Network Engineering - Mid-Term exam - Group 2 - 2020/2021

Next-generation WiFi will implement Multi-link Operation (MLO). This will allow multi-band Access Points to transmit data to a single station at the same time through multiple network interfaces, each one operating at a different frequency band. Figure 1 depicts the considered scenario. It can be seen that the arriving packets to the AP are distributed over the different interfaces following the weights α_1, α_2, and α_3, where $\alpha_1 + \alpha_2 + \alpha_3 = 1$. Also, since different propagation conditions can be found at the different bands, each interface can transmit data at a different rate.

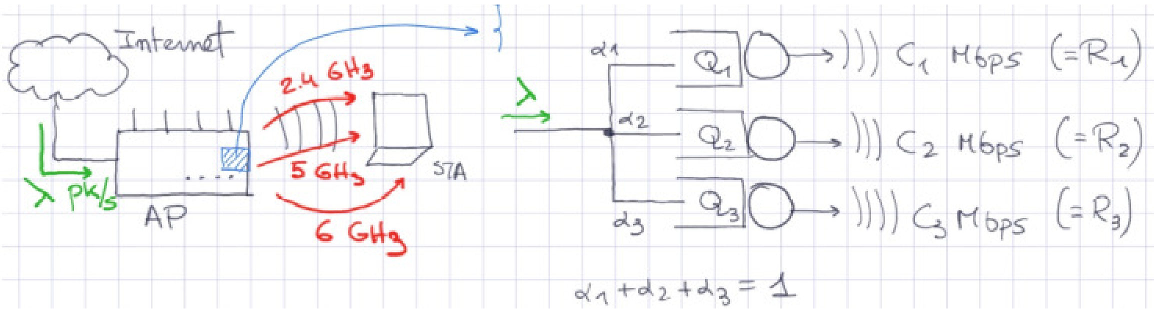

Figure 1: Multi-Link Operation

Problem 1 (30 mins) - 5 points

Following the previous description, let us consider that each network interface can be modelled as an M/M/1 system (i.e., the buffer (Q_i) is large enough to be considered as infinite). The transmission rates (link capacities) of each interface are given by $R_1 = 2 \cdot x_1$ Mbps, $R_2 = 1.5 \cdot x_1$ Mbps, and $R_3 = 1 \cdot x_1$ Mbps, respectively, with x_1 the 4th number of your NIA. Packets arrive following a Poisson process of rate λ and their size is exponentially distributed with average $\mathbb{E}[L] = 1000 \cdot x_2$ bits, with x_2 the second number of your NIA. In case x_1 or x_2 are zero, use 1.

1. (0.5 points) Write your NIA, and the values of R_1, R_2, R_3, and $\mathbb{E}[L]$.
2. (2.5 points) Calculate the average packet delay in each network interface if the arriving packets are uniformly distributed over the 3 network interfaces, i.e., $\alpha_1 = \alpha_2 = \alpha_3 = 1/3$. To do so, consider the traffic load arriving at the AP is $2.75 \cdot x_1$ Mbps.
3. (2 points) Is the previous traffic balancing policy (i.e., the α values) optimal? Justify the answer, and if not, propose a policy that offers a better performance. Quantify numerically the gains, and explain the results.

Problem 2 (45 mins) - 5 points

Alternatively to the case considered in previous problem, in MLO, the three interfaces can also share the same buffer, as illustrated in Figure 2. In this case, let us consider that the system can be modelled by a M/M/3/K system, with $Q = \min(4, 2 \cdot x_3)$, and x_3 the 5th value of your NIA (use 1 if it is 0). Also, consider that the three interfaces now use the same transmission rate, $R = R_2$.

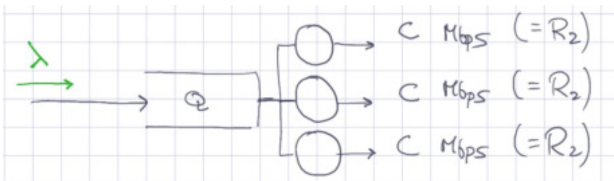

Figure 2: Multi-Link Operation - Shared buffer

1. (1.25 points) Draw the Markov chain that represents the M/M/3/K system, write its local balance equations, and find its equilibrium distribution.
2. (1.25 points) Obtain the blocking probability, and the probability the three interfaces are transmitting packets.
3. (1.25 points) Calculate the average packet delay, the average waiting delay and system throughput.
4. (1.25 points) Compare the results obtained in Problem 2 with the ones from Problem 1. Note that the set-up is not exactly the same, and so the numerical values are not directly comparable. Therefore, keep the discussion at the qualitative level.

```
% MidTerm - Grup 2 - 2020/2021

function Midterm_Group2_20202021()

Problem1();
Problem2();

end


% -------------------------- Problem 1

function Problem1()

% a)

x1=1;

R1 = 2E6*x1;
R2 = 1.5E6*x1;
R3 = 1E6*x1;

x2=4;

EL = 1000*x2;

% b)

alfa = 1/3;

lambda = 2.75E6 * x1 / EL;

lambda1 = alfa * lambda;
lambda2 = alfa * lambda;
lambda3 = alfa * lambda;

disp('Packet Arrival Rate: Aggregate | 1 | 2 | 3');
disp([lambda lambda1 lambda2 lambda3]);

mu1=R1/EL;
mu2=R2/EL;
mu3=R3/EL;

a1 = lambda1/mu1;
a2 = lambda2/mu2;
a3 = lambda3/mu3;

disp('Traffic load (a) per interface');
disp([a1 a2 a3]);

ED1 = 1 /(mu1-lambda1);
ED2 = 1 /(mu2-lambda2);
ED3 = 1 /(mu3-lambda3);

EED = (lambda1/lambda)*ED1 + (lambda2/lambda)*ED2 + (lambda3/lambda)*ED3;

disp('Delays in each interface | Average across interfaces (seconds)');
disp([ED1 ED2 ED3 EED]);

% c)

% It is not optimal. The optimal one requires a proportional share taking
```

```
% into account the link capacities

alfa1 = R1/(R1+R2+R3);
alfa2 = R2/(R1+R2+R3);
alfa3 = R3/(R1+R2+R3);

disp('Alfa values');
disp([alfa1 alfa2 alfa3]);

lambda1 = alfa1 * lambda;
lambda2 = alfa2 * lambda;
lambda3 = alfa3 * lambda;


disp('[Optimal] Packet Arrival Rate: Aggregate | 1 | 2 | 3');
disp([lambda lambda1 lambda2 lambda3]);

mu1=R1/EL;
mu2=R2/EL;
mu3=R3/EL;

a1 = lambda1/mu1;
a2 = lambda2/mu2;
a3 = lambda3/mu3;

disp('[Optimal] Traffic load (a) per interface');
disp([a1 a2 a3]);

ED1 = 1 /(mu1-lambda1);
ED2 = 1 /(mu2-lambda2);
ED3 = 1 /(mu3-lambda3);

EED = (lambda1/lambda)*ED1+(lambda2/lambda)*ED2+(lambda3/lambda)*ED3;

disp('[Optimal] Delays in each interface | Average (seconds)');
disp([ED1 ED2 ED3 EED]);


end

% --------------------------- Problem 2

function Problem2()

x3=5;

Q = min(4,2*x3);

R = 1.5E6;

% 1) M/M/3/7 system

lambda = 687.5;
EL = 4000;
mu=R/EL;
a=lambda/mu;

disp('Normalized Traffic Load');
disp(a);

pi0=(1/(1 + a + a^2/2 + a^3/6 + a^4/18 + a^5/54 + a^6/162 + a^7/486));
pi1=a*pi0;
pi2=(a^2/2)*pi0;
```

```
pi3=(a^3/6)*pi0;
pi4=(a^4/18)*pi0;
pi5=(a^5/54)*pi0;
pi6=(a^6/162)*pi0;
pi7=(a^7/486)*pi0;

disp('Eq. distribution | sum (for validation)');
disp([pi0 pi1 pi2 pi3 pi4 pi5 pi6 pi7 pi0+pi1+pi2+pi3+pi4+pi5+pi6+pi7]);

% 2)

Pb = pi7;
P3tx = pi3+pi4+pi5+pi6+pi7;

disp('Prob Blocking | Prob the 3 interfaces are tx');
disp([Pb P3tx]);

% 3)

EN = pi1 + 2* pi2 + 3* pi3 + 4* pi4 +5* pi5 +6* pi6 + 7* pi7;
ED = EN/(lambda*(1-Pb));

disp('Av. delay in the AP: E[D]');
disp(ED);

EDs = EL/R;
EDq = ED - EL/R;

disp('Av. waiting delay in the AP: E[Dq]');
disp(EDq);

% Alternative
%ENq=1* pi4 +2* pi5 +3* pi6 + 4* pi7;
%EDq = ENq / (lambda*(1-Pb))

Th = lambda*(1-Pb);

disp('Throughput');
disp(Th);

% 4) sharing the same buffer is a much better solution since we guarantee
% that all 3 interfaces are transmitting packets when there are packets
% waiting. Otherwise, we can have interfaces that are not used, while
% others have many packets waiting.


end
```