UCSD ECE 158A
Data Networks I
Massimo Franceschetti

Extra Problems Review 1.

Problems

Problem 1 Assume that a host A in Berkeley sends a stream of packets to a host B in Boston. Assume also that all links operate at 100Mbps and that it takes 130ms for the first acknowledgment to come back after A sends the first packet. Say that A sends one packet of 1KByte and then waits for an acknowledgment before sending the next packet, and so on. What is the long-term average bit rate of the connection? Assume now that A sends N packets before it waits for the first acknowledgment, and that A sends the next packet every time an acknowledgement is received. Express the long-term average bit rate of the connection as a function of N . [Note: 1Mbps = 10^6 bits per second; 1ms = 1 millisecond = 10^{-3} second.]

SOLUTION:

In the first case the long term average bit-rate is $B_1 = 1kByte/0.13s = 61,538bps$.

For the second case, we proceed as follows. Let $T = 130ms$. It can be written as $T = 2T_0 + T_1$, where T_0 is the propagation delay and T_1 is the time to send a $1kByte$ packet (at time T_0 the first bit of the packet reaches B, at time $T_0 + T_1$ the last bit of the packet reaches B, and the ACK is received by A at $T_0 + T_1 + T_0$). We have that $T_1 = 1kByte/1000Mbps = 0.08ms$ and $T_0 = (T - T_1)/2 = 64.96ms$.

When A sends N packets before it waits for the first ACK, we have that N packets are sent to B by time $2T_0 + T_1$. The resulting bandwidth would be

$$B(N) = N \cdot B_1 = N \cdot 61,538bps. \tag{1}$$

Equation 1 suggests that we can reach an arbitrarily large bit-rate by increasing N, which is not consistent with the link capacity of $100Mbps$. However, we cannot increase the bit-rate if A completes the transmission of the N-th packet after the is received 1-st ACK. In such a case A has to complete the transmission of the N-th packet before starting the transmission of the transmission of the $(N + 1)$-th packet. A completes the transmission of the N-th packet at time $N \cdot T_1$, while A receives the first ACK at time $2T_0 + T_1$. Therefore, Equation 1 is valid for values of N such that

$$N \cdot T_1 \leq 2T_0 + T_1,$$

that is $N \leq N^* = 2T_0/T_1 + 1 = 1,625$. The resulting bit rate is

$$B(N) = \begin{cases} N \cdot 61,538bps, & N \leq N^* \\ N^* \cdot 61,538bps = 100Mbps, & N > N^*. \end{cases}$$

Problem 2. We want to multicast N packets $p_1, \ldots, p_N$ across a link, but unfortunately packets can get lost. We decide to implement a FEC strategy, which consists in generating and transmitting $M > N$ packets $c_1, \ldots, c_M$. Each packet c_i is obtained as a combination (modulo-2 sum) of a subset of $p_1, \ldots, p_N$. For example, fixed $a_{i,1}, a_{i,2}, \ldots, a_{i,N} \in \{0, 1\}$

$$c_i = \begin{bmatrix} a_{i,1} & a_{i,2} & \ldots & a_{i,N} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \ldots \\ p_N \end{bmatrix}$$

means that c_i is obtained as the (bit-by-bit) modulo-2 sum of the packets p_j corresponding to all coefficients $a_{i,j}$ that are equal to 1.

Out of all M transmitted packets, $M' \leq M$ are received. Let these packets be $c_{i_1}, c_{i_2}, \ldots, c_{i_{M'}}$, where $i_1, \ldots, i_{M'}$ are distinct indices in $\{1, \ldots, M\}$. The received packets $c_{i_1}, \ldots, c_{i_{M'}}$ and the original packets $p_1, \ldots, p_N$ are related as

$$\begin{aligned} \begin{bmatrix} c_{i_1} \\ c_{i_2} \\ \ldots \\ c_{i_{M'}} \end{bmatrix} &= \begin{bmatrix} a_{i_1,1} & a_{i_1,2} & \ldots & a_{i_2,N} \\ a_{i_2,1} & a_{i_2,2} & \ldots & a_{i_2,N} \\ \vdots & \vdots & & \vdots \\ a_{i_{M'},1} & a_{i_{M'},2} & \ldots & a_{i_{M'},N} \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ \ldots \\ p_N \end{bmatrix} \\ &= A \begin{bmatrix} p_1 \\ p_2 \\ \ldots \\ p_N \end{bmatrix}. \end{aligned}$$

Note that A is a $M' \times N$ matrix. We have that $p_1, \ldots, p_N$ can be recovered from $c_{i_1}, \ldots, c_{i_{M'}}$ if and only if A has rank N.

1) What is the minimum M' below which it is not possible to recover $p_1, \ldots, p_N$?

If A has rank N, we can select N linearly independent rows of it and obtain a $N \times N$ matrix A' of rank N (note that A' is therefore invertible).

2) How can we use A' to recover $p_1, \ldots, p_N$ from the received packets $c_{j_1}, \ldots, c_{j_N}$ corresponding to the chosen rows of A'?

Assume now that we want to transmit packets p_1, p_2, p_3 and that we choose to adopt a FEC strategy sending 5 packets $c_1, \ldots, c_5$ obtained as follows

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix}.$$

3) If we receive c_1, c_2, c_4, can we recover p_1, p_2, p_3? If so, explain how, if not, explain why.

4) If we receive c_1, c_2, c_4, c_5, can we recover p_1, p_2, p_3? If so, explain how, if not, explain why.

5) Assume that each of the packets $c_1, \ldots, c_5$ get lost with probability ρ independently of the other. What is the probability that p_1, p_2, p_3 can be recovered?

What is the minimum M' below which it is not possible to recover $p_1, \ldots, p_N$?

SOLUTION:

1) We need to receive $M' \geq N$ packets (so the minimum is N). If $M' < N$ we cannot build a $N \times N$ matrix A' which is invertible and allows to recover the original packets. If $M' \geq N$, then the packets can be recovered if and only if there is a $N \times N$ submatrix A' of A (obtained choosing N linearly independent rows of A) that is invertible.

2) How can we use A' to recover $p_1, \ldots, p_N$ from the received packets $c_{j_1}, \ldots, c_{j_N}$ corresponding to the chosen rows of A'? Let $\mathbf{c}$ be the vector of the N received packets corresponding to the selected row in A' (which is invertible and has inverse $(A')^{-1}$). Let $\mathbf{p}$ be the vector of the original packets. Then $\mathbf{p}$ can be recovered as

$$\mathbf{p} = (A')^{-1}\mathbf{c}.$$

3) If we receive c_1, c_2, c_4, can we recover p_1, p_2, p_3? If so, explain how, if not, explain why.

The matrix A corresponding to the received packets c_1, c_2, c_4 is

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix}$$

The matrix has rank 2, and thus it is not invertible. Therefore, p_1, p_2, p_3 cannot be recovered.

4) If we receive c_1, c_2, c_4, c_5, can we recover p_1, p_2, p_3? If so, explain how, if not, explain why.

The matrix A corresponding to the received packets c_1, c_2, c_4, c_5 is

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

The matrix has rank 3. We can choose the rows of A the received packets c_1, c_2, c_6, obtaining

$$A' = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix},$$

which is, of course, invertible, and can be used to recover p_1, p_2, p_3.

5) Assume that each of the packets $c_1, \ldots, c_5$ get lost with probability ρ independently of the other. What is the probability that p_1, p_2, p_3 can be recovered?

Consider the case of 3 received packets out of $c_1, \ldots, c_5$. Out of the $\binom{5}{3=10}$ possibilities, only 2 choices do not allow to build a matrix with rank 3 and therefore do not allow to recover p_1, p_2, p_3. These cases are c_1, c_2, c_4 c_2, c_3, c_5. The other 8 possibilities lead to correct recovery p_1, p_2, p_3. Given a set of 3 packets, the event of receiving these 3 packets and not receiving the other 2 packets has probability $(1-\rho)^3\rho^2$. Therefore the probability of the 8 possibilities is $8(1-\rho)^3\rho^2$.

Consider the case of 4 received packets out of $c_1, \ldots, c_5$. Out of the $\binom{5}{4=5}$ possibilities, all lead to correct recovery p_1, p_2, p_3. Given a set of 4 packets, the event of receiving these 4 packets and not receiving the other packet has probability $(1-\rho)^4\rho$. Therefore the probability of the 8 possibilities is $5(1-\rho)^4\rho$.

Consider the case of 5 received packets out of $c_1, \ldots, c_5$. This possibility leads to correct recovery p_1, p_2, p_3, and has probability $(1-\rho)^5$.

The probability of successful recovery is therefore

$$P_{recovery} = 8(1-\rho)^3\rho^2 + 5(1-\rho)^4\rho + (1-\rho)^5.$$

Problem 3. In class we saw that power law distributions have been proposed to model the popularity of websites, the degree distribution of the hyperlink structure of the World Wide Web, the connections between routers in the Internet. In this problem we will discover the close relationship between one of such distributions (called the Pareto distribution) and the Exponential distribution.

A random variable X has the Pareto distribution with scale parameter $x_m > 0$ and index $\alpha > 0$ if its Cumulative Distribution Function is given by

$$\begin{aligned} F_p(x) =& \Pr(X \le x) = 1 - \left(\frac{x_m}{x}\right)^\alpha, && x \ge x_m \\ F_p(x) =& 0, && x < x_m. \end{aligned}$$

1) What it the probability density function $p_p(x)$ of X? SOLUTION: Taking the derivative of $F_p(x)$ we get $p_p(x) = \alpha(x_m)^\alpha x^{-\alpha-1}$ for $x \ge x_m$ and 0 otherwise.

2) Show that the probability density function $p_p(x)$ of X integrates to 1. SOLUTION: Compute the integral.

3) Let the survival function of X be $\bar{F}_p(x) = \Pr(X > x)$. What is $\bar{F}(x)$? SOLUTION: $\bar{F}(x) = \left(\frac{x_m}{x}\right)^\alpha$ for $x \ge x_m$, 1 otherwise.

4) For which values of α is $E[X]$ (the expected value of X) finite? SOLUTION: Compute the expectation and show that if $\alpha > 1$ then the integral is finite and $\mathbb{E}(X) = \frac{\alpha x_m}{\alpha - 1}$. If $\alpha \le 1$ then the integral is infinite.

5) What happens to the distribution of X when $\alpha \to \infty$? SOLUTION:

$$p_p(x) = \alpha(x_m/x)^\alpha x^{-1} \quad \text{for x} \geq x_m$$

As $\alpha \to \infty$ the above expression is infinite when $x = x_m$ while it is zero for $x > x_m$. For $x < x_m$ the pdf is zero. Since it needs to sum to one, it follows that the pdf tends to a Dirac impulse centered ata $x = x_m$.

A variable Y has the Exponential distribution with rate λ if its Cumulative Distribution Function is $F_e(y) = \Pr(Y \leq y) = 1 - e^{-\lambda y}$ for all $y \geq 0$, and $F_e(y) = 0$ for $y < 0$. Let X have a Pareto distribution with parameters x_m and α. Let $Z = \log(X/x_m)$, where log denotes the natural logarithm.

6) Show that $\Pr(Z < z) = \Pr(X < x_m e^z)$.

7) From the last answer, conclude that Z is Exponential with rate α.

Problem 4. Consider two different links that can be modeled as M/M/1 queues, denoted respectively as Q1 and Q2. Q1 has a server with service rate $\mu_1 = 200$ packets per second (in this problem, we assume all packets have the same length). Q2 has a server with service rate $\mu_2 = 300$ packets per second. There is a single incoming packet flow with arrival rate $\lambda = 400$ packets per second, and we need to split it among Q1 and Q2 in a fair way. λ_1 will denote the flow sent to Q1, while λ_2 will denote the flow sent to Q2.

Part I. Observing that $\mu_2 = 3\mu_1/2$, we try to split the incoming flow in the same proportion, that is, $\lambda_2 = 3\lambda_1/2$. We get $\lambda_1 = 160$ packets per second, $\lambda_2 = 240$ packets per second.

a) Compute the load factors ρ_1 and ρ_2 of Q1 and Q2.

$$\rho_1 = \frac{\lambda_1}{\mu_1} = 160/200 = 0.8$$
$$\rho_2 = \frac{\lambda_2}{\mu_2} = 240/300 = 0.8.$$

b) Compute the average time T_1 that a packet spends in Q1, and the average time T_2 that a packet spends in Q2.

$$T_1 = \frac{1}{\mu_1 - \lambda_1} = \frac{1}{200 - 160} = 25ms$$
$$T_2 = \frac{1}{\mu_2 - \lambda_2} = \frac{1}{300 - 240} = 16.7ms$$

c) In your opinion, was the traffic flow split in a fair way between Q1 and Q2?

No, even if the load factor is the same in the two queue, the average times that packets spends in them are different.

Part II. We want to split the traffic flow (λ_1 to Q1, λ_2 to Q2) in such a way that the average times T_1 and T_2 are equal.

d)Write T_2 as a function of μ_2 and λ_1.

$$T_2 = \frac{1}{\mu_2 - (\lambda - \lambda_1)}$$

e) Compute the value of λ_1 such that $T_1 = T_2$.

$$\frac{1}{\mu_1 - \lambda_1} = \frac{1}{\mu_2 - (\lambda - \lambda_1)}$$

$$\mu_2 - (\lambda - \lambda_1) = \mu_1 - \lambda_1$$

$$\lambda_1 = (\mu_1 - \mu_2 + \lambda)/2 = 150 \text{ packets per second}$$

f) Compute the corresponding value of λ_2.

$$\lambda_2 = \lambda - \lambda_1 = 250 \text{ packets per second}$$

g) Compute the load factors ρ_1 and ρ_2 of Q1 and Q2.

$$\rho_1 = \frac{\lambda_1}{\mu_1} = 150/200 = 0.75$$
$$\rho_2 = \frac{\lambda_2}{\mu_2} = 250/300 = 0.833$$

h) Compute the average time T_1 that a packets spends in Q1, and the average time T_2 that a packets spends in Q2.

$$T_1 = \frac{1}{\mu_1 - \lambda_1} = \frac{1}{200 - 150} = 20ms$$
$$T_2 = \frac{1}{\mu_2 - \lambda_2} = \frac{1}{300 - 250} = 20ms$$

i) In terms of average time a packet spends in the system, is it better to have the traffic split between Q1 and Q2 (as we just computed), or to have the entire flow λ sent to a queue Q3 with service rate $\mu_3 = \mu_1 + \mu_2 = 500$ packets per second. As part of your answer, you need to compute the average time T_3 a packets would spend in Q3.

$$T_3 = \frac{1}{\mu_3 - \lambda} = \frac{1}{500 - 400} = 10ms. \text{ This is better}$$

Part III. We want to split the traffic flow (λ_1 to Q1, λ_2 to Q2) in a way that $T_1 = T_2/2$.

j) Find $\lambda_1, \lambda_2, \rho_1, \rho_2, T_1, T_2$ in this case. You must choose λ_1 and λ_2 to be integer, but such that $|T_1 - T_2/2| < 0.2ms$.

$$\frac{1}{\mu_1 - \lambda_1} = \frac{1}{2}\frac{1}{\mu_2 - (\lambda - \lambda_1)}$$

$$2\mu_2 - 2(\lambda - \lambda_1) = \mu_1 - \lambda_1$$

$$\lambda_1 = (\mu_1 - 2\mu_2 + 2\lambda)/3 = 133 \text{ packets per second}$$
$$\lambda_2 = \lambda - \lambda_1 = 267 \text{ packets per second}$$

$$\rho_1 = \lambda_1/\mu_1 = 133/200 = 0.665$$
$$\rho_2 = \lambda_2/\mu_2 = 267/300 = 0.89$$

$$T_1 = \frac{1}{\mu_1 - \lambda_1} = \frac{1}{200 - 133} = 14.9ms$$
$$T_2 = \frac{1}{\mu_2 - \lambda_2} = \frac{1}{300 - 267} = 30ms$$