**ΣΥΣΤΗΜΑΤΑ ΑΝΑΜΟΝΗΣ**

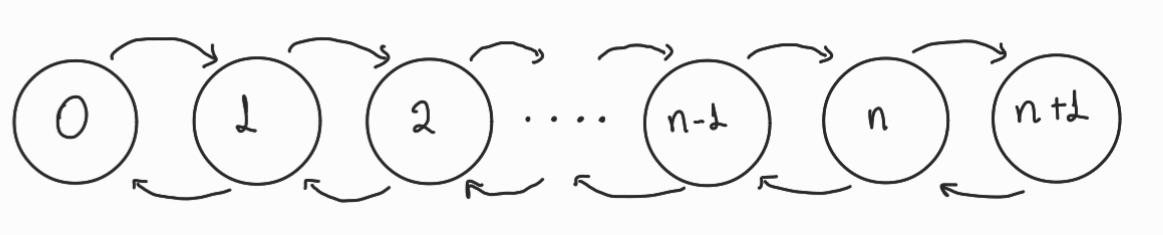
ΝΤΟΝΤΟΡΟΣ ΗΛΙΑΣ

el19206

**ΑΣΚΗΣΗ 1η**

Ερώτημα 1ο

Για να είναι μια ουρά Μ/Μ/1 εργοδική πρέπει η ένταση κυκλοφορίας του συστήματος ρ=λ/μ να είναι μικρότερη της μονάδας. H ένταση κυκλοφορίας ονομάζεται επίσης και βαθμός χρησιμοποίησης του εξυπηρετητή καθώς εκφράζει την πιθανότητα να μην είναι άδειο το σύστημα. Επομένως πρέπει να η ισχύει η παρακάτω συνθήκη (συνθήκη Erlang) :



Γενικά πρέπει να ισχύουν και οι εξισώσεις ισορροπίας για ένα μοντέλο αναμονής γεννήσεων-θανάτων :

Έτσι χρησιμοποιώντας τις παραπάνω εξισώσεις έχουμε :

Αλλά εφόσον έχουμε εργοδική ουρά ισχύει 0<ρ<1 άρα η σειρά που προκύπτει συγκλίνει στο άρα καταλήγουμε :

Δηλαδή έχουμε γεωμετρική κατανομή.

Ερώτημα 2ο

Σύμφωνα με τον τύπο του Little, ο μέσος χρόνος παραμονής στο σύστημα θα είναι:

Επίσης χρόνος παραμονής ενός πελάτη στο σύστημα είναι το άθροισμα του χρόνου αναμονής και του χρόνου εξυπηρέτησης. Άρα ο μέσος χρόνος αναμονής θα είναι:

Ερώτημα 3ο

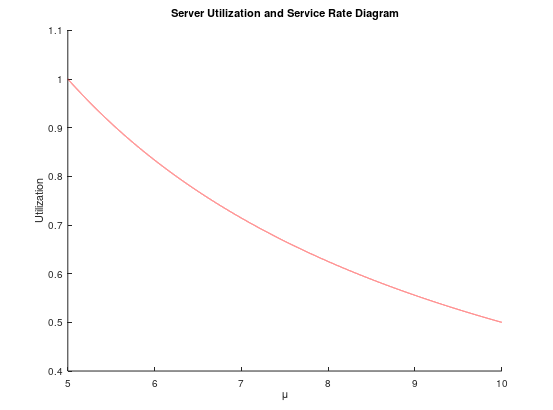
Γνωρίζουμε ότι ισχύει άρα για και για η Pk είναι θετική άρα το σύστημα μπορεί να βρεθεί με 57 πελάτες ωστόσο ό η πιθανότητα εξαρτάται από το ρ ().

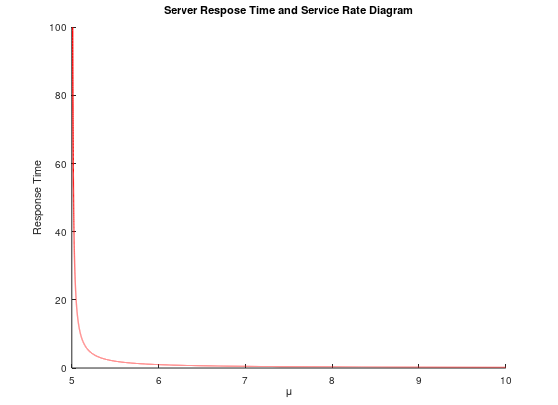
**ΑΣΚΗΣΗ 2η**

Ερώτημα 1ο

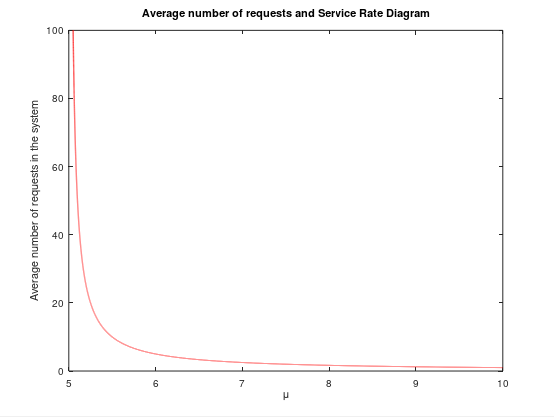
Για να είναι το σύστημα εργοδικό θα πρέπει να ισχύει άρα για πρέπει

Ερώτημα 2ο

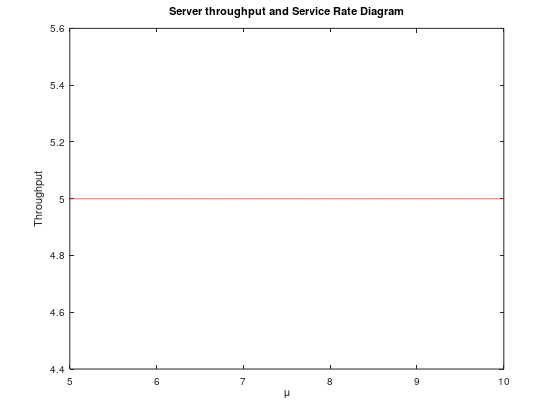
1. Βαθμός χρησιμοποίησης (utilization) ως προς το ρυθμό εξυπηρέτησης.
2. pkg load queueing;
4. lamda = 5;
5. mu = 5.0001 : 0.0001 : 10;
7. [U, R, Q, X, p0] = qsmm1 (lamda, mu);
8. # Task 1
9. figure(1);
10. hold on;
11. plot(mu,U,"r","linewidth",1.2);
12. hold off;
13. title("Server Utilization and Service Rate Diagram");
14. xlabel("\\mu");
15. ylabel("Utilization");
16. Μέσος χρόνος καθυστέρησης του συστήματος Ε(Τ) ως προς το ρυθμό εξυπηρέτησης.



1. # Task 2
2. figure(2);
3. hold on;
4. plot(mu,R,"r","linewidth",1.2);
5. axis([5 10 0 100]);
6. hold off;
7. title("Server Respose Time and Service Rate Diagram");
8. xlabel("\\mu");
9. ylabel("Response Time");
10. Μέσος αριθμός πελατών στο σύστημα ως προς το ρυθμό εξυπηρέτησης.



1. # Task 3
2. figure(3);
3. plot(mu,Q,"r","linewidth",1.2);
4. axis([5 10 0 100]);
5. title("Average number of requests and Service Rate Diagram");
6. xlabel("\\mu");
7. ylabel("Average number of requests in the system");
8. Ρυθμαπόδοση (throughput) πελατών ως προς το ρυθμό εξυπηρέτησης



1. # Task 4
2. figure(4);
3. plot(mu,X,"r","linewidth",1.2);
4. title("Server throughput and Service Rate Diagram");
5. xlabel("\\mu");
6. ylabel("Throughput");

Ερώτημα 3ο

Από το διάγραμμα του μέσου χρόνου καθυστέρησης παρατηρούμε ότι ο χρόνος καθυστέρησης είναι αντιστρόφως ανάλογος από τον ρυθμό εξυπηρέτησης και ότι από ένα σημείο και μετά η διάρκεια του χρόνου καθυστέρησης είναι αρκετά μικρή σχεδόν σταθερή. Οπότε η καλύτερη επιλογή θα ήταν περίπου στο ώστε να έχουμε γρήγορο χρόνο εξυπηρέτησης χωρίς το κόστος που θα δαπανούσαμε αν επιλέγαμε .

Ερώτημα 4ο

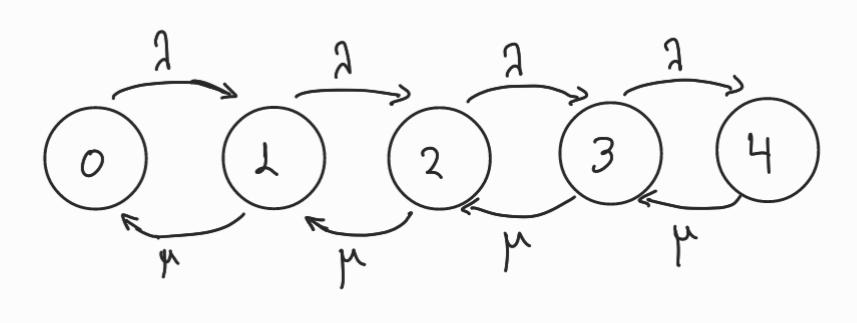
Στο συγκεκριμένο σύστημα Μ/Μ/1 υπάρχει άπειρη χωρητικότητα πελατών επόμενος η πιθανότητα να απορριφθεί κάποιος πελάτης είναι 0. Άρα επομένως από τον τύπο της ρυθμαπόδοσης :

εχουμε ότι , όπως φαίνεται και στο διάγραμμα

**ΑΣΚΗΣΗ 3η**

Ερώτημα 1ο

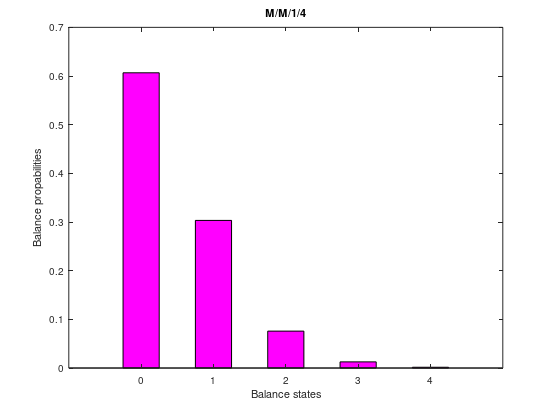
Το διάγραμμα γεννήσεων-θανάτων του συστήματος Μ/Μ/1/4 μπορεί να αναπαρασταθεί από τη παρακάτω εικόνα :

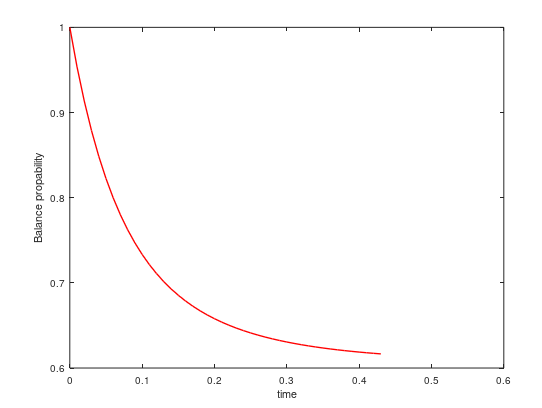


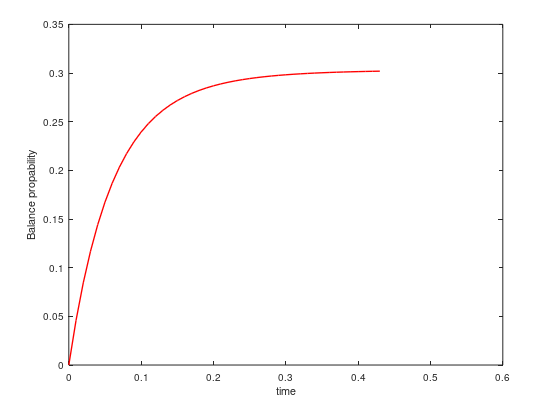
Για και από τις εξισώσεις ισορροπίας και την συνθήκη κανονικοποίησης μπορούμε να υπολογίσουμε τις εργοδικές πιθανότητες ως εξής :

Επίσης η τελευταία πιθανότητα είναι ίση με την πιθανότητα απόρριψης πελάτη καθώς όταν το σύστημα βρίσκεται σε αυτή την κατάσταση δεν μπορεί να δεχτεί επιπλέων πελάτες άρα .

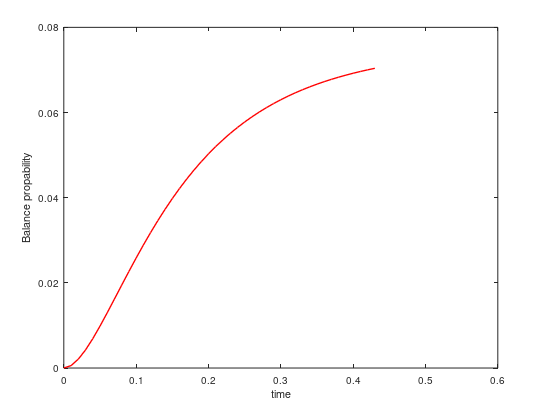
Ερώτημα 2ο

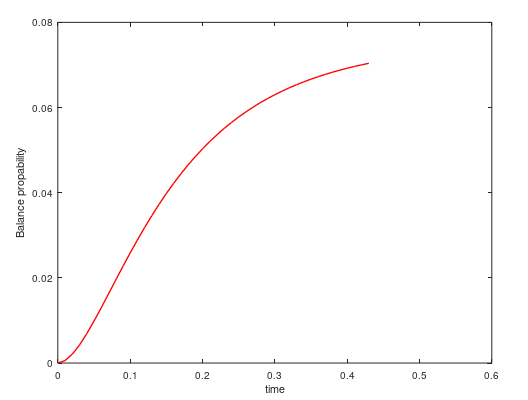
1. Μήτρα ρυθμού μεταβάσεων :
2. -5.0000 5.0000 0 0 0
3. 10.0000 -12.5000 2.5000 0 0
4. 0 10.0000 -11.6667 1.6667 0
5. 0 0 10.0000 -11.2500 1.2500
6. 0 0 0 10.0000 -10.0000
7. Υπολογισμός εργοδικών πιθανοτήτων μέσω Octave :
8. 0.6066
9. 0.3033
10. 0.075829
11. 0.012638
12. 1.5798e-03
13. Μέσος αριθμός πελατών στην κατάσταση ισορροπίας :
14. mean\_clients = 0.4992
15. Πιθανότητα απόρριψης πελάτη στην κατάσταση ισορροπίας :
16. Blocking probability =
17. 1.5798e-03
18. Διαγράμματα των πιθανοτήτων των καταστάσεων του συστήματος σαν συναρτήσεις του χρόνου από την αρχική κατάσταση.

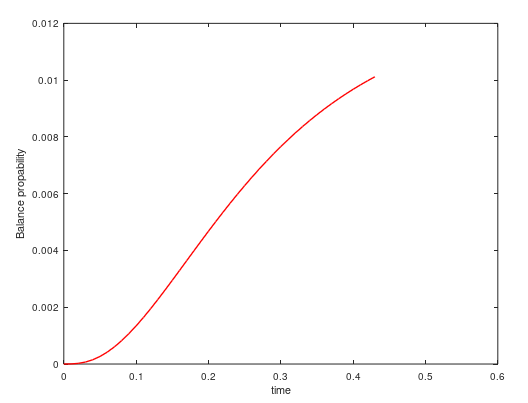
State 0:

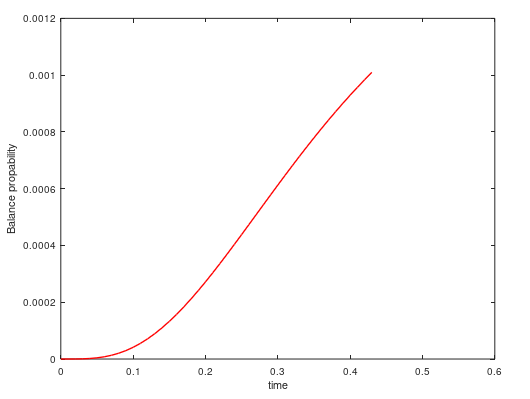


State 1:

State 2 :

State 3 :

State 4 :

State 5 :

1. clc;
2. clear all;
3. close all;
5. pkg load queueing;
6. # Task 2.1
7. lamda = 5;
8. mu = 10;
9. states = [0, 1, 2, 3, 4];
10. initial\_state = [1, 0, 0, 0, 0];
11. births\_B = [lamda, lamda/2, lamda/3, lamda/4];
12. deaths\_D = [mu, mu, mu, mu];
14. transition\_matrix = ctmcbd(births\_B, deaths\_D);
15. display(transition\_matrix);
17. # Task 2.2
18. P = ctmc(transition\_matrix);
19. for i=1:5
20. display(P(i));
21. endfor
22. figure(1);
23. bar(states, P, "m", 0.5);
24. title("M/M/1/4");
25. xlabel("Balance states");
26. ylabel("Balance propabilities");
28. # Task 2.3
29. mean\_clients = 0;
30. for i = 1 : 1 : 5
31. mean\_clients = mean\_clients + P(i)\*(i-1);
32. endfor
33. display(mean\_clients);
35. # Task 2.4
36. display("Blocking probability = ");
37. display(P(5));
39. # Task 2.5
40. for j = 1:1:5
41. index = 0;
42. for T = 0 : 0.01 : 50
43. index = index + 1;
44. P0 = ctmc(transition\_matrix, T, initial\_state);
45. Prob0(index) = P0(j);
46. if P0 - P < 0.01
47. break;
48. endif
49. endfor
51. t = 0 : 0.01 : T;
52. figure(j+1);
53. plot(t, Prob0, "r", "linewidth", 1.2);
54. xlabel("time");
55. ylabel("Balance propability");
56. Endfor