Συστήματα Αναμονής Άσκηση 5^{η}

Αυγερινός Πέτρος 03115074

Contents

1 Δίκτυο με εναλλακτική δρομολόγηση

1.1 Ερώτημα 1

Για να μπορεί το σύστημα να μοντελοποιηθεί ως ουρά M/M/1 θα πρέπει τα ρ_i να είναι μικρότερα της μονάδας.

Για την γραμμή 1 έχουμε ότι:

$$\lambda_1 = a \cdot 10^4 pps \text{ και } \mu_1 = \frac{c_1}{packet} = \frac{15 \cdot 10^6}{64 \cdot 8} = 29.296 \cdot 10^3 pps \tag{1}$$

Για την γραμμή 2 έχουμε ότι:

$$\lambda_2 = (1-a) \cdot 10^4 pps \text{ και } \mu_2 = \frac{c_2}{packet} = \frac{12 \cdot 10^6}{64 \cdot 8} = 23.437 \cdot 10^3 pps \tag{2}$$

Άρα θα πρέπει να ισχύει:

$$\rho_1 = \frac{\lambda_1}{\mu_1} < 1 \text{ και } \rho_2 = \frac{\lambda_2}{\mu_2} < 1 \tag{3}$$

Για το ρ_1 έχουμε:

$$\rho_1 = \frac{a \cdot 10^4}{29.296 \cdot 10^3} < 1 \Rightarrow a < 2.9296 \tag{4}$$

Για το ρ_2 έχουμε:

$$\rho_2 = \frac{(1-a) \cdot 10^4}{23.437 \cdot 10^3} < 1 \Rightarrow 1 - a < 2.3437 \Rightarrow a > -1.3437 \tag{5}$$

Όμως οι (4) και (5) ισχύουν καθώς το α είναι ποσοστό, και παίρνει τιμές μεταξύ 0 και 1. Επομένως το σύστημα είναι εργοδικό και το σύστημα μπορεί να μοντελοποιηθεί ως ουρά M/M/1.

1.2 Ερώτημα 2

Ο μέσος αριθμός πελατών στο σύστημα είναι:

$$E[N] = E[N_1] + E[N_2] \tag{6}$$

Και γνωρίζουμε από τον τύπο του Little ότι ο μέσος χρόνος καθυστέρησης είναι:

$$E(T) = \frac{E[n]}{\lambda} \tag{7}$$

Οι τιμές για την ελάχιστη καθυστέρηση για συγκεκριμένο α είναι:

$$Delay = 4.60382e - 05 \text{ για } a = 0.674$$

Παρακάτω φαίνεται η γραφική παράσταση της μέσης καθυστέρησης για τιμές του α από 0 έως 1.

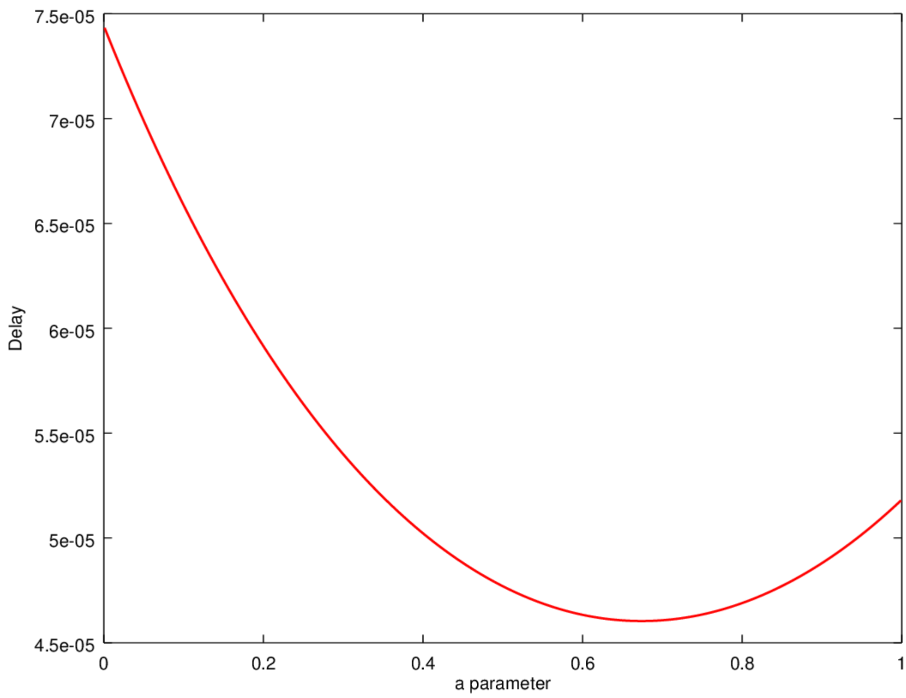

Figure 1: Μέση καθυστέρηση για τιμές του α

Ο κώδικας που χρησιμοποιηθήκε για τα παραπάνω είναι ο εξής:

```
pkg load queueing;

a = 0.001:0.001:0.999;
lambda = 10000;
lambda_1 = a*lambda;
lambda_2 = (1-a)*lambda;
mu_1 = 29296;
mu_2 = 23437;

[_,_,Q1,_,_] = qsmm1(lambda_1,mu_1);
[_,_,Q2,_,_] = qsmm1(lambda_2,mu_2);

clients = Q1 + Q2;
time = clients/lambda;

figure(1);
plot(a,time,"r","linewidth",1.2);
xlabel("a parameter");
ylabel("Delay");

filename = "../images/delay_to_a.png";
print("-dpng", filename);

[minimum, minimum_a] = min(min(time,[],1));
fd = fopen("ask1.txt", "w");
min_a = 0.001*minimum_a
text = sprintf("Minimum Delay=%d for a=%.3f", minimum, min_a);
fprintf(fd,text);
fclose(fd);
```

2 Ανοιχτό δίκτυο ουρών αναμονής

2.1 Ερώτημα 1

Οι παραδοχές για να μπορεί να μελετηθεί το συγκεκριμένο σύστημα ως ανοιχτό δίκτυο με το θεώρημα Jackson είναι:

- Τα Q_i συνιστούν δικτυακούς κόμβους εξυπηρέτησης κορμού με εκθετικούς ρυθμούς μ_i
- Οι αφίξεις των πελατών προέρχονται από εξωτερικές πηγές που είναι άμεσα συνδεδεμένες στους δικτυακούς κόμβους κορμού Q_1 και Q_2 και καταλήγουν στον κόμβο Q_5 με απώλειες κατά τη διαδρομή. Οι ροές μεταξύ των δικτυακών κόμβων είναι ανεξάρτητες ροές Poisson με μέσο ρυθμό $\gamma_{i,j}$ και η συνολική εξωγενής ροή Poisson είναι ίση με $\gamma_i = \sum_{j=1}^{5} \gamma_{i,j}$ με $i \neq j$
- Η εσωτερική δρομολόγηση μεταξύ των κόμβων(ουρών) γίνεται με τυχαίο τρόπο από και πιθανότητα $r_{i,j}$
- Οι ροές που διαπερνούν το κόμβο Q_i έχουν συνολικό μέσο ρυθμό $\lambda_j = \gamma_j + \sum_{i=1}^{5} r_{i,j}\lambda_i$
- Οι χρόνοι εξυπηρέτησεις όπως διαπερνούν το δίκτυο δεν διατηρούν την τιμή τους, είναι δηλαδή χωρίς μνήμη. Η τιμή τους εξαρτάται από την κατανομή του κάθε εξυπηρετητή.

Τα $r_{i,j}$ είναι τα εξής:

- $r_{1,2} = \dfrac{1}{2}$
- $r_{1,3} = \dfrac{1}{4}$
- $r_{2,3} = \dfrac{1}{3}$
- $r_{2,4} = \dfrac{2}{3}$
- $r_{4,5} = \dfrac{3}{5}$

Τα ρ_i είναι τα εξής:

- $\rho_1 = \dfrac{\lambda_1}{\mu_1}$
- $\rho_2 = \dfrac{\lambda_2 + r_{1,2} \cdot \lambda_1}{\mu_2}$
- $\rho_3 = \dfrac{r_{2,3} \cdot r_{1,2} \cdot \lambda_1 + r_{1,3} \cdot \lambda_1 + r_{2,3} \cdot \lambda_2}{\mu_3}$
- $\rho_4 = \dfrac{\lambda_2 \cdot r_{2,4}}{\mu_4}$
- $\rho_5 = \dfrac{\lambda_1 \cdot r_{1,3} \cdot r_{3,5} + \lambda_2 \cdot r_{2,4} \cdot r_{4,5}}{\mu_5}$

2.2 Ερώτημα 2

Ο κώδικας της συνάρτησης intesities φαίνεται παρακάτω:

```
function [r_1, r_2, r_3, r_4, r_5, e] = intensities(lambda_1, lambda_2, mu_1, mu_2, mu_3, mu_4, mu_5)
  r_1 = (lambda_1/mu_1);
  r_2 = ((lambda_2+(1/2)*lambda_1)/mu_2);
  r_3 = ((2/3)*(1/2)*lambda_1 + (1/4)*lambda_1 + (1/3)*lambda_2)/mu_3;
  r_4 = ((2/3)*lambda_2/mu_4);
  r_5 = ((1/4)*lambda_1 + (2/3)*(3/5)*lambda_2)/mu_5;
  if((r_1 < 1) && (r_2 < 1) && (r_3 < 1) && (r_4 < 1) && (r_5 < 1))
    e = 1;
  else
    e = 0;
  endif
endfunction
```

2.3 Ερώτημα 3

Ο κώδικας της συνάρτησης mean_clients φαίνεται παρακάτω:

```
% addpath('./intensities.m');
function [Q1, Q2, Q3, Q4, Q5] = mean_clients(lambda_1, lambda_2, mu_1, mu_2, mu_3, mu_4, mu_5)
    [r_1, r_2, r_3, r_4, r_5, e] = intensities(lambda_1, lambda_2, mu_1, mu_2, mu_3, mu_4, mu_5);
    Q1 = r_1/(1-r_1);
    Q2 = r_2/(1-r_2);
    Q3 = r_3/(1-r_3);
    Q4 = r_4/(1-r_4);
    Q5 = r_5/(1-r_5);
endfunction
```

2.4 Ερώτημα 4

Οι ροές και η μέση καθυστέρηση για το συγκεκριμένο σύστημα είναι:

```
==================================
 Intensities
----------------------------------
 r_1 = 0.666667
 r_2 = 0.550000
 r_3 = 0.783333
 r_4 = 0.285714
 r_5 = 0.408333
==================================
Mean Clients
----------------------------------
Q1 = 2.000000
Q2 = 1.222222
Q3 = 3.615385
Q4 = 0.400000
Q5 = 0.690141
==================================
Mean Delay
----------------------------------
D = 0.495484
==================================
```

Ο κώδικας για τα παραπάνω αποτελέσματα δίνεται παρακάτω:

```
lambda_1 = 10;
lambda_2 = 6;
mu_1 = 15;
mu_2 = 20;
mu_3 = 10;
mu_4 = 14;
mu_5 = 12;

[r_1, r_2, r_3, r_4, r_5, e] = intensities(lambda_1, lambda_2, mu_1, mu_2, mu_3, mu_4, mu_5);
if (e > 1)
    disp("Unstable system");
    return;
endif

[Q1, Q2, Q3, Q4, Q5] = mean_clients(lambda_1, lambda_2, mu_1, mu_2, mu_3, mu_4, mu_5);

E = Q1 + Q2 + Q3 + Q4 + Q5;
gamma = lambda_1 + lambda_2;
D = E/gamma;
text = sprintf("D = %f\n", D);

fd = fopen('output.txt', 'w');
fprintf(fd, "======================\n");
fprintf(fd, "Intensities\n");
fprintf(fd, "----------------------\n");
fprintf(fd, "r_1 = %f\n", r_1);
fprintf(fd, "r_2 = %f\n", r_2);
fprintf(fd, "r_3 = %f\n", r_3);
fprintf(fd, "r_4 = %f\n", r_4);
fprintf(fd, "r_5 = %f\n", r_5);
fprintf(fd, "======================\n");
fprintf(fd, "Mean Clients\n");
fprintf(fd, "----------------------\n");
fprintf(fd, "Q1 = %f\n", Q1);
fprintf(fd, "Q2 = %f\n", Q2);
fprintf(fd, "Q3 = %f\n", Q3);
fprintf(fd, "Q4 = %f\n", Q4);
fprintf(fd, "Q5 = %f\n", Q5);
fprintf(fd, "======================\n");
fprintf(fd, "Mean Delay\n");
fprintf(fd, "----------------------\n");
fprintf(fd, "%s", text);
fprintf(fd, "======================\n");

fclose(fd);
```

2.5 Ερώτημα 5

Ο στενωπός είναι η ουρά 1 καθώς έχει τη μεγαλύτερη ροή σύμφωνα με τα αποτελέσματα του παραπάνω ερωτήματος.

Για να μεγιστοποιήσουμε το λ_1 ενώ παράλληλα διατηρούμε τα ρ_i μικρότερα της μονάδας θα πρέπει να λύσουμε το σύστημα των ανισώσεων που προκύπτουν από τις παραπάνω σχέσεις. Η λύση του συστήματος είναι προφανής και είναι ίση με το $\lambda_1 = 15$ η οποία προκύπτει απο τη ρ_1 καθώς όλες οι υπόλοιπες ροές προκαλούν μεγαλύτερα αποτελέσματα στο άνω άκρο των ανισώσεων.

2.6 Ερώτημα 6

Το διάγραμμα που απεικονίζει τη μέση καθυστέρηση για λ_1 από 1,5 μέχρι 14,85 με βήμα 0,15 φαίνεται παρακάτω:

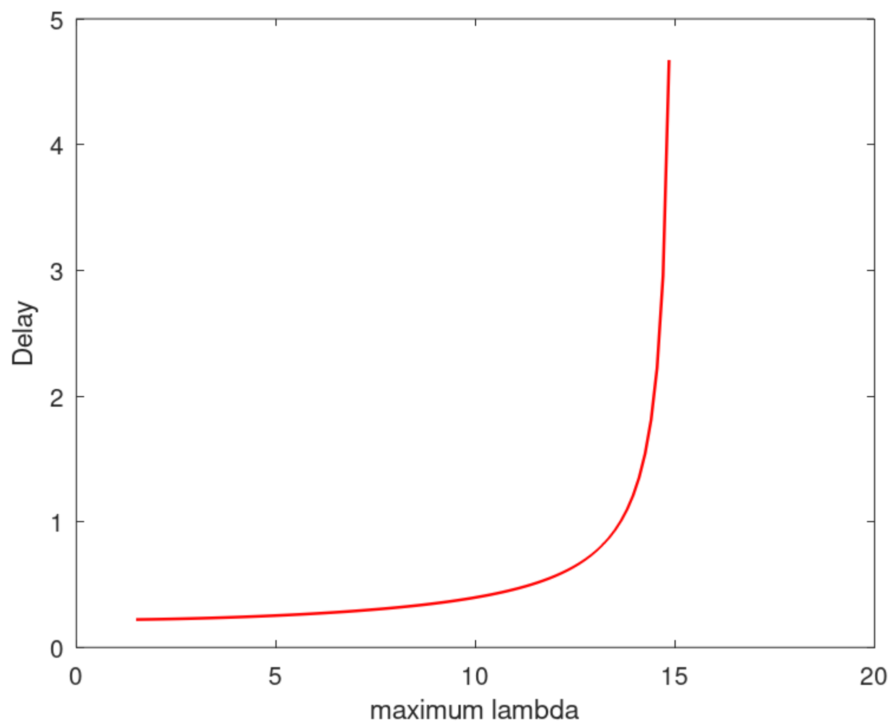

Figure 2: Μέση καθυστέρηση για τιμές του λ_1

Ενώ ο κώδικας για την πραγματοποίηση του παραπάνω διαγράμματος είναι ο εξής:

```
lambda_1 = 10;
lambda_2 = 6;
mu_1 = 15;
mu_2 = 20;
mu_3 = 10;
mu_4 = 14;
mu_5 = 12;

max_lambda = 15;
lambda = (0.1*max_lambda):(0.01*max_lambda):(0.99*max_lambda);

for i = 1:1:90
    percentage = i/100;
    lambda_1 = percentage*max_lambda;
    [Q1, Q2, Q3, Q4, Q5] = mean_clients(lambda_1, lambda_2, mu_1, mu_2, mu_3, mu_4, mu_5);
   E(i)= (Q1+Q2+Q3+Q4+Q5)/(lambda_1+lambda_2);
endfor

figure(1);
plot(lambda, E, "r", "linewidth", 1.2);
xlabel("maximum lambda");
ylabel("Delay");
filename = "../images/delay_to_lambda.png";
print("-dpng", filename);
```