

# ΣΥΣΤΗΜΑΤΑ ΑΝΑΜΟΝΗΣ

ΝΤΟΝΤΟΡΟΣ ΗΛΙΑΣ

e119206

## ΑΣΚΗΣΗ 1<sup>η</sup>

Για να κάνουμε την προσομοίωση θα χρησιμοποιήσουμε κάποιες ενδεικτικές τιμές έστω  $\lambda=5$  και  $\mu=5$ . Χρησιμοποιώντας την `rand()` παράγουμε έναν τυχαίο αριθμό ο οποίος αν είναι μικρότερος από το `threshold` θα θεωρούμε ότι έχουμε άφιξη ενώ αν είναι μεγαλύτερος θα θεωρούμε ότι έχουμε αναχώρηση από το σύστημα. Ορίζουμε το `threshold` ως  $\lambda/(\lambda+\mu)$  δηλαδή το διάστημα που έχουμε άφιξη προς το διάστημα που πιθανών μεταβάσεων. Η προσομοίωση σταματάει όταν φτάσουμε στις 1.000.000 μεταβάσεις ή όταν η διαφορά ανάμεσα σε δύο διαδοχικούς μέσους όρους αριθμού πελατών είναι μικρότερη από 0.001%. Οι 30 πρώτες μεταβάσεις της προσομοίωσης φαίνονται παρακάτω:

```
1. Current state =
2. 0
3. Next transition
4. arrival
5. Total arrivals in current state =
6. 0
7. Current state =
8. 1
9. Next transition
10. arrival
11. Total arrivals in current state =
12. 0
13. Current state =
14. 2
15. Next transition
16. departure
17. Total arrivals in current state =
18. 0
19. Current state =
20. 1
21. Next transition
22. arrival
23. Total arrivals in current state =
24. 1
25. Current state =
26. 2
27. Next transition
28. departure
29. Total arrivals in current state =
30. 0
31. Current state =
32. 1
33. Next transition
34. departure
35. Total arrivals in current state =
36. 2
37. Current state =
38. 0
39. Next transition
40. arrival
```

41. Total arrivals in current state =  
42. 1  
43. Current state =  
44. 1  
45. Next transition  
46. arrival  
47. Total arrivals in current state =  
48. 2  
49. Current state =  
50. 2  
51. Next transition  
52. arrival  
53. Total arrivals in current state =  
54. 0  
55. Current state =  
56. 3  
57. Next transition  
58. departure  
59. Total arrivals in current state =  
60. 0  
61. Current state =  
62. 2  
63. Next transition  
64. departure  
65. Total arrivals in current state =  
66. 1  
67. Current state =  
68. 1  
69. Next transition  
70. departure  
71. Total arrivals in current state =  
72. 3  
73. Current state =  
74. 0  
75. Next transition  
76. arrival  
77. Total arrivals in current state =  
78. 2  
79. Current state =  
80. 1  
81. Next transition  
82. departure  
83. Total arrivals in current state =  
84. 3  
85. Current state =  
86. 0  
87. Next transition  
88. arrival  
89. Total arrivals in current state =  
90. 3  
91. Current state =  
92. 1  
93. Next transition  
94. departure  
95. Total arrivals in current state =  
96. 3  
97. Current state =  
98. 0  
99. Next transition  
100. arrival  
101. Total arrivals in current state =  
102. 4  
103. Current state =  
104. 1  
105. Next transition  
106. arrival  
107. Total arrivals in current state =  
108. 3  
109. Current state =  
110. 2  
111. Next transition  
112. arrival  
113. Total arrivals in current state =  
114. 1

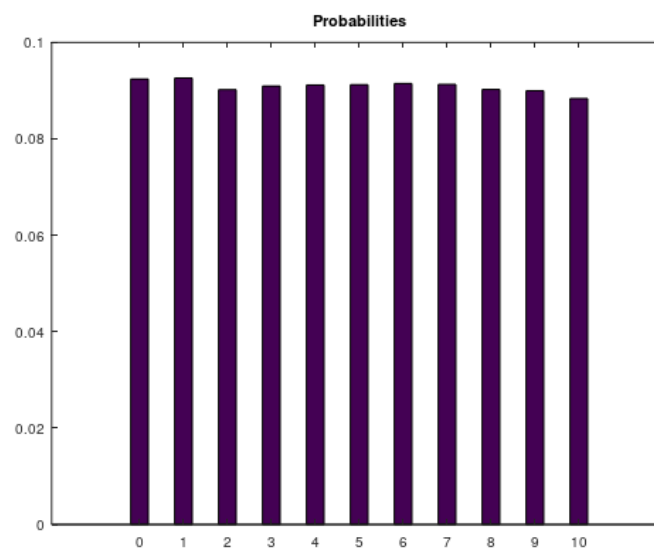
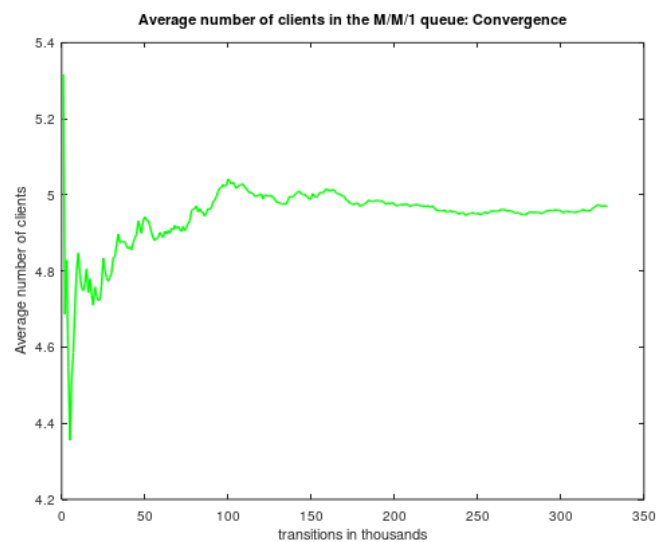
115.Current state =  
116.3  
117.Next transition  
118.departure  
119.Total arrivals in current state =  
120.0  
121.Current state =  
122.2  
123.Next transition  
124.arrival  
125.Total arrivals in current state =  
126.2  
127.Current state =  
128.3  
129.Next transition  
130.departure  
131.Total arrivals in current state =  
132.0  
133.Current state =  
134.2  
135.Next transition  
136.departure  
137.Total arrivals in current state =  
138.3  
139.Current state =  
140.1  
141.Next transition  
142.departure  
143.Total arrivals in current state =  
144.4  
145.Current state =  
146.0  
147.Next transition  
148.arrival  
149.Total arrivals in current state =  
150.5  
151.Current state =  
152.1  
153.Next transition  
154.departure  
155.Total arrivals in current state =  
156.4  
157.Current state =  
158.0  
159.Next transition  
160.arrival  
161.Total arrivals in current state =  
162.6  
163.Current state =  
164.1  
165.Next transition  
166.arrival  
167.Total arrivals in current state =  
168.4  
169.Current state =  
170.2  
171.Next transition  
172.departure  
173.Total arrivals in current state =  
174.3  
175.Current state =  
176.1  
177.Next transition  
178.arrival  
179.Total arrivals in current state =  
180.5

Οι πιθανότητες κατάστασης του συστήματος:

1. 0.092409
2. 0.092625
3. 0.090212
4. 0.090952
5. 0.091173
6. 0.091238
7. 0.091482
8. 0.091302
9. 0.090253
10. 0.089985
11. 0.088370
- 12.

Ο μέσος χρόνος καθυστέρησης και πιθανότητα Blocking :

1. Average delay time =
2. 1.0904
3. Blocking propability =
4. 0.088370
- 5.



Ο κώδικας που χρησιμοποιήθηκε :

```
1. # Task 1
2.
3. clc;
4. clear all;
5. close all;
6.
7. P = [0,0,0,0,0,0,0,0,0,0,0];
8. arrivals = [0,0,0,0,0,0,0,0,0,0,0];
9. total_arrivals = 0;
10. current_state = 0;
11. previous_mean_clients = 0;
12. index = 0;
13.
14. lamda = 5;
15. mu = 5;
16. threshold = lamda/(lamda + mu);
17.
18. transitions = 0;
19.
20. while transitions >= 0
21.     transitions = transitions + 1;
22.
23.     if mod(transitions,1000) == 0
24.         index = index + 1;
25.         for i=1:length(arrivals)
26.             P(i) = arrivals(i)/total_arrivals;
27.         endfor
28.
29.         mean_clients = 0;
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 > 1000000
37.             break;
38.         endif
39.
40.         previous_mean_clients = mean_clients;
41.
42.     endif
43.
44.     random_number = rand(1);
45.     if current_state == 0 || random_number < threshold
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;
57.             if current_state < 10
58.                 current_state = current_state + 1;
59.             endif
60.         endif
61.     else
62.         if current_state != 0
63.             if transitions < 31
```

```

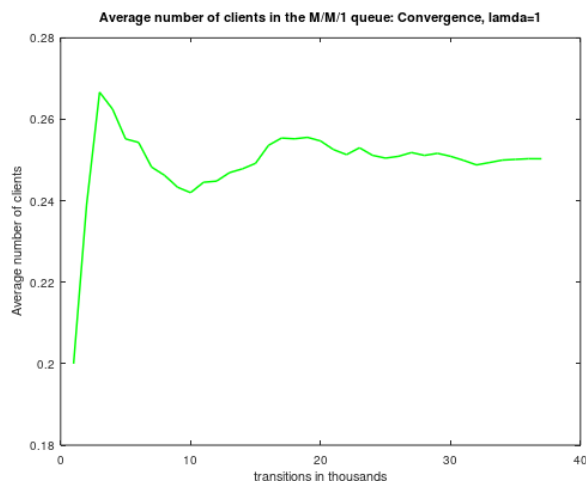
64.         display("Current state = ");
65.         disp(current_state);
66.         display("Next transition");
67.         display("departure");
68.         display("Total arrivals in current state = ");
69.         disp(arrivals(current_state+1));
70.     endif
71.     current_state = current_state - 1;
72. endif
73. endif
74. endwhile
75.
76.
77.
78. display("State propabilities:");
79. for i=1:length(arrivals)
80.     display(P(i));
81. endfor
82.
83. g = lamda*(1-P(11));
84. average_delay_time = mean_clients / g;
85. display("Average delay time =");
86. disp(average_delay_time);
87. display("Blocking propability =");
88. disp(P(11));
89.
90. figure(1);
91. plot(to_plot,"g","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")

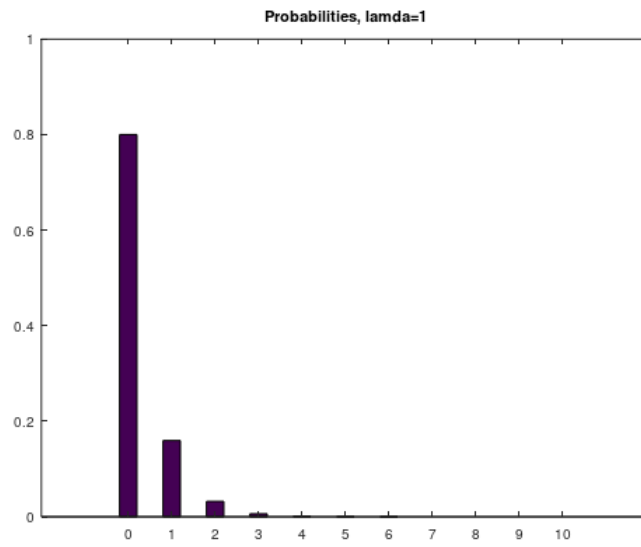
```

## ΑΣΚΗΣΗ 2<sup>η</sup>

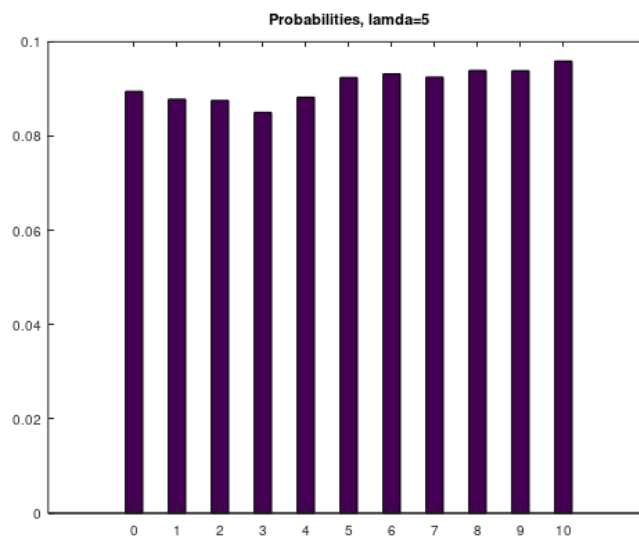
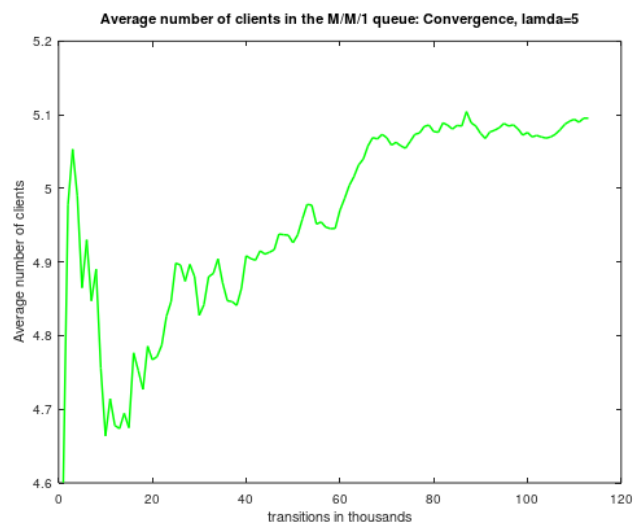
Εκτελούμε την παραπάνω προσομοίωση για  $\lambda=1, 5, 10$  και φτιάχνουμε τα διαγράμματα για τις εργοδίκες πιθανότητες και την εξέλιξη του μέσου αριθμού πελατών στο σύστημα :

I.  $\lambda=1$

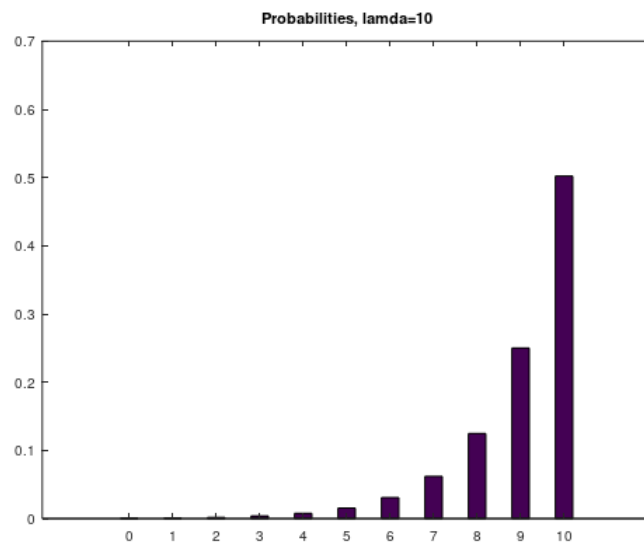
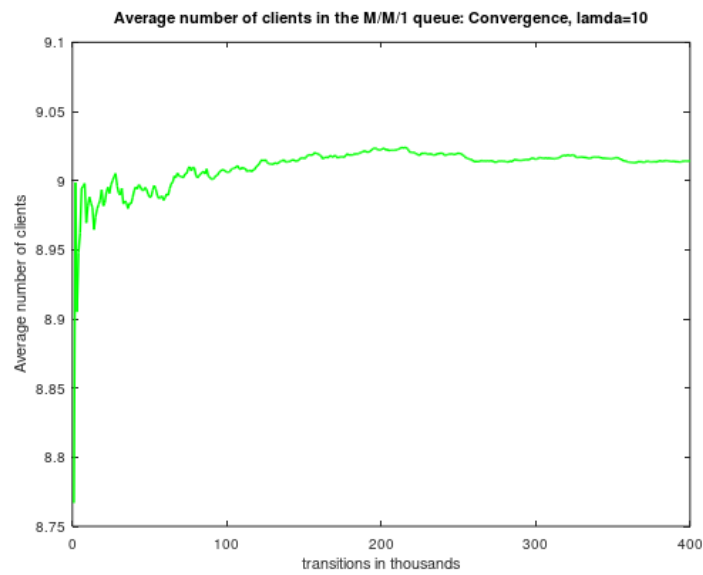




II.  $\lambda=5$



### III. $\lambda=10$



Ο κώδικας που χρησιμοποιήσαμε για τις προσομοιώσεις είναι ο ίδιος για όλες με μονή διαφορά την τιμή του  $\lambda$ :

```
1. # Task 2 (lamda= 1)
2.
3. clc;
4. clear all;
5. close all;
6.
7. rand("seed",1);
8.
9. P = [0,0,0,0,0,0,0,0,0,0,0];
10. arrivals = [0,0,0,0,0,0,0,0,0,0,0];
11. total_arrivals = 0;
12. current_state = 0;
```



```

13. previous_mean_clients = 0;
14. index = 0;
15.
16. lamda = 1;
17. mu = 5;
18. threshold = lamda/(lamda + mu);
19.
20. transitions = 0;
21.
22. while transitions >= 0
23.     transitions = transitions + 1;
24.
25.     if mod(transitions,1000) == 0
26.         index = index + 1;
27.         for i=1:length(arrivals)
28.             P(i) = arrivals(i)/total_arrivals;
29.         endfor
30.
31.         mean_clients = 0;
32.         for i=1:length(arrivals)
33.             mean_clients = mean_clients + (i-1).*P(i);
34.         endfor
35.
36.         to_plot(index) = mean_clients;
37.
38.         if abs(mean_clients - previous_mean_clients) < 0.00001 || transitions > 1000000
39.             break;
40.         endif
41.
42.         previous_mean_clients = mean_clients;
43.
44.     endif
45.
46.     random_number = rand(1);
47.     if current_state == 0 || random_number < threshold
48.         if current_state < 11
49.             total_arrivals = total_arrivals + 1;
50.             arrivals(current_state + 1) = arrivals(current_state + 1) + 1;
51.             if current_state < 10
52.                 current_state = current_state + 1;
53.             endif
54.         endif
55.     else
56.         if current_state != 0
57.             current_state = current_state - 1;
58.         endif
59.     endif
60. endwhile
61.
62.
63. figure(1);
64. plot(to_plot,"g","linewidth",1.3);
65. title("Average number of clients in the M/M/1 queue: Convergence, lamda=1");
66. xlabel("transitions in thousands");
67. ylabel("Average number of clients");
68.
69. x=[0,1,2,3,4,5,6,7,8,9,10];
70. figure(2);
71. bar(x,P,0.4);
72. title("Probabilities, lamda=1");
73. display(transitions);

```

### **ΑΣΚΗΣΗ 3<sup>η</sup>**

Παρατηρούμε ότι όσο αυξάνεται το  $\lambda$  η ταχύτητα της σύγκλισης μειώνεται δηλαδή χρειάζεται μεγαλύτερος αριθμός μεταβάσεων για να ικανοποιηθεί το κριτήριο σύγκλισης. Αυτό συμβαίνει γιατί ο αριθμός των αφίξεων αυξάνεται ενώ ο ρυθμός εξυπηρέτησης παραμένει σταθερός. Η τελευταία γραμμή του κώδικα του προηγούμενου ερωτήματος υπάρχει ώστε να εκτυπώνει πόσες μεταβάσεις έγιναν συνολικά και τα αποτελέσματα είναι :

$\lambda=1$  : transitions = 37000

$\lambda=5$  : transitions = 113000

$\lambda=10$  : transitions = 400000

Σύμφωνα με τα διαγράμματα του δεύτερου ερωτήματος μπορούμε να αγνοήσουμε 25000, 90000, 200000 μεταβάσεις για  $\lambda= 1, 5, 10$  αντίστοιχα.

### **ΑΣΚΗΣΗ 4<sup>η</sup>**

Αν ο ρυθμός εξυπηρέτησης ήταν συνάρτηση της κατάστασης που είναι το σύστημα σύμφωνα με την σχέση  $\mu_i = \mu * (i + 1)$  τότε και το threshold θα ήταν συνάρτηση της κατάστασης που βρίσκεται το σύστημα και ο τύπος θα ήταν :

$$threshold = \frac{\lambda}{\lambda + \mu_i}$$

Στην προσομοίωση θα έπρεπε μετά από την αλλαγή του current\_state να ξαναυπολογίσουμε το  $\mu$ , δηλαδή  $\mu = 5 * (current_{state} + 1)$  , και το threshold.