

Network Engineering - Mid-Term exam - 2019/2020

Problem 1 - 5 points

Calculate the expected end to end delay for the packets generated by A1. These packets, once they are transmitted from the server, traverse router 1 (R1), router 2 (R2), and an access point (AP) until they reach the destination. Note that background traffic (BG) appears in the network interface of R1. Justify all steps and results.

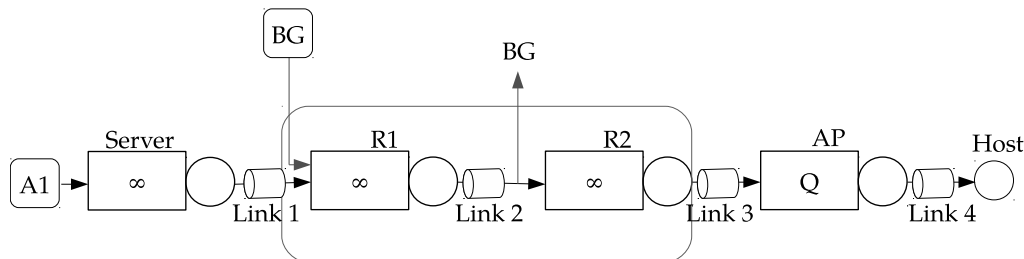


Figure 1: End to end path

Consider the following data:

- A1: Generates packets following a Poisson process with rate $\lambda = \frac{B_{A1}}{E[L]}$ packets/second. The size of the generated packets is random and follows an exponential distribution with average value $E[L]$. The traffic flow generated by A1 is equal to $B_{A1} = 14$ Mbps.
- BG: Generates packets following a Poisson process with rate $\lambda_{BG} = 6500$ packets/second. The size of the generated packets is random and follows an exponential distribution with average value $E[L]$ (the same than for the packets generated by A1).
- Link 1 has a capacity of $R_1 = 18$ Mbps. Link 2 has a capacity of R_2 Mbps. Link 3 has a capacity of $R_3 = 50$ Mbps, and link 4 has a capacity of $R_4 = 12$ Mbps.
- The propagation delay of links 1, 2 and 3 is 10 ms. The propagation delay of link 4 is negligible.
- The buffer size of the AP is $Q = 6$ packets.

Important: Choose the missing values as you consider most appropriate. The only requirement is that all network interfaces must be stable. In all links, consider that the expected transmission time is $E[D_s] = E[L]/R$.

Problem 2 - 5 points

Consider that a task server in the cloud can be modelled by a M/M/3/5 system. Tasks arrive to this system with rate λ . The expected service time of a task is $E[D_s]$. Choose the missing values as you consider most appropriate. Justify all results obtained.

1. (1 point) Given that $X(t)$ is the stochastic process representing the number of tasks in the server, identify the state space of $X(t)$.
2. (1 point) Draw the Markov chain representing the system. Indicate the arrival and departure rates for each state, and calculate the value of the normalized traffic load.
3. (1 point) Calculate the task blocking probability, the probability the server is empty, and the probability that at least two servers are busy.
4. (1 point) Calculate the expected number of tasks in service.
5. (1 point) Calculate the expected delay that the tasks spend in service using previous result.

Problema 1 - 5 points

Calculeu el retard mig extrem a extrem per als paquets generats per A1. Aquests paquets, un cop transmesos des del servidor, travessen l'encaminador 1 (R1), l'encaminador 2 (R2) i un punt d'accés (AP) fins arribar a la destinació. Tingueu en compte que el trànsit de fons (BG) apareix a la interfície de xarxa de R1. Justifiquen tots els passos i resultats.

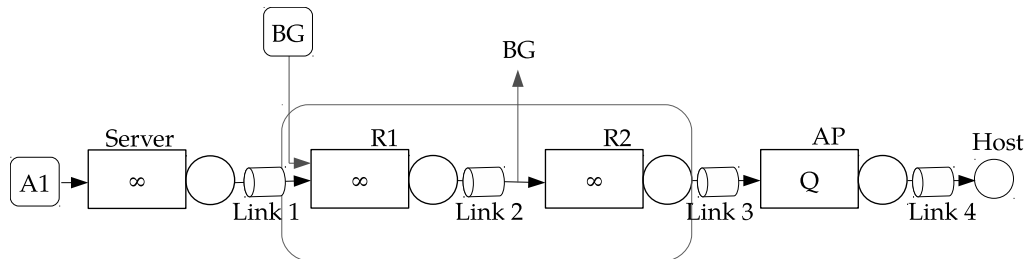


Figure 2: End to end path

Consider the following data:

- A1: Genera paquets després d'un procés de Poisson amb la taxa $\lambda = \frac{B_{A1}}{E[L]}$ paquets / segon. La mida dels paquets generats és aleatòria i segueix una distribució exponencial amb valor mig $E[L]$. El flux de trànsit generat per A1 és igual a $B_{A1} = 14$ Mbps.
- BG: genera paquets seguint un procés de Poisson amb taxa $\lambda_{BG} = 6500$ paquets / segon. La mida dels paquets generats és aleatòria i segueix una distribució exponencial amb un valor mig $E[L]$ (el mateix que per als paquets generats per A1).
- Enllaç 1 té una capacitat de $R_1 = 18$ Mbps. L'enllaç 2 té una capacitat de R_2 Mbps. L'enllaç 3 té una capacitat de $R_3 = 50$ Mbps, i l'enllaç 4 té una capacitat de $R_4 = 12$ Mbps.
- El retard de propagació dels enllaços 1, 2 i 3 és de 10 ms. El retard de propagació de l'enllaç 4 és insignificant.
- La mida del buffer de l'AP és de $Q = 6$ paquets.

Important: Trieu els valors que falten que considereu més adequats. L'únic requisit és que totes les interfícies de xarxa han de ser estables. En tots els enllaços, considereu que el temps previst de transmissió és $E[D_s] = E[L]/R$.

Problema 2 - 5 points

Considereu un servidor que executa tasques al núvol pot ser modelat per un sistema M/M/3/5. Les tasques arriben a aquest sistema amb taxa λ . El temps previst del servei d'una tasca és $E[D_s]$. Trieu els valors que falten que considereu més adequats. Justifiquen tots els resultats obtinguts.

1. (1 punt) Tenint en compte que $X(t)$ és el procés estocàstic que representa el nombre de tasques en el servidor, identifiqueu l'espai d'estat de $X(t)$.
2. (1 punt) Dibuixeu la cadena de Markov que representa el sistema. Indiqueu les taxes d'arribada i sortida per a cada estat i calculeu el valor de la càrrega de trànsit normalitzada.
3. (1 punt) Calculeu la probabilitat de bloqueig de tasques, la probabilitat que el servidor estigui buit i la probabilitat que almenys dos servidors estiguin ocupats.
4. (1 punt) Calculeu el nombre mig de tasques en servei.
5. (1 punt) Calculeu el retard mig que les tasques passen en servei a partir del resultat anterior.

Solution 1

```
function Exercise1()

% Parameters
BA1=14E6;
lambdaBG=6500;
R1=18E6;
R2=100E6;
R3=50E6;
R4=12E6;

Dprop=10E-3;
Q=6;
EL=10000;
lambda=BA1/EL;

% Norm. traffic loads
a1=lambda*EL/R1;
a2=(lambda+lambdaBG)*EL/R2;
a3=(lambda)*EL/R3;
a4=lambda*EL/R4; % Normalized traffic load at the AP

disp('Norm traffic loads');
disp([a1 a2 a3 a4]);

% Since link4 is a MM1K is always stable

% Links
ED1 = 1/(R1/EL-lambda) + Dprop;
ED2 = 1/(R2/EL-lambda-lambdaBG) + Dprop;
ED3 = 1/(R3/EL-lambda)+Dprop;

EN4=0;
K=Q+1;
for k=0:K
    EN4=EN4+k*(1-a4)*a4^k/(1-a4^(K+1));
end

Pb4 = (1-a4)*a4^K/(1-a4^(K+1));
ED4 = EN4/(lambda*(1-Pb4));

EDendtoend = ED1+ED2+ED3+ED4;

disp('Delays | End to end');
disp([ED1 ED2 ED3 ED4 EDendtoend]);

end
```

Solution 2

```
function Exercise2()

% State Space: 0-1-2-3-4-5

% Considering the following values
lambda = 3;
EDs = 0.6;
mu=1/EDs;
a=lambda/mu;

% pi0 * lambda = pi1 * mu
% pi1 * lambda = pi2 * 2*mu
% pi2 * lambda = pi3 * 3*mu
% pi3 * lambda = pi4 * 3*mu
% pi4 * lambda = pi5 * 3*mu

% pi1 = a * pi0
% pi2 = a/2 * pi1 = a^2/2*pi0
% pi3 = a/3 * pi2 = a^3/6*pi0
% pi4 = a/3 * pi3 = a^4/18*pi0
% pi5 = a/3 * pi4 = a^5/54*pi0

pi0=1/(1+a+a^2/2+a^3/6+a^4/18+a^5/54);
pi1 = a*pi0;
pi2 = a^2/2*pi0;
pi3 = a^3/6*pi0;
pi4 = a^4/18*pi0;
pi5 = a^5/54*pi0;

disp('Stationary Distribution');
disp([pi0 pi1 pi2 pi3 pi4 pi5]);

Pb = pi5;
Pe = pi0;
P2seversatleast = 1-pi0-pi1;

disp('Blocking | Empty | At least 2 servers busy');
disp([Pb Pe P2seversatleast]);

ENs = 1*pi1+2*pi2+3*pi3+3*pi4+3*pi5;

EDs = ENs/(lambda*(1-Pb));

disp('E[Ns] | E[Ds]');
disp([ENs EDs 1/mu]);

end
```