

Exam 1 (Midterm), Fall 1999
Computer Science 516
Computer Communication Networks
University at Albany

Prof. William A. Maniatty

October 26, 1999

1 Some Hints

Your professor suggests the following preparation strategies:

- This exam is open book and open notes (your own books and notes that is!). Calculators are permitted, networked devices are not.
- Write neat clean answers, since if the grader cannot understand you on the real exam, it will go badly for you.
- Show your work, if you are guessing the grader will not give much credit (even if you get lucky and guess right).
- Define your notation (you can use tables like the lecture notes if you like).
- Set up the solution symbolically and simplify before plugging numbers in, it is easier to follow for the grader.
- You can solve problems out of order, but keep the work for each problem in one place, and mark it clearly.

2 The Problems

1. Fourier Analysis (15 Points): Consider a Unipolar encoding of 8 bits, where the maximum voltage is $+V$ transmitted over a period of $1\mu\text{sec}$. Also recall that all periodic functions, $g(t)$, can be described using a summation of sine and cosine functions:

$$g(t) = \frac{C}{2} + \left[\sum_{n=1}^{\infty} a_n \sin(2\pi nft) \right] + \left[\sum_{n=1}^{\infty} b_n \cos(2\pi nft) \right] \quad (1)$$

- (a) (5 Points) The ascii code for the letter 'a' (the grade you want!) is encoded '`\0x61`'. Show your derivation for the value of the term C in the Fourier series.

First plot the unipolar encoding, it should look like Figure 1. From our given, we can assume have a period of $T = 1\mu\text{sec}$.

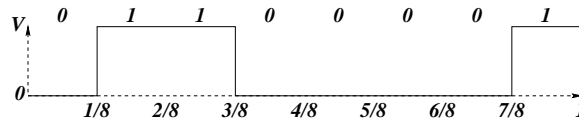


Figure 1: The unipolar encoding of 'a' = $0x61 = 01100001_2$ in Problem 1a

We denote this function as $g(t)$, and from our given, $g(t)$ has a period of $T = 1\mu\text{sec}$. We then can compute C as:

$$C = \frac{2}{T} \int_0^T g(t) dt \quad (2)$$

$$= \frac{2}{T} \left[\int_{\frac{1}{8}}^{\frac{3}{8}} V dt + \int_{\frac{6}{8}}^{\frac{7}{8}} V dt \right] \quad (3)$$

$$= 2 \left[\frac{2V}{8} + \frac{V}{8} \right] = \frac{3V}{4} \quad (4)$$

- (b) (10 Points) The ascii code for the letter 'b' is encoded '`\0x62`'. Show your derivation for b_n (hint it is a function of n) in the Fourier series.

Again, we first begin by drawing the unipolar encoding, as shown in Figure 2. From our given, we can assume have a

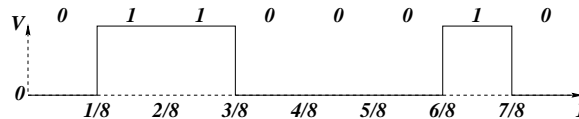


Figure 2: The unipolar encoding of 'b' = $0x62 = 01100010$ in Problem 1b

period of $T = 1\mu\text{sec}$. We denote this function as $g(t)$, and from our given, $g(t)$ has a period of $T = 1\mu\text{sec}$. To make things easier, just like in the example in the lecture notes, we can substitute $\beta = 2\pi n$, and $u = \beta t$. This means $\frac{du}{dt} = \beta dt$ and $dt = \frac{du}{\beta}$ which is also used in the solution. We then can compute b_n as:

$$b_n = \frac{2}{T} \int_0^T g(t) \cos(2\pi nt) dt \quad (5)$$

$$= 2 \left[\int_{\frac{1}{8}}^{\frac{3}{8}} V \cos(2\pi nt) dt + \int_{\frac{5}{8}}^{\frac{7}{8}} V \cos(2\pi nt) dt \right] \quad (6)$$

$$= 2V \left[\int_{\frac{1}{8}}^{\frac{3}{8}} \cos(u) dt + \int_{\frac{5}{8}}^{\frac{7}{8}} \cos(u) dt \right] \quad (7)$$

$$= \frac{2V}{\beta} \left[\int_{\frac{1}{8}}^{\frac{3}{8}} \cos(u) du + \int_{\frac{5}{8}}^{\frac{7}{8}} \cos(u) du \right] \quad (8)$$

$$= \frac{2V}{\beta} \left[\left[\sin\left(\frac{3u}{8}\right) - \sin\left(\frac{u}{8}\right) \right] + \left[\sin\left(\frac{7u}{8}\right) - \sin\left(\frac{6u}{8}\right) \right] \right] \quad (9)$$

$$= \frac{V}{\pi n} \left[\left[\sin\left(\frac{3\pi n}{4}\right) - \sin\left(\frac{\pi n}{4}\right) \right] + \left[\sin\left(\frac{7\pi n}{4}\right) - \sin\left(\frac{6\pi n}{4}\right) \right] \right]$$

2. Channel Performance Analysis (15 points): Suppose you are a developer and you want to upgrade a remote software installation as soon as possible, and it is now 2 p.m. The upgrade requires transferring 10 gigabytes of programs and data to your customer's machine. You can send the data over a 100 Mbps network, but you only get 1% of that bandwidth on average. Otherwise, you could transfer the data to a tape and have an overnight delivery arrive at the customer site by 10:00 A. M. tomorrow. It takes two additional hours to read the tape into the customer's machine after it arrives. You can assume you and your customer are in the same time zone. Which solution will get your customer up and running faster, the overnight delivery service, or sending the files over the wire.

In this problem we want to select the minimum of two times:

- (a) The time to transfer the data over a wire, which can be solved using:

$$\text{Wire Transfer Time} = \frac{\text{Message Size}}{\text{Effective Bandwidth}} \quad (10)$$

$$\text{Effective Bandwidth} = \frac{100 \times 10^6 \text{bits}}{\text{sec}} \times \frac{0.01 \text{ } 10^6 \text{bits}}{= \text{sec}} \quad (11)$$

$$\text{Wire Transfer Time} = \frac{10 \times 10^9 \text{bytes} \times \frac{8 \text{bits}}{\text{byte}}}{\frac{10^6 \text{bits}}{\text{sec}}} = 8000 \text{sec} \approx 22.2 \text{h} \quad (12)$$

- (b) The time to transfer the data using overnight delivery can be solved:

$$\begin{aligned} \text{Mail Transfer Time} &= \text{Shipping Time} + \text{Tape Reading Time} \\ &= 20\text{hours} + 2\text{hours} = 22\text{hours}. \end{aligned} \quad (14)$$

And we see that

$$\min(\text{Wire Transfer Time}, \text{Mail Transfer Time}) = \text{Mail Transfer Time} = 22\text{hours} \quad (15)$$

So using the overnight delivery is faster!

3. Multiplexing and ATM (20 Points): Consider the network shown in Figure 3, and recall that ATM has 155 Mbps, and cells arrive in the configuration shown.

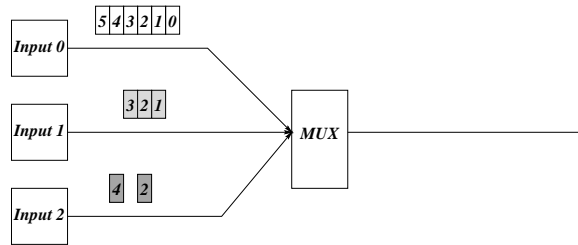


Figure 3: Multiplexing and ATM system from problem 3

- (a) (10 Points) Suppose that a time division multiplexer is used.
- i. (5 Points) What is the minimum bandwidth of the multiplexed link?

Since the output link has to be fast enough to handle all inputs:

$$\begin{aligned} \text{Min. Output Bandwidth} &= \sum_{i=1}^{\text{number of inputs}-1} \text{Max Input Bandwidth}_i \quad (16) \\ &= 155\text{Mbps} + 155\text{Mbps} + 155\text{Mbps} = 465\text{Mbps} \end{aligned}$$

- ii. (5 Points) What is the utilization of the multiplexed link's bandwidth?

To do this, we assume that the output link is allocated the minimum necessary bandwidth (typically done to keep costs down). Utilization, ρ , is then determined by counting the number of incoming packets, and dividing

by the number of input packets that could have arrived (had data been available).

$$\rho = \frac{6 + 3 + 2}{6 + 6 + 6} = \frac{11}{18} \quad (18)$$

(b) (10 Points) Suppose that a statistical multiplexer is used and that the traffic rates shown are representative of the average rate of traffic, and that the multiplexed link has 80% of the bandwidth you computed in problem 3(a)i.

i. (5 Points) How fast must the link be to handle average traffic, and is the proposed link fast enough (derive, do not guess!)?

Using the utilization computed above, and applying the definition of utilization, we get:

$$\begin{aligned} \text{Required bandwidth} &= \rho \times \sum_{i=1}^{\text{number of inputs}-1} \text{Max Input Bandwidth} \\ &= \frac{11}{18} \times 465 \text{Mbps} \approx 284 \text{Mbps} \end{aligned} \quad (20)$$

We can see that is less than the provisioned bandwidth capacity of $0.8 \times 465 \text{Mbps} = 372 \text{Mbps}$. Another fast check that could be done would have been to see if the actual utilization would have been greater than 0.8.

ii. (5 Points) What is the number of buffers (one buffer per 53 byte cell) required for a burst on all inputs lasting 0.5 seconds?

We want to solve for:

$$\text{buffers needed} = 3 \times \text{input burst size} - (\text{duration} \times \text{output bandwidth}) \quad (21)$$

Each input link can process (at maximum throughput):

$$\begin{aligned} \text{input burst size} &= \text{duration} \times \text{input bandwidth} \quad (22) \\ &= 0.5 \text{sec} \times \frac{155 \times 10^6 \text{bits}}{\text{sec}} \times \frac{1 \text{packet}}{53 \text{bytes} \times \frac{8 \text{bits}}{\text{byte}}} \end{aligned}$$

$$\approx 1.82 \times 10^5 \text{packets} \times \frac{1 \text{buffer}}{1 \text{packet}} \quad (24)$$

$$= 1.82 \times 10^5 \text{buffers} \quad (25)$$

I normalized the measurements to buffers to make life easy. The bandwidth of the output (in packets per second) is:

$$\text{output bandwidth} = \frac{372 \times 10^6 \text{bytes}}{\text{sec}} \times \frac{1 \text{packet}}{53 \text{bytes} \times \frac{8 \text{bits}}{1 \text{byte}}} \approx \frac{8.77 \times 10^5 \text{packets}}{\text{sec}} \quad (26)$$

Substituting our results from equation (26) and our given into equation (21) yields:

$$\begin{aligned} \text{buffers needed} &= 5.46 \times 10^5 \text{packets} - \left(0.5 \text{sec} \times \frac{8.77 \times 10^5 \text{packets}}{\text{sec}} \times \frac{1 \text{buffer}}{1 \text{packet}}\right) \\ &= 5.46 \times 10^5 \text{buffers} - 4.38 \times 10^5 \text{buffers} = 1.10 \times 10^5 \text{buffers} \end{aligned}$$

4. Protocol Layering (10 points):

- (a) (5 points) How is randomization used in ethernet protocols (at the data link layer), and how do token ring networks avoid randomization?

Recall our in class discussion of this prior to the exam. Ethernet uses carrier sense multiple access (CSMA) with exponential backoff. However, in order to prevent collision on the retries, a small random increment is added to the timer of each node involved in a collision. Token ring does not use this, since ownership of the token is a prerequisite to transmission (as per the protocol).

- (b) (5 Points) Show the PPP frame containing the following C literal as the data sent using the IP protocol

"I deserve \0x7d an \0x7e A on \0xC0 this test!"

Where \0x7e, \0x7d and \0xC0 are the corresponding byte values injected into the stream of text.

flag	addr	control	protocol	information	crc	flag
0x7e	0xff	0x03	0x0021	I deserve \0x7d\0x5e an \0x7d\0x5d A on \0xC0 this test		0x7e

5. Design principles and Queuing Theory (10 points): Suppose that a network interface card (NIC) implements the IP layer of the TCP/IP protocol in hardware (using a dedicated processor). Assume that when a packet is received it takes $5\mu\text{sec}$ (on average) to process the headers and trailers of received packets at the IP layer, and $12\mu\text{sec}$ (on average) for the software drivers to process the headers for the host to host layer.

- (a) (5 points) What is the maximum throughput of this system for received data if both the NIC hardware and host-to-host software can process packets in constant time?

The system is pipelined, as seen in Figure 4. Since the throughput of a pipeline is equal to the throughput of its slowest stage:

$$\begin{aligned} \text{Max Packet Throughput} &= \min(\text{NIC Throughput}, \text{TCP Throughput}) \\ &= \min\left(\frac{1 \text{packet}}{5 \mu\text{sec}}, \frac{1 \text{packet}}{12 \mu\text{sec}}\right) \end{aligned} \tag{30}$$

$$= \frac{1 \text{packet}}{12 \mu\text{sec}} \tag{31}$$

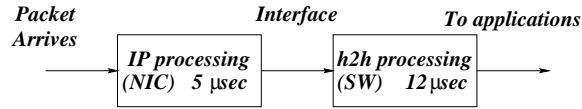


Figure 4: Pipelined System described in Problem 5a

- (b) If packets arrive every $20\mu\text{sec}$ on average (exponentially distributed), and the IP and host-to-host layers have an exponential distribution, what is the mean time to service a packet (5 points)?

The system is as diagrammed in Figure 5. Note that for

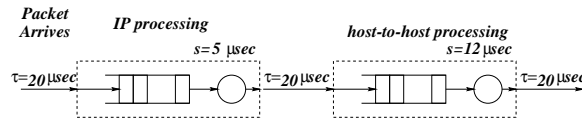


Figure 5: Queuing System described in Problem 5b

stability, the interarrival times, $\tau = \frac{1}{\lambda} = 20\mu\text{sec}$ out of each stage must match. If we call t_{ip} the mean time spent in the IP processing stage, and the host-to-host layers take time t_{h2h} :

$$t_q = t_{ip} + t_{h2h} \quad (32)$$

$$= \frac{s_{ip}}{1 - \rho_{ip}} + \frac{s_{ip}}{1 - \rho_{ip}} \quad (33)$$

$$= \frac{s_{ip}}{1 - \frac{s_{ip}}{\tau}} + \frac{s_{ip}}{1 - \frac{s_{ip}}{\tau}} \quad (34)$$

$$= \frac{5\mu\text{sec}}{1 - \frac{5\mu\text{sec}}{20\mu\text{sec}}} + \frac{12\mu\text{sec}}{1 - \frac{12\mu\text{sec}}{20\mu\text{sec}}} \quad (35)$$

$$= \frac{20}{3}\mu\text{sec} + 30\mu\text{sec} = \frac{110}{3}\mu\text{sec} \quad (36)$$

6. Network Systems programming (10 Points):

- (a) (5 points) Why do many flavors of unix support scatter/gather programming using readv and writev systems calls?

Scatter/Gather programming allows us to replace multiple systems calls doing data transfer between discontinuous data locations and a contiguous data location with a single system call. This reduces the number of systems calls, which improves performance. Additionally the systems programmers can optimize across multiple systems calls (e.g. by prefetching data).

(b) (5 points) What functionality does the connect system call provide?

The connect system call is part of the Unix (BSD) sockets interface. Connect is called by a client to create a mapping (connection) from a client side socket to a server side socket. If the connect call succeeds, data written to the socket will be sent to the server, and data read from the socket will be received from the server.