

Συστήματα Αναμονής

3^η Ομάδα Ασκήσεων

Αννα Κουτσώνη 03120019

Προσομοίωση συστήματος M/M/1/10

1. Για την μελέτη ενός συστήματος M/M/1/10, για το οποίο επιλέγουμε $\lambda=5$ και $\mu=5$, όπως δίνεται, χρησιμοποιούμε τον παρακάτω κώδικα σε octave:

```
1 %1
2
3 clc;
4 clear all;
5 close all;
6
7 P = [0,0,0,0,0,0,0,0,0,0];
8 arrivals = [0,0,0,0,0,0,0,0,0,0];
9 total_arrivals = 0; % to measure the total number of arrivals
10 current_state = 0; % holds the current state of the system
11 previous_mean_clients = 0; % will help in the convergence test
12 index = 0;
13
14 lambda = 5;
15 mu = 5;
16 threshold = lambda/(lambda + mu); % the threshold used to calculate probabilities
17
18 transitions = 0; % holds the transitions of the simulation in transitions steps
19
20 while transitions >= 0
21     transitions = transitions + 1; % one more transitions step
22
23     if mod(transitions,1000) == 0 % check for convergence every 1000 transitions steps
24         index = index + 1;
25         for i=1:length(arrivals)
26             P(i) = arrivals(i)/total_arrivals; % calculate the probability of every state in
27         endfor
28
29         mean_clients = 0; % calculate the mean number of clients in the system
30         for i=1:length(arrivals)
31             mean_clients = mean_clients + (i-1).*P(i);
32         endfor
33
34         to_plot(index) = mean_clients;
35
36         if abs(mean_clients - previous_mean_clients) < 0.00001 || transitions > 200000 % co
37             break;
38         endif
39
40         previous_mean_clients = mean_clients;
41
42     endif
43
44     random_number = rand(1); % generate a random number (Uniform distribution)
45     if current_state == 0 || random_number < threshold % arrival
46         if current_state < 11
47             total_arrivals = total_arrivals + 1;
48             if transitions < 31
49                 display("Current state = ");
50                 disp(current_state);
51                 display("Next transition");
52                 display("arrival");
53                 display("Total arrivals in current state = ");
54                 disp(arrivals(current_state+1));
55             endif
56             arrivals(current_state + 1) = arrivals(current_state + 1) + 1; % increase the num
57             if current_state < 10
58                 current_state = current_state + 1;
59             endif
60         endif
61     else % departure
62         if current_state != 0 % no departure from an empty system
63             if transitions < 31
64                 display("Current state = ");
65                 disp(current_state);
66                 display("Next transition");
67                 display("departure");
68                 display("Total arrivals in current state = ");
```

```

69         disp(arrivals(current_state+1));
70     endif
71     current_state = current_state - 1;
72 endif
73 endif
74 endwhile
75
76
77
78 display("State probabilities:");
79 for i=1:length(arrivals)
80     display(P(i));
81 endfor
82
83 g = lambda*(1-P(11));
84 average_delay_time = mean_clients / g;
85 display("Average delay time =");
86 disp(average_delay_time);
87 display("Blocking probability =");
88 disp(P(11));
89
90 figure(1);
91 plot(to_plot,"c","linewidth",1.3);
92 title("Average number of clients in the M/M/1 queue: Convergence");
93 xlabel("transitions in thousands");
94 ylabel("Average number of clients");
95
96 x=[0,1,2,3,4,5,6,7,8,9,10];
97 figure(2);
98 bar(x,P,0.4);
99 title("Probabilities")

```

Για το debugging της προσομοίωσης:

- Οι 30 πρώτες μεταβάσεις:

```

Current state =
0
Next transition
arrival
Total arrivals in current state =
0
Current state =
1
Next transition
departure
Total arrivals in current state =
0
Current state =
0
Next transition
arrival
Total arrivals in current state =
1
Current state =
1
Next transition
departure
Total arrivals in current state =
0
Current state =
0
Next transition
arrival
Total arrivals in current state =
2
Current state =
1
Next transition
departure
Total arrivals in current state =
0
Current state =
0
Next transition
arrival

```

```

Total arrivals in current state =
3
Current state =
1
Next transition
arrival
Total arrivals in current state =
0
Current state =
2
Next transition
arrival
Total arrivals in current state =
0
Current state =
3
Next transition
departure
Total arrivals in current state =
0
Current state =
2
Next transition
departure
Total arrivals in current state =
1
Current state =
1
Next transition
departure
Total arrivals in current state =
1
Current state =
0
Next transition
arrival
Total arrivals in current state =
4
Current state =
1
Next transition
departure
Total arrivals in current state =
1
Current state =
0
Next transition
arrival
Total arrivals in current state =
5
Current state =
1
Next transition
departure
Total arrivals in current state =
1
Current state =
0
Next transition
arrival
Total arrivals in current state =
6
Current state =
1
Next transition
arrival
Total arrivals in current state =
1
Current state =
2
Next transition
arrival
Total arrivals in current state =
1
Current state =
3
Next transition
arrival
Total arrivals in current state =
0

```

```

Current state =
4
Next transition
arrival
Total arrivals in current state =
0
Current state =
5
Next transition
departure
Total arrivals in current state =
0
Current state =
4
Next transition
arrival
Total arrivals in current state =
1
Current state =
5
Next transition
arrival
Total arrivals in current state =
0
Current state =
6
Next transition
departure
Total arrivals in current state =
0
Current state =
5
Next transition
departure
Total arrivals in current state =
1
Current state =
4
Next transition
arrival

```

```

Total arrivals in current state =
2
Current state =
5
Next transition
arrival
Total arrivals in current state =
1
Current state =
6
Next transition
departure
Total arrivals in current state =
0
Current state =
5
Next transition
departure
Total arrivals in current state =
2

```

- Οι πιθανότητες καταστάσεως του συστήματος

```

State propabilities:
0.091377
0.089441
0.089877
0.088985
0.091348
0.088900
0.090096
0.091149
0.093588
0.092477
0.092762

```

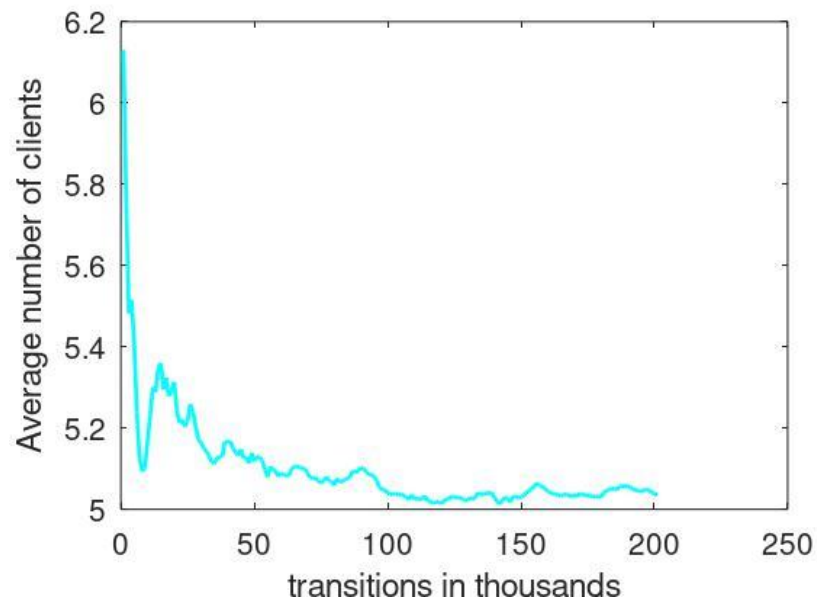
- Η πιθανότητα απόρριψης πελάτη από το σύστημα (Blocking Probability)

```
Blocking probability =
0.092762
\\ |
```

- Ο μέσος χρόνος καθυστέρησης ενός πελάτη στο σύστημα

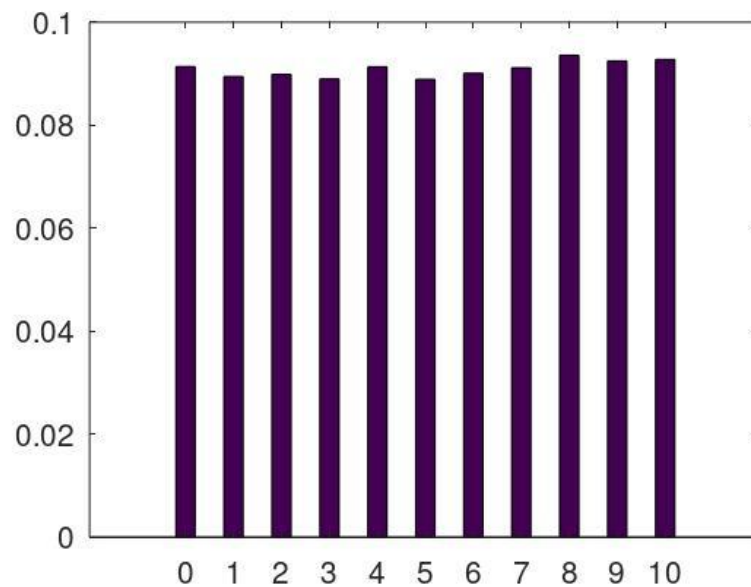
```
Average delay time =
1.1096
```

- **Average number of clients in the M/M/1 queue: Converge**



-

Probabilities

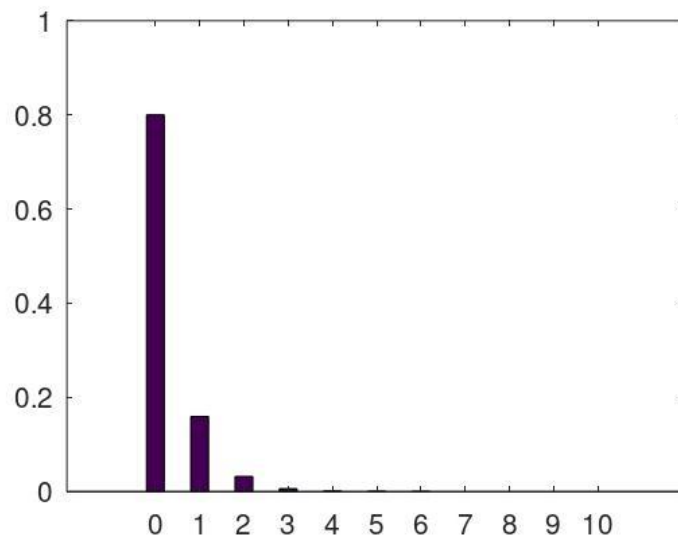


2.

- Προσομοίωση για $\lambda=1$

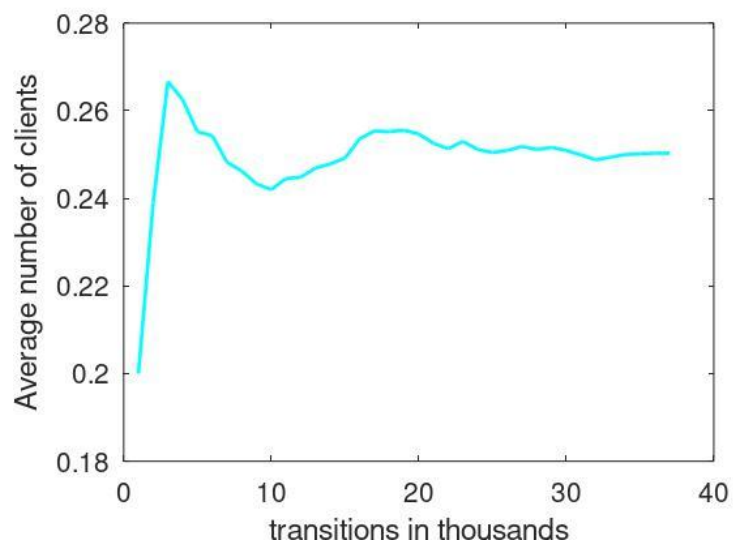
Γραφική παράσταση εργοδικών πιθανοτήτων:

Probabilities, lambda=1



Γραφική παράσταση εξέλιξης του μέσου αριθμού πελατών στο σύστημα:

e number of clients in the M/M/1 queue: Convergence,



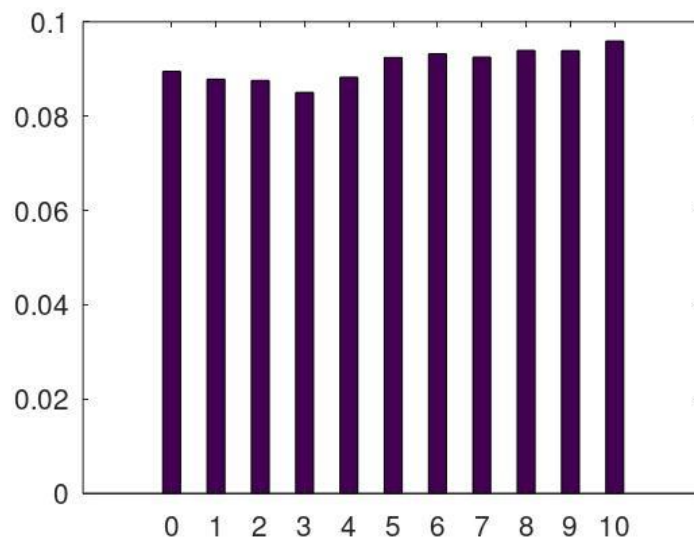
Κώδικας octave:

```

1 %2
2 %lambda=1
3
4 clc;
5 clear all;
6 close all;
7
8 rand("seed",1);
9
10 P = [0,0,0,0,0,0,0,0,0,0];
11 arrivals = [0,0,0,0,0,0,0,0,0,0];
12 total_arrivals = 0; % to measure the total number of arrivals
13 current_state = 0; % holds the current state of the system
14 previous_mean_clients = 0; % will help in the convergence test
15 index = 0;
16
17 lambda = 1;
18 mu = 5;
19 threshold = lambda/(lambda + mu); % the threshold used to calculate probabilities
20
21 transitions = 0; % holds the transitions of the simulation in transitions steps
22
23 while transitions >= 0
24     transitions = transitions + 1; % one more transitions step
25
26     if mod(transitions,1000) == 0 % check for convergence every 1000 transitions steps
27         index = index + 1;
28         for i=1:length(arrivals)
29             P(i) = arrivals(i)/total_arrivals; % calculate the probability of every state
30         endfor
31
32         mean_clients = 0; % calculate the mean number of clients in the system
33         for i=1:length(arrivals)
34             mean_clients = mean_clients + (i-1).*P(i);
35
36         endfor
37
38         to_plot(index) = mean_clients;
39
40         if abs(mean_clients - previous_mean_clients) < 0.00001 || transitions > 1000000 :
41             break;
42         endif
43
44         previous_mean_clients = mean_clients;
45
46     endif
47
48     random_number = rand(1); % generate a random number (Uniform distribution)
49     if current_state == 0 || random_number < threshold % arrival
50         if current_state < 11
51             total_arrivals = total_arrivals + 1;
52             arrivals(current_state + 1) = arrivals(current_state + 1) + 1; % increase the
53             if current_state < 10
54                 current_state = current_state + 1;
55             endif
56         endif
57     else % departure
58         if current_state != 0 % no departure from an empty system
59             current_state = current_state - 1;
60         endif
61     endif
62 endwhile
63
64 display("State propabilities:");
65 for i=1:length(arrivals)
66     display(P(i));
67 endfor
68
69
70 g = lambda*(1-P(11));
71 average_delay_time = mean_clients / g;
72 display("Average delay time =");
73 disp(average_delay_time);
74 display("Blocking propability =");
75 disp(P(11));
76
77 figure(1);
78 plot(to_plot,"c","linewidth",1.3);
79 title("Average number of clients in the M/M/1 queue: Convergence, lambda=1");
80 xlabel("transitions in thousands");
81 ylabel("Average number of clients");
82
83 x=[0,1,2,3,4,5,6,7,8,9,10];
84 figure(2);
85 bar(x,P,0.4);
86 title("Probabilities, lambda=1")
87 display(transitions);
88

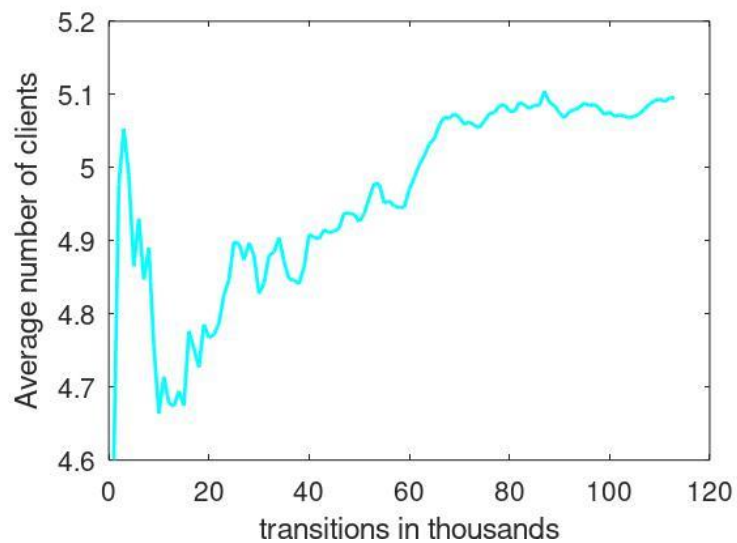
```

- Προσομοίωση για $\lambda=5$
Γραφική παράσταση εργοδικών πιθανοτήτων:
Probabilities, lambda=5



Γραφική παράσταση εξέλιξης του μέσου αριθμού πελατών στο σύστημα:

e number of clients in the M/M/1 queue: Convergence, l



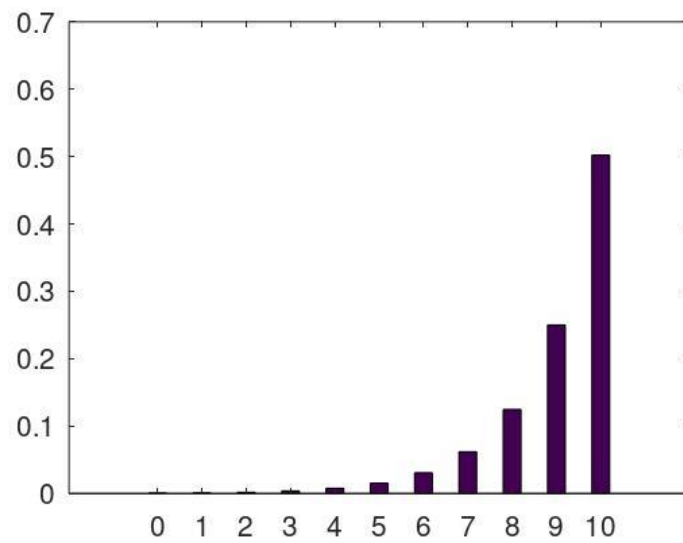
Κώδικας octave:


```

89 %λ=5
90 clc;
91 clear all;
92 close all;
93
94 rand("seed",1);
95
96 P = [0,0,0,0,0,0,0,0,0,0];
97 arrivals = [0,0,0,0,0,0,0,0,0,0];
98 total_arrivals = 0; % to measure the total number of arrivals
99 current_state = 0; % holds the current state of the system
100 previous_mean_clients = 0; % will help in the convergence test
101 index = 0;
102
103 lambda = 5;
104 mu = 5;
105 threshold = lambda/(lambda + mu); % the threshold used to calculate probabilities
106
107 transitions = 0; % holds the transitions of the simulation in transitions steps
108
109 while transitions >= 0
110     transitions = transitions + 1; % one more transitions step
111
112     if mod(transitions,1000) == 0 % check for convergence every 1000 transitions steps
113         index = index + 1;
114         for i=1:length(arrivals)
115             P(i) = arrivals(i)/total_arrivals; % calculate the probability of every state i
116         endfor
117
118         mean_clients = 0; % calculate the mean number of clients in the system
119         for i=1:length(arrivals)
120             mean_clients = mean_clients + (i-1).*P(i);
121         endfor
122
123         to_plot(index) = mean_clients;
124
125         if abs(mean_clients - previous_mean_clients) < 0.00001 || transitions > 1000000 %
126             break;
127         endif
128
129         previous_mean_clients = mean_clients;
130
131     endif
132
133     random_number = rand(1); % generate a random number (Uniform distribution)
134     if current_state == 0 || random_number < threshold % arrival
135         if current_state < 11
136             total_arrivals = total_arrivals + 1;
137             arrivals(current_state + 1) = arrivals(current_state + 1) + 1; % increase the nu
138             if current_state < 10
139                 current_state = current_state + 1;
140             endif
141         endif
142     else % departure
143         if current_state != 0 % no departure from an empty system
144             current_state = current_state - 1;
145         endif
146     endif
147 endwhile
148
149
150
151 display("State probabilities:");
152 for i=1:length(arrivals)
153     display(P(i));
154 endfor
155
156 g = lambda*(1-P(11));
157
158 average_delay_time = mean_clients / g;
159 display("Average delay time =");
160 disp(average_delay_time);
161 display("Blocking propability =");
162 disp(P(11));
163
164 figure(1);
165 plot(to_plot,"c","linewidth",1.3);
166 title("Average number of clients in the M/M/1 queue: Convergence, lambda=5");
167 xlabel("transitions in thousands");
168 ylabel("Average number of clients");
169
170 x=[0,1,2,3,4,5,6,7,8,9,10];
171 figure(2);
172 bar(x,P,0.4);
173 title("Probabilities, lambda=5");
174 display(transitions);

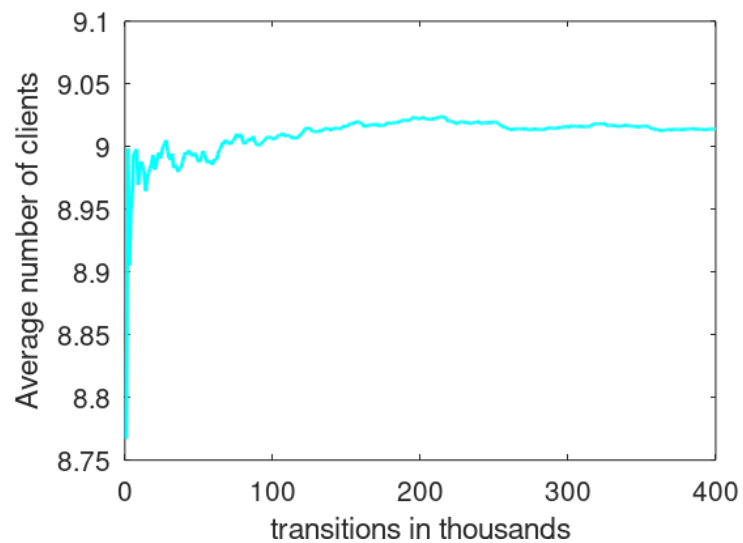
```

- Προσομοίωση για $\lambda=10$
Γραφική παράσταση εργοδικών πιθανοτήτων:
Probabilities, lambda=10



Γραφική παράσταση εξέλιξης του μέσου αριθμού πελατών στο σύστημα:

number of clients in the M/M/1 queue: Convergence, I



Κώδικας octave:

```

175 %lambda=10
176 clc;
177 clear all;
178 close all;
179
180 rand("seed",1);
181
182 P = [0,0,0,0,0,0,0,0,0,0];
183 arrivals = [0,0,0,0,0,0,0,0,0,0];
184 total_arrivals = 0; % to measure the total number of arrivals
185 current_state = 0; % holds the current state of the system
186 previous_mean_clients = 0; % will help in the convergence test
187 index = 0;
188
189 lambda = 10;
190 mu = 5;
191 threshold = lambda/(lambda + mu); % the threshold used to calculate probabilities
192
193 transitions = 0; % holds the transitions of the simulation in transitions steps
194
195 while transitions <= 1000000
196     transitions = transitions + 1; % one more transitions step
197
198     if mod(transitions,1000) == 0 % check for convergence every 1000 transitions steps
199         index = index + 1;
200         for i=1:length(arrivals)
201             P(i) = arrivals(i)/total_arrivals; % calculate the probability of every state i
202         endfor
203
204         mean_clients = 0; % calculate the mean number of clients in the system
205         for i=1:length(arrivals)
206             mean_clients = mean_clients + (i-1).*P(i);
207         endfor
208
209         to_plot(index) = mean_clients;
210
211         if abs(mean_clients - previous_mean_clients) < 0.00001 || transitions > 1000000 %
212             break;
213         endif
214
215         previous_mean_clients = mean_clients;
216
217     endif
218
219     random_number = rand(1); % generate a random number (Uniform distribution)
220     if current_state == 0 || random_number < threshold % arrival
221         if current_state < 11
222             total_arrivals = total_arrivals + 1;
223             arrivals(current_state + 1) = arrivals(current_state + 1) + 1; % increase the nu
224             if current_state < 10
225                 current_state = current_state + 1;
226             endif
227         endif
228     else % departure
229         if current_state != 0 % no departure from an empty system
230             current_state = current_state - 1;
231         endif
232     endif
233 endwhile
234
235
236
237 display("State probabilities:");
238 for i=1:length(arrivals)
239     display(P(i));
240 endfor
241
242 g = lambda*(1-P(11));
243
244 average_delay_time = mean_clients / g;
245 display("Average delay time =");
246 disp(average_delay_time);
247 display("Blocking probability =");
248 disp(P(11));
249
250 figure(1);
251 plot(to_plot,"c","linewidth",1.3);
252 title("Average number of clients in the M/M/1 queue: Convergence, lambda=10");
253 xlabel("transitions in thousands");
254 ylabel("Average number of clients");
255
256 figure(2);
257 bar(X,P,0.4);
258 title("Probabilities, lambda=10")
259 display(transitions);

```

3. Χρησιμοποιούμε την τελευταία εντολή του κάθε κώδικα για κάθε τιμή του λ για να υπολογίσουμε και να εμφανίσουμε το αποτέλεσμα των μεταβάσεων που χρειάζεται το σύστημα για να επέλθει σε ισορροπία. Προκύπτουν:

Για $\lambda=1$

```
transitions = 37000  
>> |
```

Για $\lambda=5$

```
transitions = 113000  
>> |
```

Για $\lambda=10$

```
transitions = 400000  
>> |
```

Παρατηρούμε ότι με την αύξηση του λ , η ταχύτητα σύγκλισης της προσομοίωσης μειώνεται, και άρα απαιτείται μεγαλύτερος αριθμός μεταβάσεων για την ικανοποίηση του κριτηρίου σύγκλισης. Αυτό είναι αναμενόμενο λόγω του τύπου $\rho = \frac{\lambda}{\mu}$, καθώς όσο αυξάνεται το λ και το μ παραμένει σταθερό τότε η ένταση του φορτίου ρ αυξάνεται. Αυτό σημαίνει ότι οι πελάτες φτάνουν με αυξημένο ρυθμό, ενώ το σύστημα διατηρεί τον ίδιο ρυθμό εξυπηρέτησης και άρα το σύστημα υπερφορτώνεται, κάτι το οποίο συνεπάγεται επίμηκυνση της μεταβατικής κατάστασης(της κατάστασης του συστήματος πριν επέλθει η εργοδική ισορροπία). Παρατηρώντας τα παραπάνω διαγράμματα καταλήγουμε στο ότι οι αρχικές μεταβάσεις που μπορούν να αγνοηθούν ώστε να επιταχυνθεί η σύγκλιση της προσομοίωσης είναι:

- Για $\lambda=1$ πελάτες/λεπτό, 25000 μεταβάσεις
- Για $\lambda=5$ πελάτες/λεπτό, 90000 μεταβάσεις
- Για $\lambda=10$ πελάτες/λεπτό, 200000 μεταβάσεις

4. Εάν ο αριθμός εξυπηρέτησης μ ήταν μεταβλητός, εξαρτημένος από την κατάσταση στην οποία βρίσκεται το σύστημα $i = \{1, 2, \dots, 10\}$ και ίσος με $\mu_i = \mu * (i + 1)$ τότε θα επιλέξουμε και μεταβλητό κατώφλι ίσο με $threshold_i = i = \frac{\lambda}{\lambda + \mu_i} = \frac{\lambda}{\lambda + \mu(i+1)}$ όπου i είναι η κατάσταση στην οποία βρισκόμαστε, όπου το i ανήκει στην μεταβλητή *current state*. Στον κώδικα μετά από κάθε αλλαγή της μεταβλητής *current state*, θα πρέπει να υπολογίζεται ξανά το μ ως $\mu = 5 * (current\ state + 1)$ καθώς και το *threshold*.