Συστήματα Αναμονής
2η εργαστηριακή άσκηση

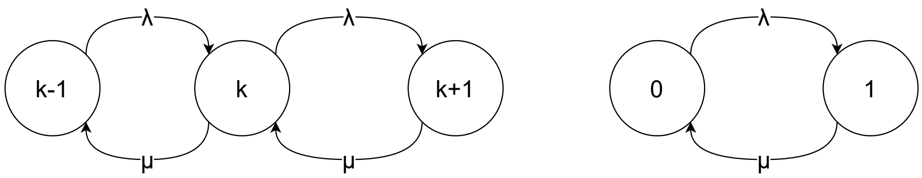
Νικόλαος Παγώνας, el18175

Θεωρητική μελέτη της ουράς M/M/1

(α)

Προκειμένου να είναι εργοδική η ουρά M/M/1, είναι απαραίτητο να ισχύει $\rho = \dfrac{\lambda}{\mu} < 1$, όπου λ ο ρυθμός αφίξεων, μ ο ρυθμός εξυπηρέτησης και ρ η ένταση φορτίου.

Σχεδιάζουμε το διάγραμμα ρυθμού μεταβάσεων για την ουρά M/M/1:



Για τον υπολογισμό των εργοδικών πιθανοτήτων των καταστάσεων του συστήματος από τις εξισώσεις ισορροπίας έχουμε:

$$(\lambda + \mu)P_k = \lambda P_{k-1} + \mu P_{k+1} \text{ για } k \geq 1 \text{ και } \lambda P_0 = \mu P_1$$

οπότε από την δεύτερη εξίσωση παίρνουμε:

$$P_1 = \left(\frac{\lambda}{\mu}\right) P_0 = \rho P_0$$

Αντικαθιστώντας στην πρώτη (για $k = 1$) έχουμε:

$$(\lambda + \mu)P_1 = \lambda P_0 + \mu P_2 \implies P_2 = \frac{\lambda}{\mu} P_1 = \left(\frac{\lambda}{\mu}\right)^2 P_0 = \rho^2 P_0$$

και γενικά:

$$P_k = \rho^k P_0, \quad k > 0$$

Επειδή επιπλέον ισχύει η ιδιότητα της κανονικοποίησης:

$$\sum_{k=0}^{\infty} P_k = 1 \implies \sum_{k=0}^{\infty} \rho^k P_0 = 1 \implies P_0 \cdot \frac{1}{1-\rho} = 1 \text{ (αφού } \rho < 1)$$

και έτσι:

$$P_0 = 1 - \rho, \; P_k = (1-\rho)\rho^k, \text{ για } k > 0$$

(β)

Θεωρούμε γνωστή την ισότητα της εκφώνησης (έχει αποδειχθεί και στο μάθημα) και χρησιμοποιούμε τον νόμο του Little:

$$E[T] = \frac{E[n(t)]}{\gamma} = \frac{E[n(t)]}{\lambda} = \frac{\rho}{(1-\rho)\lambda} = \frac{\lambda/\mu}{\lambda(1-\lambda/\mu)} = \frac{1}{\mu - \lambda}$$

(γ)

Η ουρά M/M/1 χαρακτηρίζεται από άπειρη χωρητικότητα και απείρως επισκέψιμες επαναληπτικές καταστάσεις (positive recurrent states), με μη μηδενικές εργοδικές πιθανότητες $P_k(t) \to P_k > 0, \; k = 0, 1, 2, ...$

Αυτό σημαίνει ότι (σε άπειρο χρόνο) θα συναντήσουμε όλες τις καταστάσεις, και από άπειρες φορές. Επομένως, αφού η κατάσταση 57 ανήκει στην απειρία αυτή, τότε θα υπάρξει χρονική στιγμή που το σύστημα θα βρεθεί με 57 πελάτες.

Ανάλυση ουράς M/M/1 με Octave

(α)

Προκειμένου το σύστημα να είναι εργοδικό, θα πρέπει:

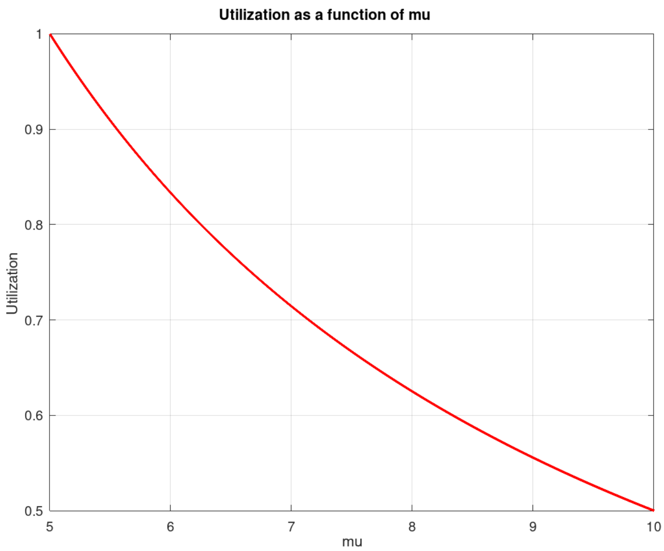
$$\rho = \lambda/\mu < 1 \implies \mu > \lambda \implies \mu > 5 \; \frac{\text{πελάτες}}{\text{min}}$$

(β)

Με τη βοήθεια της συνάρτησης qsmm1 του πακέτου queueing του Octave απεικονίζουμε τα παρακάτω 4 διαγράμματα, για τις επιτρεπόμενες τιμές του μ, δηλαδή $\mu \in (5, 10]$:

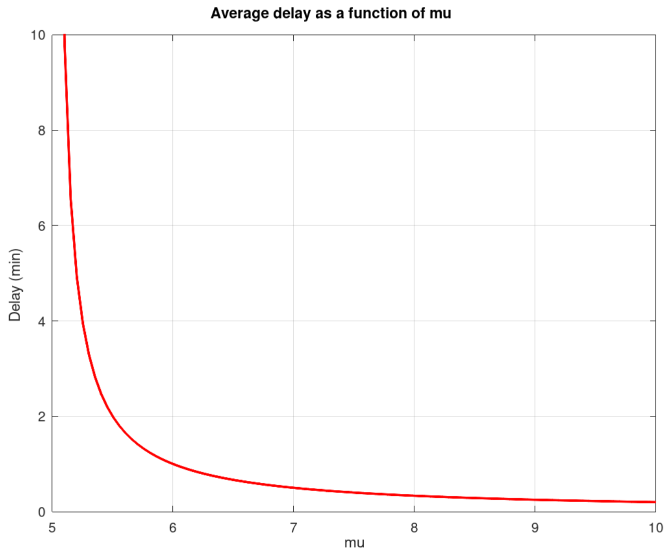
1. Βαθμός χρησιμοποίησης (utilization) ως προς το ρυθμό εξυπηρέτησης:

$$u = \frac{\gamma}{\mu} = \frac{\lambda}{\mu}$$



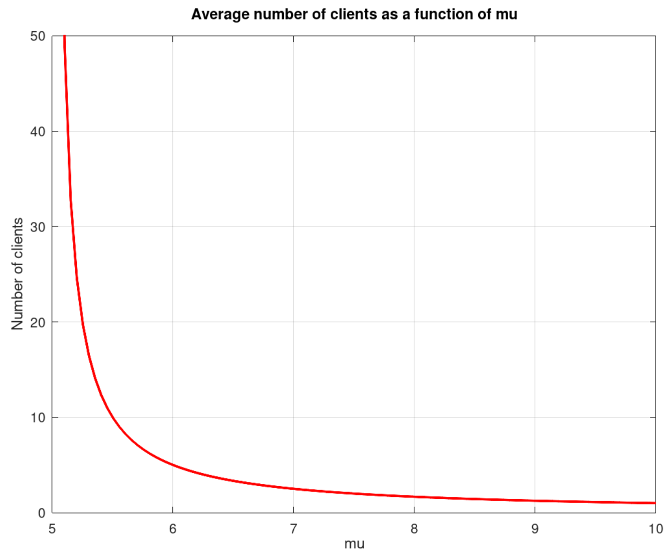
2. Μέσος χρόνος καθυστέρησης του συστήματος E[T] ως προς το ρυθμό εξυπηρέτησης:

$$E[T] = \frac{1}{\mu - \lambda}$$



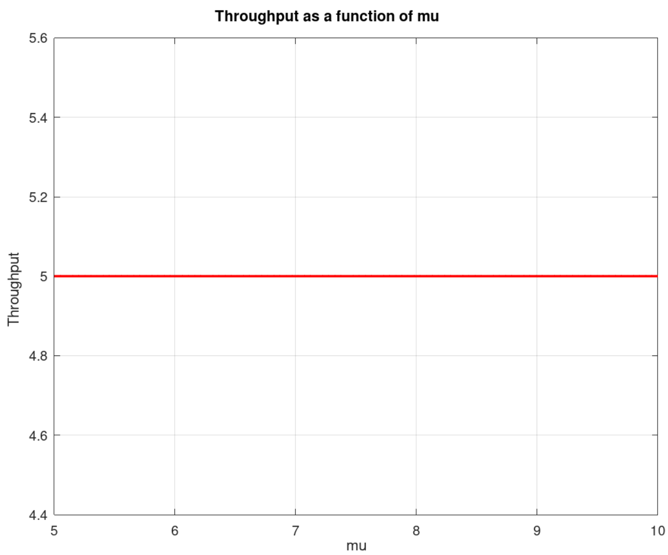
3. Μέσος αριθμός πελατών στο σύστημα ως προς το ρυθμό εξυπηρέτησης:

$$E[n(t)] = \frac{\rho}{1 - \rho}$$



4. Ρυθμαπόδοση (throughput) ως προς το ρυθμό εξυπηρέτησης:

$$\gamma = \lambda\,(1 - P[blocking]) = \lambda$$



(γ)

Παρατηρώντας το διάγραμμα του μέσου χρόνου καθυστέρησης βλέπουμε, όπως είναι λογικό, ότι όσο αυξάνεται ο ρυθμός εξυπηρέτησης, τόσο ο χρόνος καθυστέρησης μειώνεται. Όμως, η επίδραση που έχει ο ρυθμός εξυπηρέτησης είναι πολύ πιο μεγάλη για τιμές κοντά στο $\mu = 5$, και μειώνεται όσο το μ πλησιάζει στο 10. Έτσι, μπορεί ένας εξυπηρετητής με $\mu = 10$ να είναι η ιδανική περίπτωση, όμως αν οι απαιτήσεις μας δεν είναι τόσο μεγάλες, αποτελεί σπατάλη πόρων. Για παράδειγμα, πιο λογική επιλογή θα ήταν να διαλέξουμε έναν εξυπηρετητή με $\mu = 6$, ώστε να αποφύγουμε μεν το κατακόρυφο τμήμα της συνάρτησης-υπερβολής (οπότε επιτυγχάνουμε μέση καθυστέρηση 1 min), αλλά και ταυτόχρονα να μην επιλέξουμε έναν αδικαιολόγητα ακριβό εξυπηρετητή.

(δ)

Παρατηρούμε ότι το throughput πελατών σε μία ουρά M/M/1 είναι σταθερό και ίσο με λ, ανεξάρτητο του μ. Αυτό ισχύει διότι, λόγω της άπειρης χωρητικότητας, δεν υπάρχει περίπτωση να απορρίψουμε κανέναν πελάτη, και άρα

$$P[blocking] = 0.$$

Έτσι,

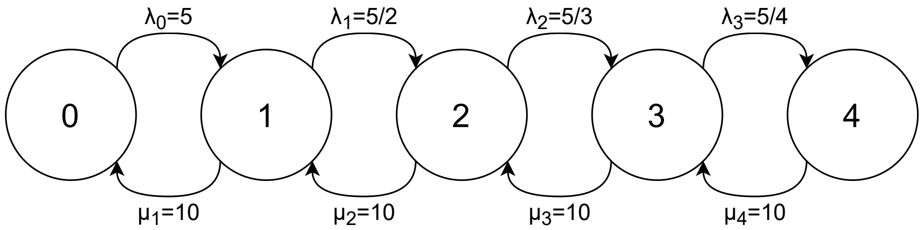
$$\gamma = \lambda(1 - P[blocking]) = \lambda.$$

Διαδικασία γεννήσεων θανάτων (birth-death process): εφαρμογή σε σύστημα M/M/1/K

(α)

Μοντελοποιούμε το σύστημα της εκφώνησης ως διαδικασία γεννήσεων-θανάτων.

Το διάγραμμα ρυθμού μεταβάσεων είναι το εξής:



Χρησιμοποιώντας τις τοπικές εξισώσεις ισορροπίας έχουμε:

$$\lambda_0 P_0 = \mu_1 P_1 \implies P_1 = \frac{\lambda_0}{\mu_1} P_0 = \frac{P_0}{2}$$

$$\lambda_1 P_1 = \mu_2 P_2 \implies P_2 = \frac{\lambda_1}{\mu_2} P_1 = \frac{P_0}{8}$$

$$\lambda_2 P_2 = \mu_3 P_3 \implies P_3 = \frac{\lambda_2}{\mu_3} P_2 = \frac{P_0}{48}$$

$$\lambda_3 P_3 = \mu_4 P_4 \implies P_4 = \frac{\lambda_3}{\mu_4} P_3 = \frac{P_0}{384}$$

$$P_0 + P_1 + P_2 + P_3 + P_4 = 1 \implies P_0 = \frac{1}{1 + 1/2 + 1/8 + 1/48 + 1/384}$$

οπότε:

$$P_0 = 60.66\%, P_1 = 30.33\%, P_2 = 7.58\%, P_3 = 1.26\%, P_4 = 0.16\%$$

και έτσι:

$$P[blocking] = P_4 = 0.16\%$$

(β)

(i) Η μήτρα ρυθμού μεταβάσεων είναι η εξής:

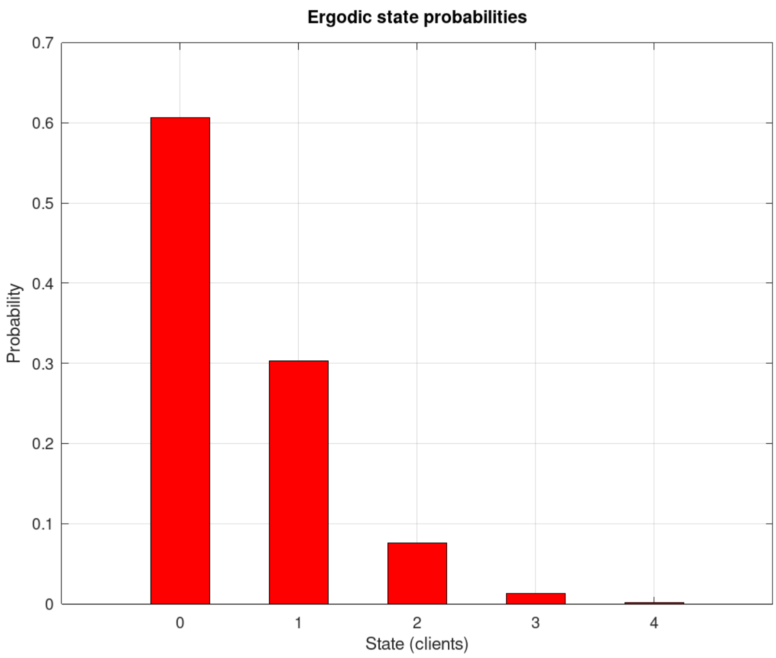
```
transition_matrix =

   -5.00000    5.00000    0.00000    0.00000    0.00000
   10.00000  -12.50000    2.50000    0.00000    0.00000
    0.00000   10.00000  -11.66667    1.66667    0.00000
    0.00000    0.00000   10.00000  -11.25000    1.25000
    0.00000    0.00000    0.00000   10.00000  -10.00000
```

(ii) Με την εντολή ctmc βρίσκουμε τις εργοδικές πιθανότητες των καταστάσεων του συστήματος:

```
P[1] = 60.6635%
P[2] = 30.3318%
P[3] = 7.58294%
P[4] = 1.26382%
P[5] = 0.157978%
```

Οι παραπάνω πιθανότητες συμφωνούν με τους θεωρητικούς υπολογισμούς του ερωτήματος (α).



(iii) Ο μέσος αριθμός πελατών στο σύστημα όταν αυτό βρίσκεται σε κατάσταση ισορροπίας υπολογίζεται από τη σχέση

$$\sum_{k=0}^{5} kP_k$$

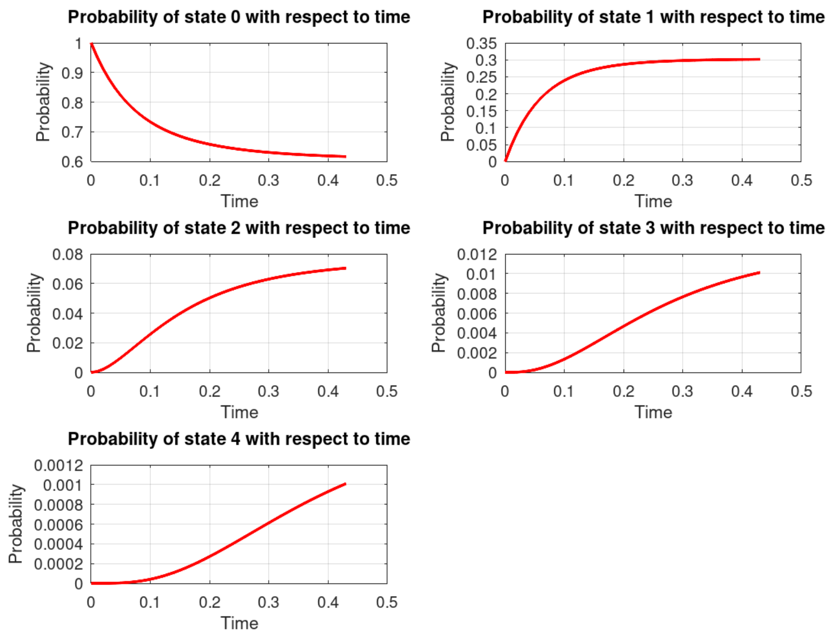
Ο παραπάνω τύπος δίνει:

```
Average number of clients = 0.49921
```

(iv) Η πιθανότητα απόρριψης πελάτη όταν το σύστημα βρίσκεται σε κατάσταση ισορροπίας είναι ίση με την πιθανότητα της τελευταίας κατάστασης, δηλαδή την P_4. Έτσι:

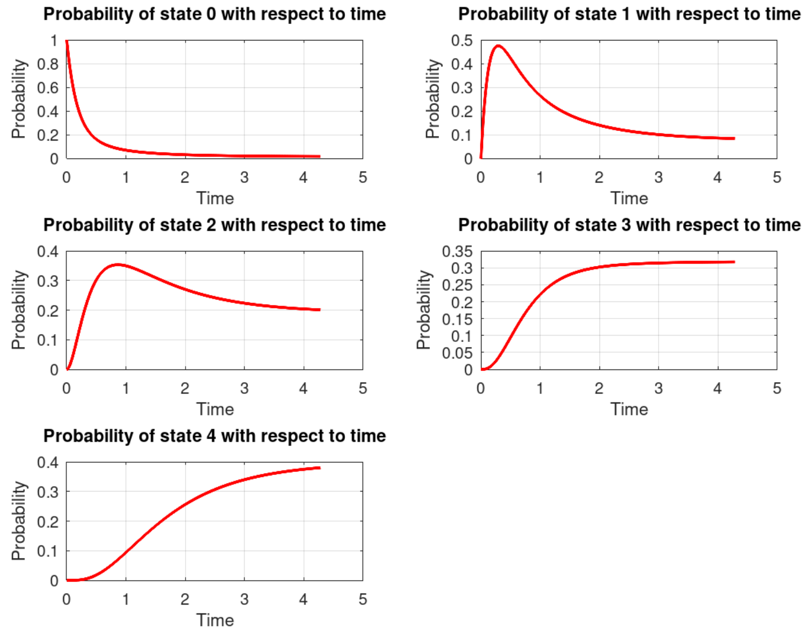
```
P[blocking] = 0.157978%
```

(v) Δημιουργούμε τα ζητούμενα διαγράμματα, τα οποία απεικονίζουν την εξέλιξη των πιθανοτήτων για κάθε κατάσταση συναρτήσει του χρόνου, από την αρχική κατάσταση μέχρι οι πιθανότητες να απέχουν λιγότερο από 1% σε σχέση με τις εργοδικές πιθανότητες του ερωτήματος (2):

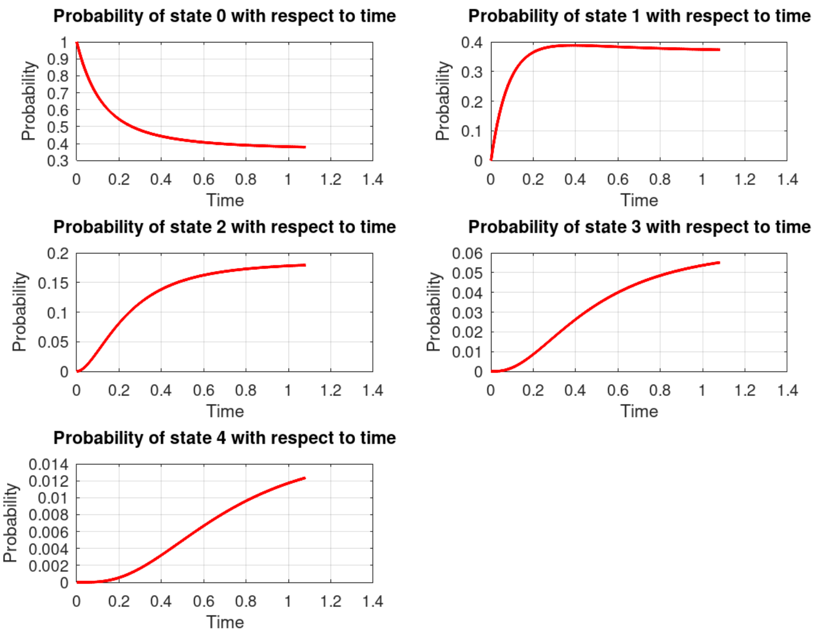


(vi) Επαναλαμβάνουμε για:

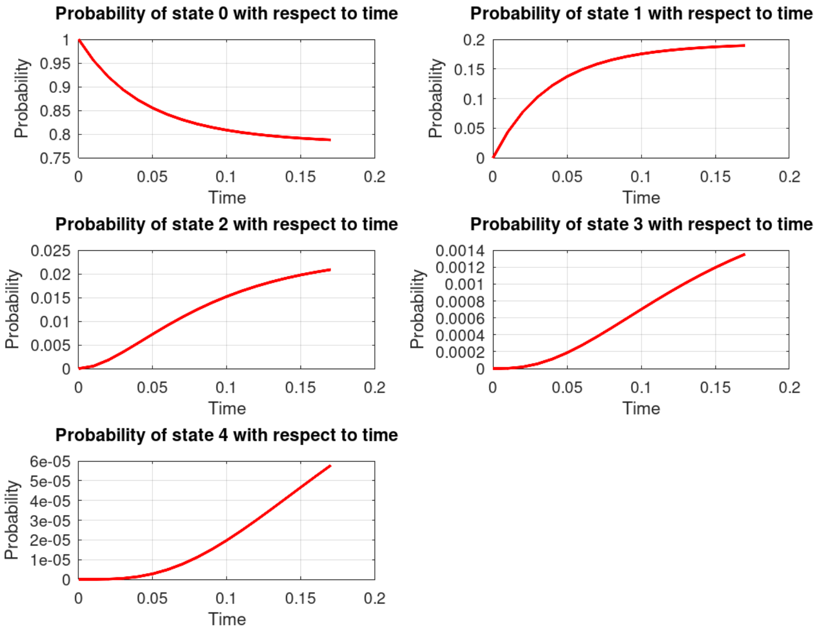
1. $\lambda = 5$, $\mu = 1$



2. $\lambda = 5$, $\mu = 5$



3. $\lambda = 5$, $\mu = 20$



Ποιοτικά, παρατηρούμε ότι όσο μεγαλώνει η τιμή της παραμέτρου μ:

- Οι πιθανότητες συγκλίνουν γρηγορότερα στις αντίστοιχες εργοδικές.
- Οι καταστάσεις με λιγότερους πελάτες έχουν όλο και μεγαλύτερη πιθανότητα να συμβούν. Αυτό συμβαίνει διότι οι πελάτες, οι οποίοι σε όλες τις περιπτώσεις έρχονται με σταθερό ρυθμό, εξυπηρετούνται γρηγορότερα, οπότε το σύστημα αδειάζει συχνότερα.

Παράρτημα κώδικα

Ανάλυση ουράς M/M/1 με Octave

```
pkg load queueing;
clc;
clear all;
close all;

if !(exist("files","dir") == 7)
  mkdir "files";
endif

samples = 100;
threshold = 5;
small = 0.001;
```

```
mu_start = threshold+small;
lambda = linspace(5,5,samples);
mu = linspace(mu_start,10,samples);

[utilization, response, requests, throughput] = qsmm1(lambda, mu);

y = [utilization; response; requests; throughput];
titles = ["Utilization as a function of mu";
          "Average delay as a function of mu";
          "Average number of clients as a function of mu";
          "Throughput as a function of mu"];
xlab = ["mu"; "mu"; "mu"; "mu"];
ylab = ["Utilization";
        "Delay (min)";
        "Number of clients";
        "Throughput"];

for i=1:4
  figure(i);
  plot(mu,y(i,:),"linewidth",2,"r");
  grid on;
  title(titles(i,:));
  xlabel(xlab(i,:));
  ylabel(ylab(i,:));
  if i == 2
    ylim([0,10]);
  endif
  if i == 3
    ylim([0,50]);
  endif
  saveas(i,strcat("files/2b",num2str(i),".png"));
endfor
```

Διαδικασία γεννήσεων θανάτων: εφαρμογή σε σύστημα M/M/1/K

```
% system M/M/1/4

clc;
clear all;
close all;
pkg load queueing;

if !(exist("files","dir") == 7)
  mkdir "files";
endif

fd = fopen("files/3bi.txt","w");
fclose(fd);

nfig = 0;
lambdas = [5,5,5,5];
mus = [10,1,5,20];

for iteration = 1:4
  lambda = lambdas(iteration);
  mu = mus(iteration);
  states = [0, 1, 2, 3, 4]; % system with capacity 4 states
  % the initial state of the system. The system is initially empty.
  initial_state = [1, 0, 0, 0, 0];

  % define the birth and death rates between the states of the
    system.
```

```
births_B = [lambda/1, lambda/2, lambda/3, lambda/4];
deaths_D = [mu, mu, mu, mu];

% (i) get the transition matrix of the birth-death process
transition_matrix = ctmcbd(births_B, deaths_D);
if iteration == 1
  diary "files/3bi.txt"
  transition_matrix
  diary off
endif

% get the ergodic probabilities of the system
P = ctmc(transition_matrix);

if iteration == 1
  % (ii) plot the ergodic probabilities (bar for bar chart)
  fd = fopen("files/3bii.txt","w");
  for i = 1:5
    fprintf(fd,"P[%d] = %d%%\n",i,100*P(i));
  endfor
  fclose(fd);

  nfig = nfig + 1;
  figure(nfig);
  bar(states, P, "r", 0.5);
  grid on;
  title("Ergodic state probabilities");
  xlabel("State (clients)");
  ylabel("Probability");

  saveas(nfig,"files/3bii.png");

  % (iii)
  average_clients = 0*P(1)+1*P(2)+2*P(3)+3*P(4)+4*P(5);

  fd = fopen("files/3biii.txt","w");
  fprintf(fd, "Average number of clients = %d", average_clients);
  fclose(fd);

  % (iv)
  fd = fopen("files/3biv.txt","w");
  fprintf(fd, "P[blocking] = %d%%\n", 100*P(5));
  fclose(fd);
endif

% (v) transient probability of all states until convergence to
  ergodic probability. Convergence takes place P0 and P differ by
   0.01

index = 0;
maxT = 0;

for T = 0 : 0.01 : 50
  index = index + 1;
  maxT = T;
  P0 = ctmc(transition_matrix, T, initial_state);
  for i = 1:5
    Prob0(index,i) = P0(i);
  endfor
  if P0 - P < 0.01
    break;
  endif
```

```
  endfor

  T = 0 : 0.01 : maxT;
  nfig = nfig + 1;
  figure(nfig);
  for i = 1:5
    subplot(3,2,i);
    plot(T, Prob0(1:index,i), "r", "linewidth", 2);

    grid on;
    title(strcat("Probability of state",{' '},num2str(i-1)," with
    respect to time"));
    xlabel("Time");
    ylabel("Probability");
  endfor
  if iteration == 1
    saveas(nfig,strcat("files/3bv.png"));
  else
    saveas(nfig,strcat("files/3bvi_",num2str(iteration-1),".png"));
  endif
endfor
```