CS/ECE 438: Communication Networks Fall 2006
Midterm Exam Solutions

*The average was 68.67 and the standard deviation was 12.19. Each question includes the average (μ), standard deviation (σ) and the correlation between the grade on the question and the grade on the rest of the exam (*corr*).*

1. **Network Architecture (19 points)** ($\mu = 15.74, \sigma = 2.92, \text{corr} = 0.45$)

   (a) **(4 points)** ($\mu = 3.23, \sigma = 0.99, \text{corr} = 0.29$) List two benefits of splitting the network architecture into separate layers.

   Some benefits are:

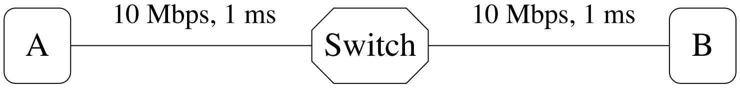
   - *Reduce complexity of each laer by abstracting away details of other layers*
   - *Support multiple compatible implementations of upper or lower layers through a well-defined interface*
   - *(related) Allow layers to be replaced and redesigned*

   Note: Some people said something along the lines of simplifying the task of application builders. This is a benefit of providing a network abstraction, but you can provide the same benefit without splitting up the network stack into several layers.

   (b) **(3 points)** ($\mu = 2.49, \sigma = 1.00, \text{corr} = 0.11$) Since nearly all Ethernet networks thesee days carry IP traffic, we could consider changing the Ethernet standard and combining it with the IP layer. What advantage might this change have?

   You could save space on headers by eliminating redundant fields and you can eliminate ARP.

   (c) **(5 points)** ($\mu = 4.41, \sigma = 1.09, \text{corr} = 0.33$) Consider two hosts, A and B, connected by a store-and-forward switch. The links between each host and the switch have a bandwidth of 10 Mbps and a propagation delay of 1ms. What is the one-way latency (first bit sent to last bit received) for sending a 1000-bit packet from A to B?



Latency from A to switch is transmission time + propagation time, i.e. 1000bit / 10 Mbps + 1ms = 1.1ms. Latency from switch to B is the same, for a total of 2.2ms.

(d) **(7 points)** ($\mu = 5.62, \sigma = 1.57, \text{corr} = 0.43$) Suppose that after A sends a packet to B, it waits for a 100-bit acknowledgment to be sent from B to A before sending the next packet. What is the long-term effective bandwidth for sending data from A to B?

The latency to transmit the acknowledgment back from B to A is $2 * (0.01ms + 1ms) = 2.02ms$. *So the total time to send 1000 bits is* $4.22ms$ *and the effective bandwidth is* $1000bits/4.22ms \approx 236Kbps$. *(Leaving it as 1000/4.22 was enough to get full marks.)*

2. **Encoding and Framing (15 points)** ($\mu = 7.56, \sigma = 4.52, \text{corr} = 0.6$)

(a) **(3 points)** ($\mu = 2.46, \sigma = 1.00, \text{corr} = 0.57$) 4B/5B encoding expands data by 25%. What is the benefit of using such an inefficient encoding?

Main reason for 4B/5B encoding is to eliminate long strings of 0's, which are a problem for the NRZI encoding. It has a couple of side benefits, such as using the unused 5B sequences for framing, and being able to detect some errors when invalid sequences show up.

However, if you mentioned only error detection, you got no credit, because 4B/5B is not an effective error detection scheme.

(b) **(5 points)** ($\mu = 2.82, \sigma = 2.00, \text{corr} = 0.60$) Consider the BISYNC framing, with STX used to signal the start of a frame and ETX used to signal the end. Supposing the frames contain a uniform distribution of characters in the range 0–255, calculate the average overhead for sending a frame of size n.

In BISYNC, you have to escape ETX as DLE–ETX and DLE as DLE–DLE. (You don't have to escape STX.) Therefore, for 2 out *of 256 characters, you add one extra byte. On average, you will stuff* $2n/256$ *bytes, plus the overhead for the two framing bytes (STX and ETX) gives the answer of* $2 + 2n/256$.

(c) **(7 points)** ($\mu = 2.28, \sigma = 2.49, \text{corr} = 0.56$) At what frame size does the original IMP–IMP encoding, which uses DLE–STX to signal the beginning of a frame and DLE–ETX to signal the end, become more efficient?

IMP–IMP only needs to escape DLE inside the data stream, so it stuffs $n/256$ *extra bytes on average, but has a fixed overhead of 4 bytes per frame, for a total of* $4 + n/256$. *Solving* $4 + n/256 < 2 + 2n/256$ *gives us* $n > 512$.

3. **Multiple Access (18 points)** ($\mu = 13.36, \sigma = 3.52, \text{corr} = 0.51$)

(a) **(6 points)** ($\mu = 3.95, \sigma = 1.45, \text{corr} = 0.40$) List three characteristics of an ideal MAC protocol

Given a total bandwidth of R:

- *When one node wants to transmit, it gets full bandwidth of R.*
- *When M nodes want to transmit, each gets bandwidth of close to R/M.*
- *Simple*
- *No centralization (including clock synchronization)*

Note that the second condition captures both fairness (each node gets equal bandwidth) and efficiency under contention.

(b) **(4 points)** ($\mu = 2.74, \sigma = 1.25, \text{corr} = 0.30$) Which of these characteristis are **not** satisfied by the slotted ALOHA protocol?

Nearly all of them! The most important factor is that under contention with M nodes, each node receives much less than R/M bandwidth because of low efficiency of the slotted ALOHA scheme. And it requires clock synchronization. Arguably, it does not satisfy the single-node efficiency requirement because framing constraints might result in a single node using less than the full available bandwidth for useful data, though if you did not mention this in your answer, you did not lose points.

(c) **(3 points)** ($\mu = 2.44, \sigma = 1.02, \text{corr} = 0.60$) Suppose the Ethernet MAC is modified to use a linear instead of an exponential backoff. That is, after k collisions, each node picks a random value between 0 and k (inclusive), rather than between 0 and $2^k - 1$. Would this make the Ethernet capture effect more or less likely?

Less likely. The Ethernet capture effect results because the range of values for k used by a node after a successful transmission is much *smaller than by a node who has seen several collisions without a successful frame being sent. With linear backoff, the difference is much less. E.g after* $k = 5$ *collisions, the node that "won" the previous rounds picks between* 0 *and* 1*, and the loser picks between* 0 *and* 5 *with linear backoff, but between* 0 *and* 31 *with exponential backoff.*

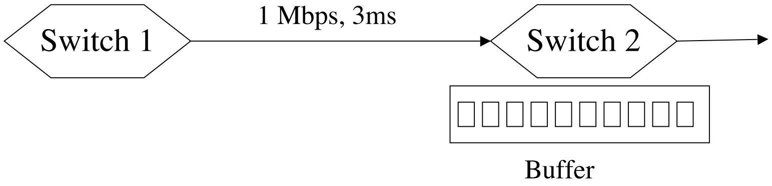
(d) **(5 points)** ($\mu = 4.23, \sigma = 1.65, \text{corr} = 0.25$) The minimum packet size for 10 Mbps Ethernet is 512 bits. Assuming that 100 Mbps Ethernet operated on networks with the same maximum propagation delay, what should the minimum packet size be in this network?

The Ethernet MAC requires that the transmission time for the minimum packet be greater than the max RTT on the network. Since 100 Mbps Ethernet has transmission times that are 10 times as fast, the packet needs to be 10 times as large, or 5120 bits.

4. Bridges, Switches, and Forwarding (19 points) ($\mu = 14.62, \sigma = 4.16, \text{corr} = 0.42$)

(a) **(7 points)** ($\mu = 4.08, \sigma = 2.40, \text{corr} = 0.30$) Suppose two switches use back pressure to control the flow of packets. Switch #1 is sending packets to switch #2, the packets are 1000 bits each, the link connecting the two switches has a bandwidth of 1 Mbps and propagation delay of 3 ms.

Switch #2 is forwarding to another link; when that link is congested, it buffers packets in a buffer big enough to store 10 packets. At what point should switch #2 send a back pressure signal to switch #1?



What we want to calculate is how many packets will arrive at switch #2 after the time it sends the back pressure signal. The signal will take 3ms to propagate to switch #1, during which time #2 will receive 3 packets. (Assuming that the back pressure signal is of negligible size.) At this point, #1 will stop sending, but there will still be 3 packets in flight over the link between the switches (bandwidth x delay product is 3000 bits or 3 packets in this case.) Therefore, after sending the signal, switch #2 may still receive 6 more packets from switch #1, so it should send the back pressure signal when the buffer has 4 full slots.

(b) **(12 points)** ($\mu = 10.54, \sigma = 2.85, \text{corr} = 0.41$) List one advantage and one disadvantage of each of the following forwarding schemes:

Some of these advantages / disadvantages are flips of the others; we allowed such answers on the midterm.

i. Datagram forwarding

+ *Can dynamically route around failures*
+ *Can send data immediately*
- *Header requires full globally unique address*
- *Might not be possible to deliver packets*
- *Successive packets may not follow the same route*
- *Global address to path translations requires significant storage*

ii. Virtual circuit forwarding

+ *Header contains only (small) VC ID, not full address*
+ *Can reserve resources for a connection at setup*
- *Usually have to wait one RTT before sending data*
- *Cannot dynamically route around failures*

    - *Must maintain (large) global address path information as well as per-connection state*

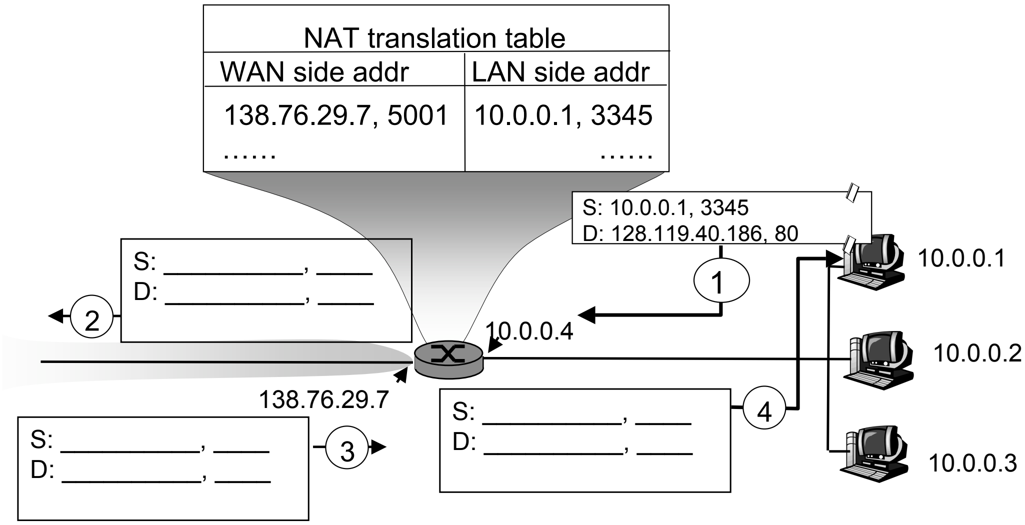
  iii. Source routing

    - \+ *Simple switches*
    - \+ *Fast forwarding*
    - \+ *Source controls route*
    - \- *Hosts must know entire topology*
    - \- *Headers can be quite large*
    - \- *Changes in link state must propagate to all hosts*

5. IP (19 points) ($\mu = 17.38, \sigma = 2.14, \text{corr} = 0.48$)

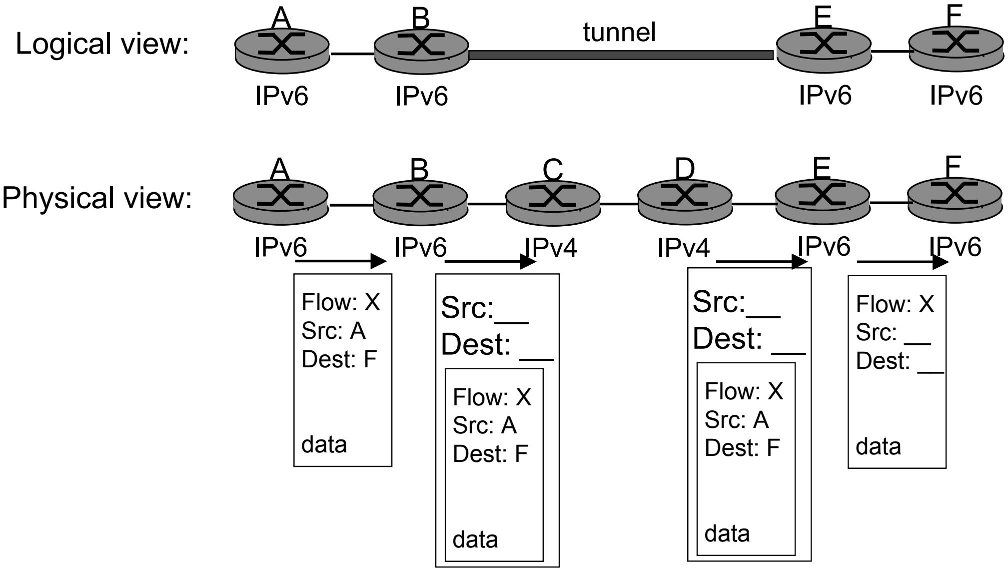
(a) **(6 points)** ($\mu = 5.85, \sigma = 0.54, \text{corr} = 0.35$) The diagram below shows a packet traveling through a NAT router. Packet 1 is sent from the internal host to the NAT router, packet 2 is sent from the NAT router to the external web server, packet 3 is received from the web server by the NAT router, and packet 4 is sent by the NAT router to the original host.

Fill in the missing source and destination IP addresses and port numbers in packets 2–4.



2: S: 138.76.29.7, 5001
D: 128.119.40.186, 80
3: S: 128.119.40.186, 80
D: 138.76.29.7, 5001
4: S: 128.119.40.186, 80
D: 10.0.0.1, 3345

(b) **(6 points)** ($\mu = 5.26, \sigma = 1.29, \text{corr} = 0.32$) The diagram below shows an IPv6 packet tunneled over IPv4. Fill in the missing source and destination addresses.



Packet 1: Src = B, Dest = E
Packet 2: Src = B, Dest = E
Packet 3: Src = A, Dest = F

(c) **(3 points)** ($\mu = 2.97, \sigma = 0.16, \text{corr} = -0.22$) What is the primary reason for transitioning to IPv6?

IPv4 is running out of IP addresses.

(d) **(4 points)** ($\mu = 3.31, \sigma = 1.26, \text{corr} = 0.39$) List the first and last address that is part of the 1.2.0.0/20 network. How many addresses are there in total in this network?

The first and last addresses are 1.2.0.1 and 1.2.15.254. 1.2.0.0 is reserved and 1.2.15.255 is usually the broadcast address for this subnet. The total number of addresses is $2^{32-20} - 2 = 4094$. *If you listed the two reserved addresses as part of the address range and/or the total number of addresses, you still received full marks.*