ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

Μάθημα: Συστήματα Αναμονής

Ονοματεπώνυμο: Ειρήνη Δόντη

Α.Μ: 03119839

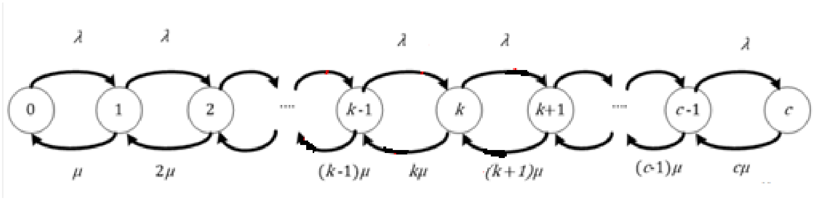
4η Ομάδα Ασκήσεων

Αθήνα 2022

Ανάλυση και Σχεδιασμός Τηλεφωνικού Κέντρου

(1)

Το διάγραμμα ρυθμού μεταβάσεων του συστήματος M/M/c/c είναι το ακόλουθο:



Από τις εξισώσεις ισορροπίας, γνωρίζουμε ότι:

$$P_k = (\frac{\lambda}{k\mu})P_{k-1} = (\frac{\rho^k}{k!})P_0, k = 1,2, \ldots, c \ \mu\varepsilon \ \rho = \frac{\lambda}{\mu} \ (1)$$

$$P_0 + P_1 + \cdots + P_c = 1 \ (2) \ ή$$

Από σχέση (1): $P_k = (\frac{\lambda}{k\mu})P_{k-1} = (\frac{\lambda}{k\mu})[(\frac{\lambda}{(k-1)\mu})P_{k-2}] = \ldots = \frac{\rho^k}{k!}P_0$ (3)

$$Με \ αντικατάσταση \ της \ (3) \ στη \ (2): P_0 + \frac{\rho^k}{1!}P_0 + \cdots + \frac{\rho^c}{c!}P_0 = 1 \ ή$$

$$από \ γνωστή \ σειρά: P_0 = \frac{1}{\sum_{k=0}^{c}\frac{\rho^k}{k!}}$$

$$και \ συνεπώς \ η \ πιθανότητα \ να \ βρίσκεται \ ένα \ σύστημα \ στην \ κατάσταση \ k$$

$$ή \ η \ πιθανότητα \ απόρριψης \ πελάτη \ είναι:$$

$$P_{blocking} = B(\rho, c) = P_c = \frac{\rho^c \ / \ c!}{\sum_{k=0}^{c}\frac{\rho^k}{k!}} \ τύπος \ Erlang - B$$

Ο μέσος αριθμός απωλειών πελατών στην ουρά είναι ίσος με:

$$\lambda - \gamma = \lambda - \lambda\,(1 - P_{blocking}) = \lambda \cdot P_{blocking} = \lambda \cdot \frac{\rho^c \,/\, c!}{\sum_{k=0}^{c} \frac{\rho^k}{k!}}$$

Υλοποιούμε, όπως φαίνεται παρακάτω, τη ζητούμενη συνάρτηση erlangb_factorial, η οποία δέχεται ως ορίσματα την ένταση του φορτίου ρ και τον αριθμό των εξυπηρετητών c:

```
% Eirini Donti

% (i)
% Implementation of erlang_factorial(ρ,c) ρ = λ/μ and c = number of servers

 function fun = erlangb_factorial(r,c)
   sum=0;
   for k = 0:1:c
     sum = sum + (power(r,k)/factorial(k));
   endfor
    fun = (power(r,c)/factorial(c)/sum);
 endfunction
```

Επιβεβαιώνουμε, όπως φαίνεται παρακάτω, την ορθότητα της συνάρτησής μας με τη χρήση της συνάρτησης erlangb:

```
>> erlangb_factorial(8,8)
ans =   0.23557
>> erlangb(8,8)
ans =   0.23557
```

Παρατηρούμε ότι, η πιθανότητα απόρριψης πελάτη από το σύστημα B(ρ,c), είναι ίδια και στις δύο περιπτώσεις.

(2)

Υλοποιούμε, όπως φαίνεται παρακάτω, τη ζητούμενη συνάρτηση erlangb_iterative, η οποία δέχεται ως ορίσματα την ένταση του φορτίου ρ και τον αριθμό των εξυπηρετητών c:

```
% (ii)
% Implementation of erlang_iterative(ρ,c) ρ = λ/μ and c = number of servers

function fun = erlangb_iterative(r,c)
  fun = 1;
  for i = 0:1:c
    fun = ((r*fun)/((r*fun)+i));
  endfor
endfunction
```

Επιβεβαιώνουμε, όπως φαίνεται παρακάτω, την ορθότητα της συνάρτησής μας:

```
>> erlangb_iterative(8,8)
ans =  0.23557
```

Παρατηρούμε ότι το αποτέλεσμα που προκύπτει, είναι το ίδιο με τα αποτελέσματα των παραπάνω συναρτήσεων.

(3)

Καλούμε τις συναρτήσεις erlangb_factorial(1024,1024) και erlangb_iterative(1024,1024) και λαμβάνουμε την εξής έξοδο:

```
>> erlangb_factorial(1024,1024)
ans =  NaN
>> erlangb_iterative(1024,1024)
ans =  0.024524
```

Παρατηρούμε ότι, καλώντας τη συνάρτηση erlangb_factorial(1024,1024), δεν επιστρέφεται συγκεκριμένο αποτέλεσμα, ενώ η συνάρτηση erlangb_iterative(1024,1024) επιστρέφει την τιμή 0.024524. Αυτό συμβαίνει, διότι ο υπολογισμός της τιμής 1024! είναι, ουσιαστικά, πολύ μεγάλος αριθμός και συνεπώς δεν μπορεί να υπολογιστεί συγκεκριμένο αποτέλεσμα.

(4)

(α)

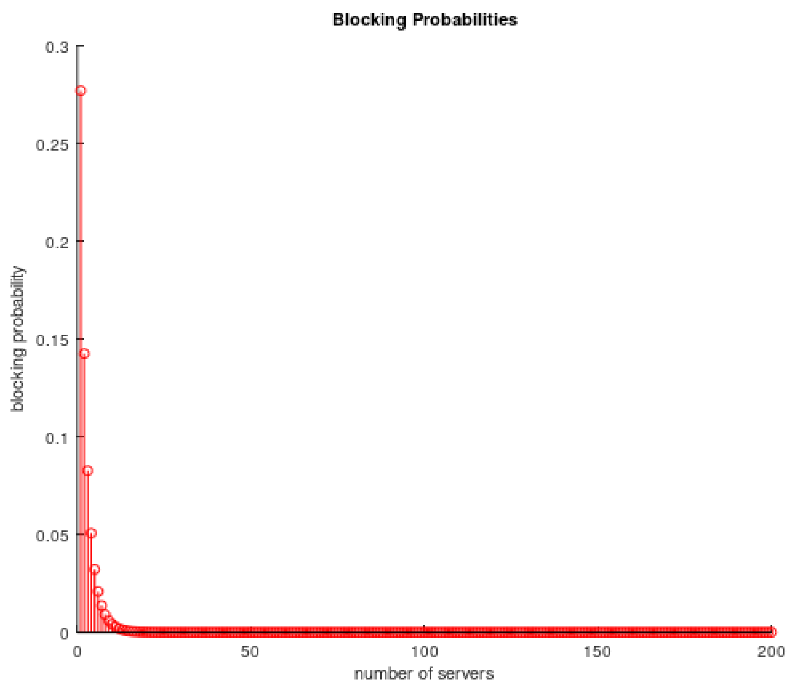
Σχεδιάζουμε από την αρχή το τηλεφωνικό δίκτυο μίας εταιρείας στην οποία απασχολούνται 200 εργαζόμενοι. Κάθε εργαζόμενος διαθέτει και μία εξωτερική γραμμή, δηλαδή η εταιρεία πληρώνει για 200 γραμμές. Για να σχεδιάσουμε ένα πιο οικονομικό δίκτυο, κάνουμε μετρήσεις στην ώρα αιχμής και βρίσκουμε ότι ο πιο απαιτητικός χρήστης χρησιμοποιεί συνολικά το τηλέφωνο του για εξωτερικές κλήσεις κατά μέσο όρο 23 λεπτά σε μία ώρα.

Η συνολική ένταση του φορτίου που καλείται να εξυπηρετήσει το τηλεφωνικό δίκτυο της εταιρείας, κάνοντας χρήση του πιο απαιτητικού προτύπου χρήστη είναι ίσο με:

$$\rho = 200\frac{23}{60} \cong 76{,}667\ Erlangs$$

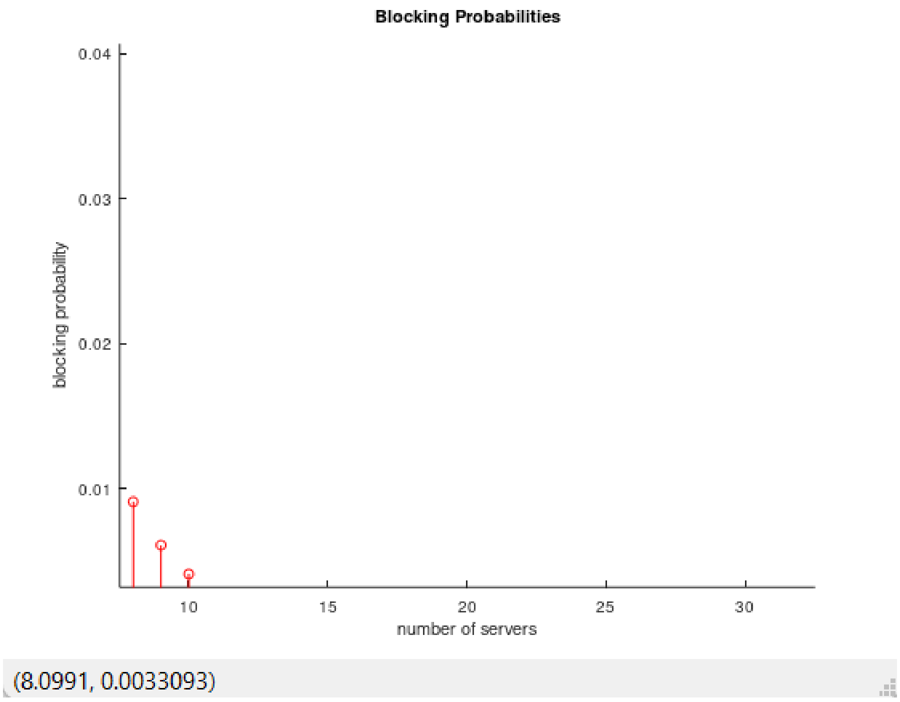
(β)

Εκτελούμε το διάγραμμα της πιθανότητας απόρριψης πελάτη από το σύστημα ως προς τον αριθμό των τηλεφωνικών γραμμών, επιλέγοντας από 1 ως 200 τηλεφωνικές γραμμές. Για τον υπολογισμό της πιθανότητας αυτής χρησιμοποιούμε τη συνάρτηση erlangb_iterative(). Οπότε, προκύπτει η παρακάτω γραφική παράσταση:



(γ)

Μεγεθύνουμε το παραπάνω διάγραμμα και συνεπώς προκύπτει το παρακάτω διάγραμμα:



Παρατηρούμε ότι ο κατάλληλος αριθμός τηλεφωνικών γραμμών που θα χρησιμοποιούσαμε ώστε η πιθανότητα απόρριψης της τηλεφωνικής κλήσης κάτω από 1%, είναι 8 τηλεφωνικές γραμμές.

Ο κώδικας που χρησιμοποιήθηκε για το παραπάνω υποερώτημα, είναι ο παρακάτω:

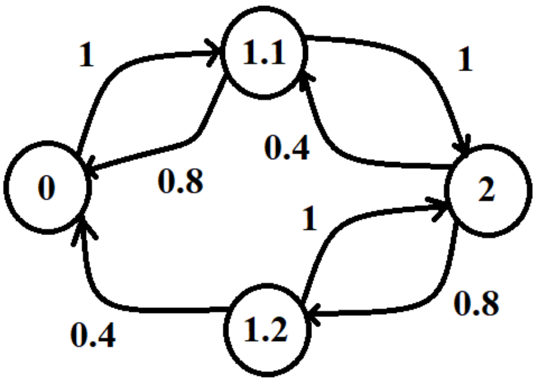
```
 % (b)

 Pr = zeros(0,200); % Run Selection
 for i = 1:1:200
   Pr(i) = erlangb_iterative(i*(23/60),i)
 endfor
figure;
stem(Pr,'r',"linewidth",0.5);
title("Blocking Probabilities");
xlabel("number of servers");
ylabel("blocking probability");
```

<u>Σύστημα εξυπηρέτησης με δύο ανόμοιους εξυπηρετητές</u>

Θεωρούμε σύστημα εξυπηρέτησης που αποτελείται από δύο εξυπηρετητές 1 και 2 χωρίς δυνατότητα αναμονής πελατών. Όταν και οι δύο εξυπηρετητές είναι διαθέσιμοι, ένας πελάτης δρομολογείται πάντα στον εξυπηρετητή 1. Σε περίπτωση που ο εξυπηρετητής 1 δεν είναι διαθέσιμος, ένας νέος πελάτης δρομολογείται στον εξυπηρετητή 2. Σε περίπτωση που και οι δύο εξυπηρετητές δεν είναι διαθέσιμοι, ένας νέος πελάτης απορρίπτεται χωρίς επανάληψη προσπάθειας εξυπηρέτησης. Οι αφίξεις πελατών στο σύστημα ακολουθούν την κατανομή Poisson με μέσο ρυθμό λ = 1 πελάτες/sec. Οι εξυπηρετήσεις είναι εκθετικά κατανεμημένες με μέσους χρόνους εξυπηρέτησης $1/\mu_1$ = 1.25sec , δηλαδή μ_1 = 0.8 πελάτες/sec και $1/\mu_2$ = 2.5 sec δηλαδή μ_2 = 0.4 πελάτες/sec, δηλαδή ο εξυπηρετητής 2 είναι χαμηλότερων δυνατοτήτων από τον εξυπηρετητή 1.

(1)

Σχεδιάζουμε το διάγραμμα ρυθμών μεταβάσεων του συστήματος στην κατάσταση ισορροπίας, όπως φαίνεται παρακάτω:



(α)

Οι εξισώσεις ισορροπίας του συστήματος είναι οι παρακάτω:

$$\lambda \cdot P_0 = \mu_1 \cdot P_{11} + \mu_2 \cdot P_{12}$$

$$(\lambda + \mu_1) \cdot P_{11} = p \cdot \lambda \cdot P_0 + \mu_2 \cdot P_2$$

$$(\lambda + \mu_2) \cdot P_{12} = (1 - p) \cdot \lambda \cdot P_0 + \mu_1 \cdot P_2$$

$$P_0 + P_{11} + P_{12} + P_2 = 1$$

Για μ_1= 0.8, $\mu_2 = 0.4, \lambda = 1 \; και \; p = 1$, οι παραπάνω εξισώσεις μετατρέπονται:

$$P_0 = 0.8 \cdot P_{11} + 0.4 \cdot P_{12}$$

$$1.8 \cdot P_{11} = P_0 + 0.4 \cdot P_2$$

$$1.4 \cdot P_{12} = 0.8 \cdot P_2$$

$$P_0 + P_{11} + P_{12} + P_2 = 1$$

Προκύπτουν οι παρακάτω εργοδικές πιθανότητες, λύνοντας το παραπάνω σύστημα:

$$P_0 = 0.24951$$

$$P_{11} = 0.21442$$

$$P_{12} = 0.19493$$

$$P_2 = 0.34113$$

(β)

Οπότε, η πιθανότητα απόρριψης πελατών από το σύστημα είναι:

$$P_{blocking} = P_2 = 0.34113$$

(γ)

Ο μέσος αριθμός πελατών στο σύστημα προκύπτει από τον παρακάτω τύπο:

$\sum_{k=0}^{2} k \cdot P(k) = 0 \cdot P_0 + 1 \cdot P_{11} + 1 \cdot P_{12} + 2 \cdot P_2 = 1.09161$

(2)

(α)

Συμπληρώνουμε τα κενά του δοσμένου προγράμματος. Το πρόγραμμα που δημιουργήσαμε, φαίνεται παρακάτω:

```
clc;
clear all;
close all;


lambda = 1;
m1 = 0.8;
m2 = 0.4;

threshold_1a = lambda/(lambda+m1);
threshold_1b = lambda/(lambda+m2);
threshold_2_first = lambda/(lambda+m2+m1);
threshold_2_second = (m1+lambda)/(lambda+m2+m1);

current_state = 0;
arrivals = zeros(1,4);
total_arrivals = 0;
maximum_state_capacity = 2;
previous_mean_clients = 0;
delay_counter = 0;
time = 0;

while 1 > 0
  time = time + 1;

  if mod(time,1000) == 0
    for i=1:1:4
      P(i) = arrivals(i)/total_arrivals;
    endfor

    delay_counter = delay_counter + 1;

    mean_clients = 0*P(1) + 1*P(2) + 1*P(3) + 2*P(4);
```

```
    delay_table(delay_counter) = mean_clients;

    if abs(mean_clients - previous_mean_clients) < 0.00001
       break;
    endif
    previous_mean_clients = mean_clients;
  endif

  random_number = rand(1);

  if current_state == 0
      current_state = 1;
      arrivals(1) = arrivals(1) + 1;
      total_arrivals = total_arrivals + 1;
  elseif current_state == 1
    if random_number < threshold_1a
      current_state = 3;
      arrivals(2) = arrivals(2) + 1;
      total_arrivals = total_arrivals + 1;
    else
      current_state = 0;
    endif
  elseif current_state == 2
    if random_number < threshold_1b
      current_state = 3;
      arrivals(3) = arrivals(3) + 1;
      total_arrivals = total_arrivals + 1;
    else
      current_state = 0;
    endif
  else
      if random_number < threshold_2_first

        arrivals(4) = arrivals(4) + 1;
        total_arrivals = total_arrivals + 1;
      elseif random_number < threshold_2_second
        current_state = 2;
      else
        current_state = 1;
      endif
   endif

endwhile

display(P(1));
display(P(2));
display(P(3));
display(P(4));
```

Τα κενά συμπληρώθηκαν με την παρακάτω λογική:

- threshold_1a: Από την κατάσταση 11 μπορούμε να έχουμε είτε αναχώρηση προς την κατάσταση 0 με ρυθμό μ_1 είτε άφιξη στην κατάσταση 2 με ρυθμό λ. Οπότε, ο λόγος $\frac{\text{άφιξη σε κατάσταση}}{\text{σύνολο καταστάσεων}} = \frac{\lambda}{\lambda + \mu_1}$.
- threshold_1b: Από την κατάσταση 12 μπορούμε να έχουμε είτε αναχώρηση προς την κατάσταση 0 με ρυθμό μ_2 είτε άφιξη στην κατάσταση 2 με ρυθμό λ. Οπότε, ο λόγος $\frac{\text{άφιξη σε κατάσταση}}{\text{σύνολο καταστάσεων}} = \frac{\lambda}{\lambda + \mu_2}$.
- threshold_2_first: Από την κατάσταση 2 μπορούμε να έχουμε άφιξη και απόρριψη πελάτη με ρυθμό λ και αναχώρηση στην κατάσταση 11 και 12 με ρυθμό μ_2 και μ_1αντίστοιχα. Οπότε, ο λόγος $\frac{\text{άφιξη σε κατάσταση}}{\text{σύνολο καταστάσεων}} = \frac{\lambda}{\lambda + \mu_1 + \mu_2}$.
- threshold_2_second: Από την κατάσταση 2 μπορούμε να έχουμε άφιξη και αναχώρηση στην κατάσταση 12 με ρυθμό μ_1 και αναχώρηση στην κατάσταση 11 και 12 με ρυθμό μ_2 και μ_1αντίστοιχα. Οπότε, ο λόγος $\frac{\text{άφιξη σε κατάσταση}}{\text{σύνολο καταστάσεων}} = \frac{\lambda + \mu_1}{\lambda + \mu_1 + \mu_2}$.

(β)

Τα κριτήρια σύγκλισης της προσομοίωσής στηρίζονται στο γεγονός ότι η διαφορά μεταξύ δύο διαδοχικών μέσων αριθμών πελατών πρέπει να είναι μικρότερη ή ίση από 0.001%.

```
abs(mean_clients - previous_mean_clients) < 0.00001
```

(γ)

Υπολογίζουμε τις εργοδικές πιθανότητες του συστήματος από την παραπάνω προσομοίωση:

```
>> display(P(1));

 0.24590
>> display(P(2));

 0.21585
>> display(P(3));

 0.19621
>> display(P(4));

 0.34204
```

Παρατηρούμε ότι τα αποτελέσματα που προκύπτουν από την παραπάνω προσομοίωση, για τις εργοδικές πιθανότητες, είναι περίπου ίσες με εκείνες που υπολογίσαμε σε θεωρητικά πλαισία.