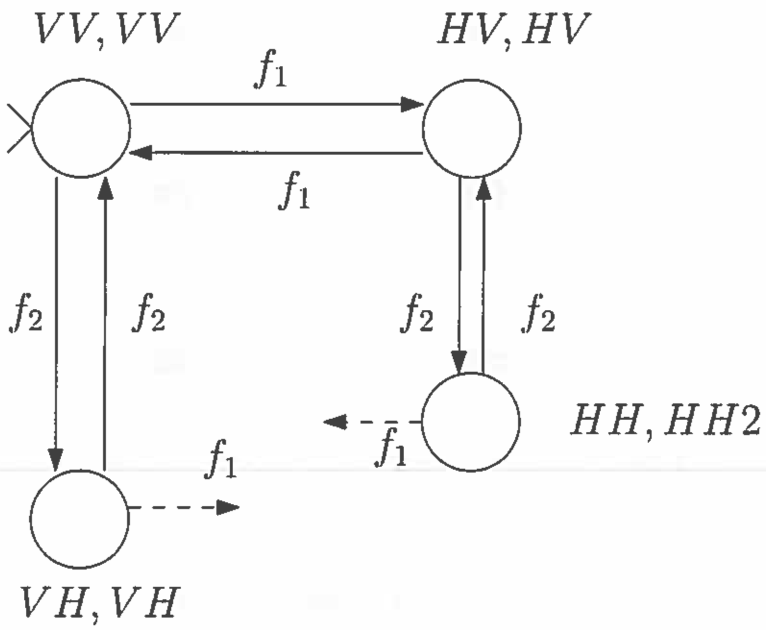

Figure 2.4 Automaton generating $\mathscr{L}(G_{sp})^{\uparrow C}$

if Fig. 2.3. From this it can be seen that no events are disabled in the states VV,VV, HV,HV and VH,VH, while event f_2 is disabled in state $HH,HH1$ and event f_1 is disabled in state $HH,HH2$. Since f_2 is uncontrollable, it cannot be disabled and therefore the specification $\mathscr{L}(G_{sp})$ is uncontrollable.

e) To compute the maximal controllable sublanguage we remove from $F||G_{sp}$ the state $HH,HH1$ where f_2 gets disabled. The resulting automaton implements an admissible supervisor as it only disables f_1 events. See Fig. 2.4. The language generated by this automaton is the maximal controllable sublanguage.

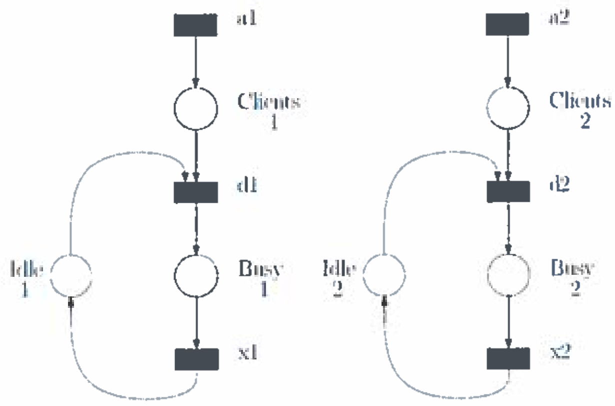

Figure 3.1 Petri Net modelling two queues in parallel

3. Exercise

a) In order to build the model of 2 parallel queues it is enough to consider the juxtaposition of two identical networks as in Fig. 3.1.

b) The Petri Net should be modified by adding a "residual capacity" place, as shown in Fig. 3.2. The initial marking of the new place should be n.

c) The incidence matrix of the Net is given by:

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

where places are taken in the following order: Clients1, Idle1, Busy1, Clients2, Idle2, Busy 2, Capacity. Transitions are instead ordered as $a1$, $d1$, $x1$, $a2$, $d2$, $x2$. Accordingly P-semiflows can be computed. There are 3 P-semiflows:

$$[1,0,0,1,0,0,1], \qquad [0,1,1,0,0,0,0], \qquad [0,0,0,0,1,1,0].$$

Notice that their sum equals $[1,1,1,1,1,1,1]$ is also a P-semiflow, and therefore the network is structurally conservative and bounded.

d) See Fig. 3.3 for a net allowing clients from queue one to be served by server two. A symmetric transition can be added to allow clients from queue two to be served from server one (not shown for the sake of clarity of the graph).

e) The Petri Net is not structurally bounded as transition a can fire infinitely many times in a sequence. This leads to unboundedness of the place "Clients". On the other hand, the total number of tokens in places Idle and Busy is conserved, $[0,1,1]$ is a P-semiflow. Hence those places are structurally bounded.

f) The coverability graph is seen in Fig. 3.4.

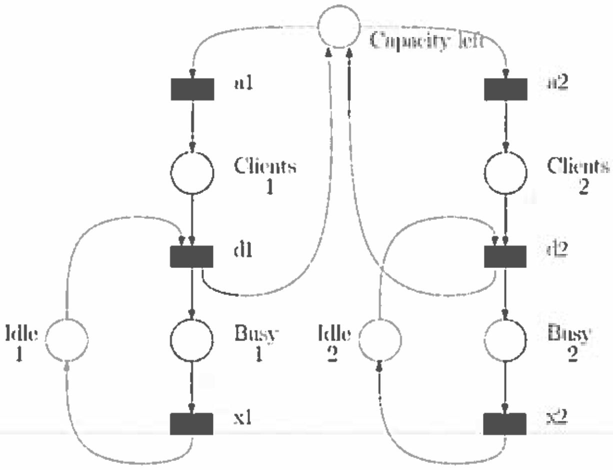

Figure 3.2 Petri Net modelling two queues with total capacity constraint

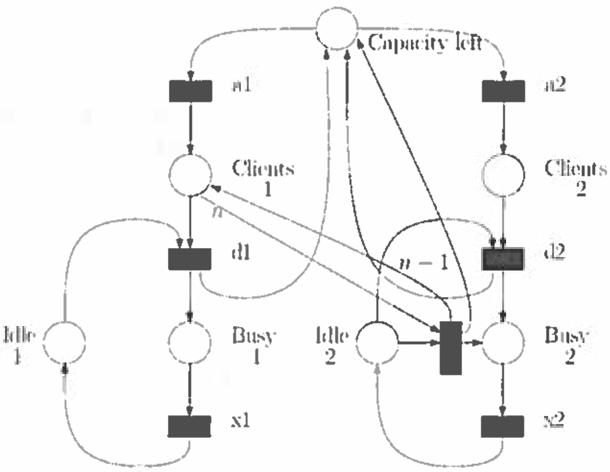

Figure 3.3 Petri Net allowing clients from queue 1 to be served by server 2

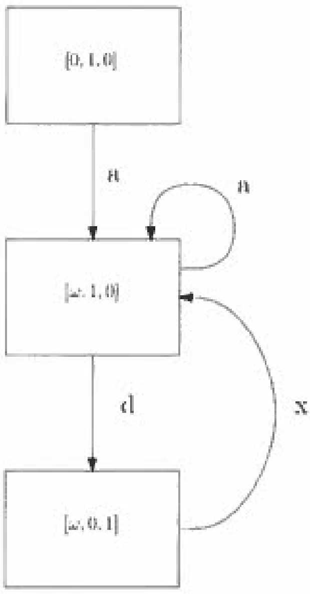

Figure 3.4 Coverability graph.

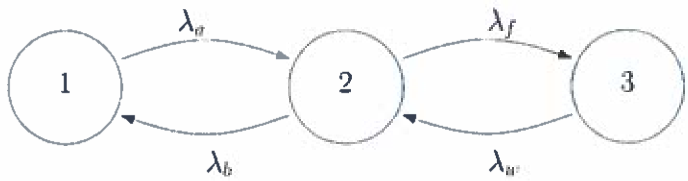

Figure 4.1 Markov chain transition diagram

4. Exercise

a) The transition diagram of the Markov chain is shown if Fig. 4.1, where state 1 denotes the car being idle at the base, state 2 denotes the car being used by a customer and state 3 denotes the car being at the workshop.

b) Letting π_i denote the probability of being in state i and $\pi = [\pi_1, \pi_2, \pi_3]$, the Kolmogorov equations are given as:

$$\dot{\pi}(t) = \pi(t)Q, \qquad Q = \begin{bmatrix} -\lambda_a & \lambda_a & 0 \\ \lambda_b & -\lambda_b - \lambda_f & \lambda_f \\ 0 & \lambda_w & -\lambda_w \end{bmatrix}.$$

c) The Markov chain is ergodic since its transition diagram is strongly connected. Hence we may compute steady-state probability distributions according to:

$$\pi_\infty Q = 0, \qquad \pi\mathbf{1} = 1.$$

In our case this gives:

$$\pi_{1\infty} = \frac{\lambda_b \lambda_w}{\lambda_a \lambda_f + \lambda_a \lambda_w + \lambda_b \lambda_w}$$

$$\pi_{2\infty} = \frac{\lambda_a \lambda_w}{\lambda_a \lambda_f + \lambda_a \lambda_w + \lambda_b \lambda_w}$$

$$\pi_{3\infty} = \frac{\lambda_a \lambda_f}{\lambda_a \lambda_f + \lambda_a \lambda_w + \lambda_b \lambda_w}.$$

The fraction of time the car is busy is given by $\pi_{2\infty}$.

d) When two cars are considered, the rate of requests of cars to be hired is still the same λ_a. On the other hand, the rate at which cars are returned and are subject to faults doubles as soon as two cars are being used by a customer. Similarly, assuming faulty cars are sent to different workshops, the rate at which cars are returned after a fault doubles if both cars are faulty. Hence the modified Markov chain is as in Fig. 4.2.

e) To compute the average time it takes for a rented car to return to the base we modify the Markov chain as shown if Fig. 4.3, and compute the average absorption time of state 3, given that the initial state is 1. Letting $\tilde{Q}$ be the matrix:

$$\tilde{Q} = \begin{bmatrix} -\lambda_f - \lambda_b & \lambda_f \\ \lambda_w & -\lambda_w \end{bmatrix}$$

we see that:

$$\dot{\pi}_3(t) = \lambda_b \pi_1(t)$$

$$[\dot{\pi}_1(t), \dot{\pi}_2(t)] = [\pi_1(t), \pi_2(t)]\tilde{Q} \qquad [\pi_1(0), \pi_2(0)] = [1, 0].$$

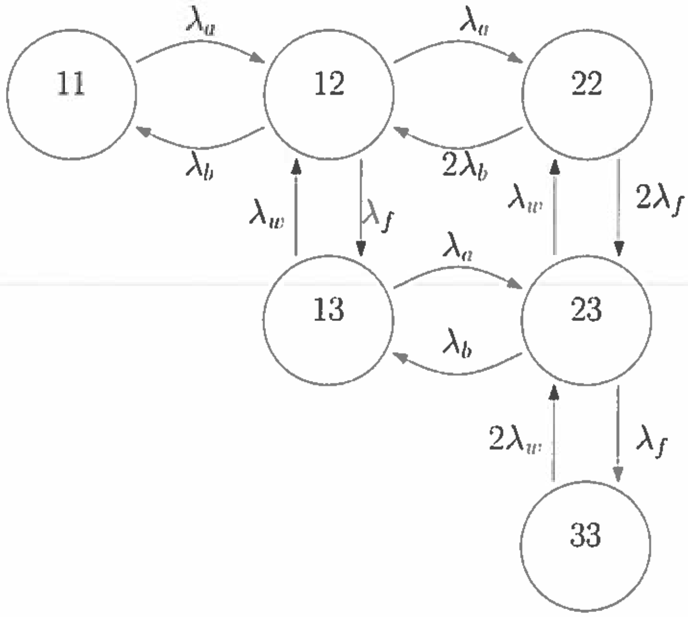

Figure 4.2 Two rented cars: modified Markov chain

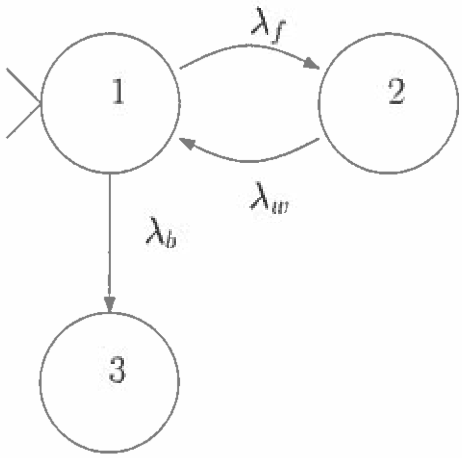

Figure 4.3 Markov chain for return time

We need to compute:

$$E[t_a] = \int_0^{+\infty} t\,\dot{\pi}_3(t)\,dt.$$

This can be done in the frequency domain by realizing that the Laplace transform of $\dot{\pi}_3(t)$ is given by:

$$\mathscr{L}(\dot{\pi}_3(t)) = [1,0]\,(sI - \tilde{Q})^{-1} \begin{bmatrix} \lambda_b \\ 0 \end{bmatrix}.$$

Differentiating with respect to s and substituting $s = 0$ yields:

$$E[t_a] = \frac{\lambda_f + \lambda_w}{\lambda_b \lambda_w}.$$