



Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Ηλεκτρολόγων Μηχανικών & Μηχανικών
Υπολογιστών

Ροή Δ- Συστήματα Αναμονής- 6ο Εξάμηνο
Ακαδημαϊκό Έτος 2020-2021

3η Εργαστηριακή Άσκηση
Θωμάς Πετρόπουλος- el18915

Στην άσκηση αυτή μας ζητείται να μελετήσουμε ένα σύστημα M/M/1/10 για $\lambda=\{1,5,10\}$ και $\mu=5$ πελάτες/min. Θα κάνουμε χρήση του αρχείου κώδικα που δίνεται μαζί με την εκφώνηση της άσκησης.

(1) Για το debugging θα κατασκευάσουμε trace table ούτως ώστε να ελέγξουμε τις 30 πρώτες μεταβάσεις και να επαληθεύσουμε ότι δεν υπάρχει κάποιο λογικό σφάλμα στον κώδικα. Για τις τρεις διαφορετικές τιμές των λ θα κατασκευάσουμε πίνακες 5 στηλών όπου η πρώτη θα αποθηκεύει τον A/A της μετάβασης, η δεύτερη την κατάσταση που βρίσκεται το σύστημα, η τρίτη τον αριθμό των αφίξεων στην παρούσα κατάσταση και οι επόμενες δυο θα έχουν 1 ή 0 ανάλογα με το αν πρόκειται για άφιξη ή αναχώρηση.

Για $\lambda=1$ έχουμε:

tracetable =

0	0	0	1	0
2	1	1	1	0
3	2	1	0	1
4	1	1	1	0
5	2	2	1	0
6	3	1	0	1
7	2	2	0	1
8	1	1	0	1
9	0	0	1	0
10	1	2	0	1
11	0	0	1	0
12	1	3	0	1
13	0	0	1	0
14	1	4	1	0
15	2	3	0	1
16	1	4	0	1
17	0	0	1	0
18	1	5	1	0
19	2	4	0	1
20	1	5	0	1
21	0	0	1	0
22	1	6	0	1
23	0	0	1	0
24	1	7	0	1
25	0	0	1	0
26	1	8	0	1
27	0	0	1	0
28	1	9	1	0
29	2	5	0	1
30	1	9	0	1

Για $\lambda=5$ έχουμε:

tracetable =

0	0	0	1	0
2	1	1	1	0
3	2	1	0	1
4	1	1	1	0
5	2	2	1	0
6	3	1	1	0
7	4	1	0	1
8	3	1	0	1
9	2	2	1	0
10	3	2	0	1
11	2	2	1	0
12	3	3	1	0
13	4	2	1	0
14	5	1	1	0
15	6	1	1	0
16	7	1	1	0
17	8	1	0	1
18	7	1	1	0
19	8	2	1	0
20	9	1	1	0
21	10	1	1	0
22	10	1	1	0
23	10	1	1	0
24	10	1	0	1
25	9	1	1	0
26	10	2	1	0
27	10	2	0	1
28	9	1	1	0
29	10	3	1	0
30	10	3	0	1

Καθώς και για $\lambda=10$ έχουμε:

tracetable =

0	0	0	1	0
2	1	1	1	0
3	2	1	0	1
4	1	1	1	0
5	2	2	1	0
6	3	1	1	0
7	4	1	0	1
8	3	1	0	1
9	2	2	1	0
10	3	2	0	1
11	2	2	1	0
12	3	3	1	0
13	4	2	1	0
14	5	1	1	0
15	6	1	1	0
16	7	1	1	0
17	8	1	1	0
18	9	1	1	0
19	10	1	1	0
20	10	1	1	0
21	10	1	1	0
22	10	1	1	0
23	10	1	1	0
24	10	1	0	1
25	9	1	1	0
26	10	2	1	0
27	10	2	0	1
28	9	1	1	0
29	10	3	1	0
30	10	3	0	1

2) Εκτελούμε τον παρακάτω κώδικα για τις διάφορες τιμές του λ :

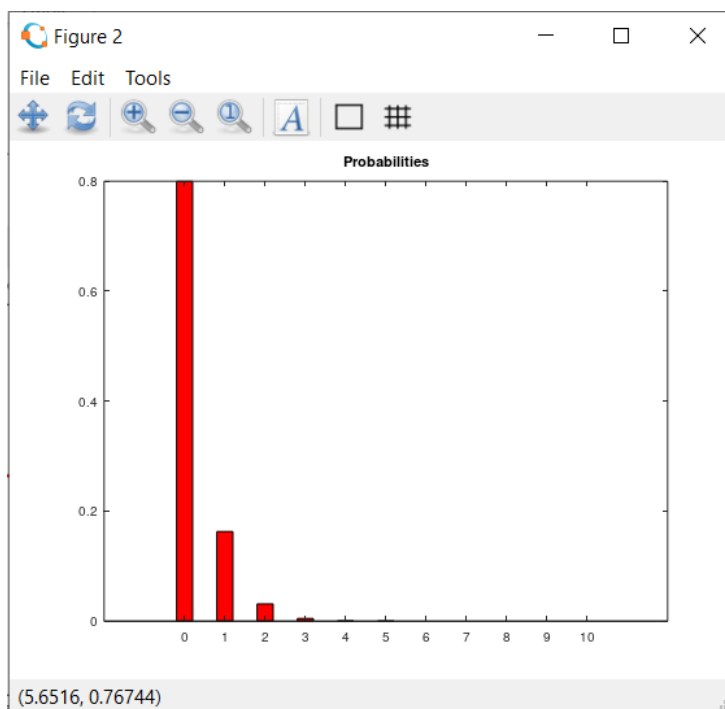
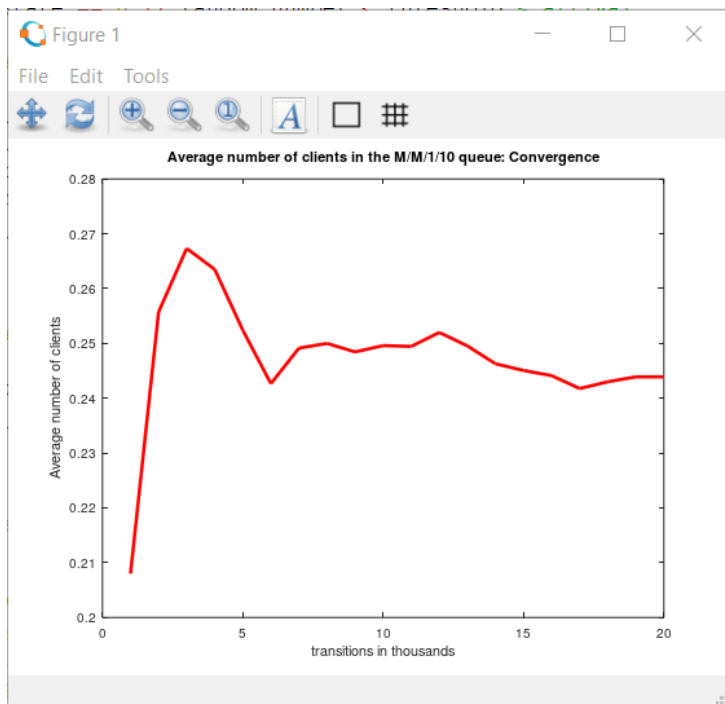
```
1 clc;
2 clear all;
3 close all;
4 rand('seed',12163);
5 total_arrivals = 0; % to measure the total number of arrivals
6 current_state = 0; % holds the current state of the system
7 previous_mean_clients = 0; % will help in the convergence test
8 index = 0;
9 arrivals= [0,0,0,0,0,0,0,0,0,0];
10 lambda = 1;
11 mu = 5;
12 threshold = lambda/(lambda + mu); % the threshold used to calculate probabilities
13 %k=0;
14 %tracetable=[];
15 transitions = 0; % holds the transitions of the simulation in transitions steps
16 while transitions >= 0 %&& transitions < 30
17     transitions = transitions + 1; % one more transitions step
18     % k = k +1;
19     %if 0 < k < 31 && current_state > 0
20         % tracetable(k,1) = k;
21         %tracetable(k,2) = current_state;
22         %tracetable(k,3) = arrivals(current_state);
23     %endif
24     if mod(transitions,1000) == 0 % check for convergence every 1000 transitions steps
25         index = index + 1;
26         for j=1:length(arrivals)
27             P(j) = arrivals(j)/total_arrivals; % calculate the probability of every state in the system
28         endfor
29         mean_clients = 0; % calculate the mean number of clients in the system
30         for j=1:length(arrivals)
31             mean_clients = mean_clients + (j-1).*P(j);
32         endfor
33         to_plot(index) = mean_clients;
34     end
35     if abs(mean_clients - previous_mean_clients) < 0.00001 || transitions > 1000000 % convergence test
36         break;
37     endif
38     previous_mean_clients = mean_clients;
```

```

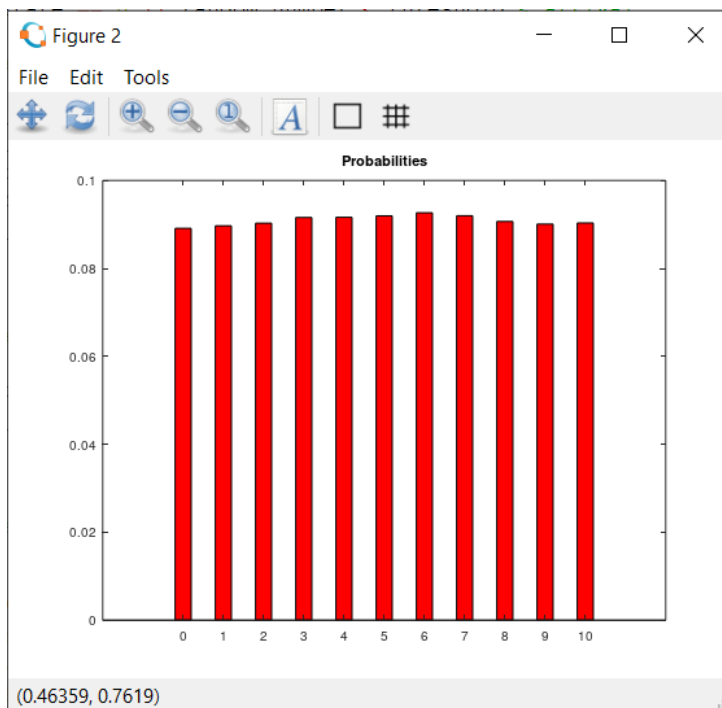
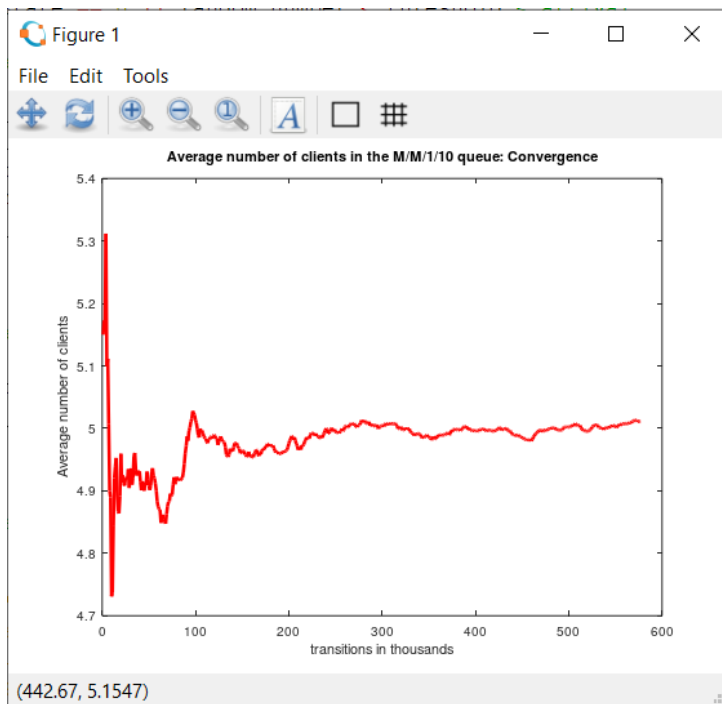
39
40     endif
41     random_number = rand(1); % generate a random number (Uniform distribution)
42     if current_state == 0 || random_number < threshold % arrival
43         %if 0 < k < 31
44         % tracetable(k,4) = 1;
45         %endif
46         total_arrivals = total_arrivals + 1;
47         x= arrivals(current_state + 1) +1;
48         arrivals(current_state + 1) =x;
49         if (current_state != 10)
50             current_state = current_state + 1;
51         endif
52     else
53         %if 0 < k < 31
54         % tracetable(k,5) = 1;
55         %endif
56         if current_state != 0 % no departure from an empty system
57             current_state = current_state - 1;
58         endif
59     endif
60 endwhile
61 %display(tracetable);
62
63 figure(1);
64 plot(to_plot,"r","linewidth",1.3);
65 title("Average number of clients in the M/M/1/10 queue: Convergence");
66 xlabel("transitions in thousands");
67 ylabel("Average number of clients");
68
69 figure(2);
70 bar(0:1:(length(arrivals)-1),P,'r',0.4);
71 title("Probabilities");

```

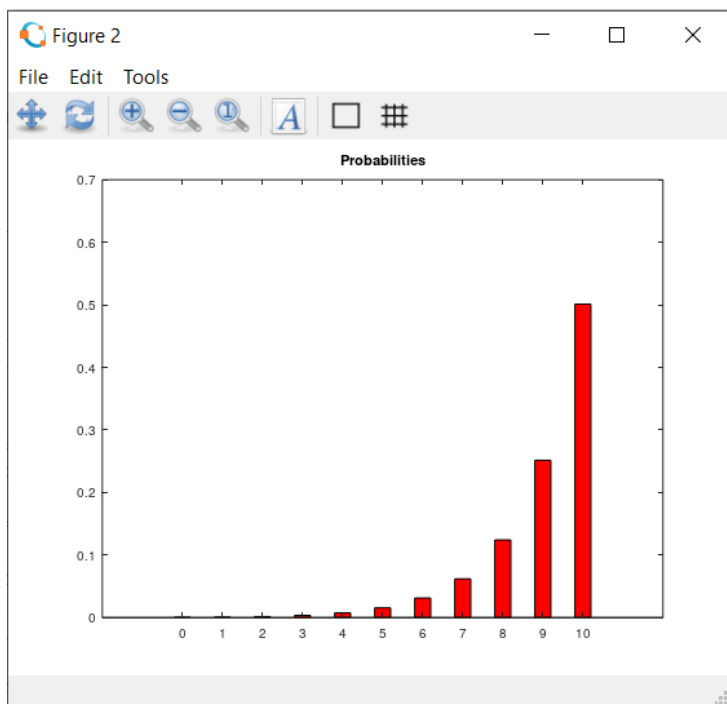
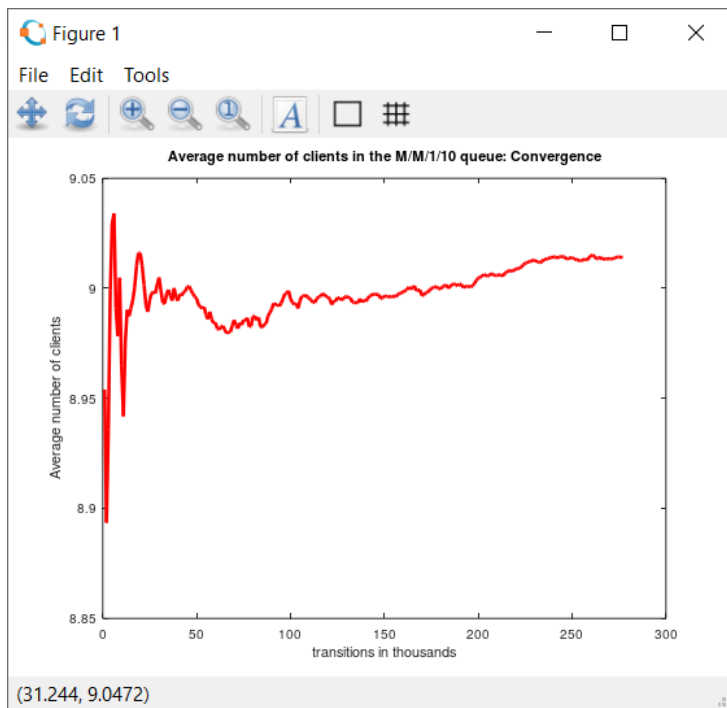
Για $\lambda=1$ έχουμε:



Για $\lambda=5$ έχουμε:



Για $\lambda = 10$ έχουμε:



3) Παρατηρούμε ότι για $|\lambda - \mu| \rightarrow 0$ χρειαζόμαστε περισσότερες μεταβάσεις για να ισχύσει το κριτήριο σύγκλισης. Συνεπώς για $\lambda=10$ συγκλίνει γρηγορότερα, κατόπιν για $\lambda=1$ και μετά για $\lambda=5$.

4) Αν είχαμε εκθετικό εξυπηρετητή που εξαρτάται απ την παρούσα κατάσταση (μέσω του i) , θα τροποποιούσαμε το threshold στο πρόγραμμα μας, μιας και το νέο μ_i δεν θα είναι σταθερή τιμή καθώς και θα το ανανεώναμε στη συνθήκη while με το νέο του i κάθε φορά.