EE450: Sample Exercises

These are sample problems from other textbooks (These textbooks are posted on our side in the Reference folder). No solutions are available but will be happy to check your work during office hours

1)

Calculate the total time required to transfer a 1.5-MB file in the following cases, assuming an RTT of 80 ms, a packet size of 1 KB data, and an initial $2 \times$ RTT of "handshaking" before data is sent:

(a) The bandwidth is 10 Mbps, and data packets can be sent continuously.

(b) The bandwidth is 10 Mbps, but after we finish sending each data packet we must wait one RTT before sending the next.

(c) The link allows infinitely fast transmit, but limits bandwidth such that only 20 packets can be sent per RTT.

(d) Zero transmit time as in (c), but during the first RTT we can send one packet, during the second RTT we can send two packets, during the third we can send four (2^{3-1}), etc. (A justification for such an exponential increase will be given in Chapter 6.)

2)

Consider a point-to-point link 50 km in length. At what bandwidth would propagation delay (at a speed of 2×10^8 m/s) equal transmit delay for 100-byte packets? What about 512-byte packets?

3)

How "wide" is a bit on a 10-Gbps link? How long is a bit in copper wire, where the speed of propagation is 2.3×10^8 m/s?

4)

Suppose a 128-kbps point-to-point link is set up between the Earth and a rover on Mars. The distance from the Earth to Mars (when they are closest together) is approximately 55 Gm, and data travels over the link at the speed of light—3×10^8 m/s.

(a) Calculate the minimum RTT for the link.

(b) Calculate the delay $\times$ bandwidth product for the link.

(c) A camera on the rover takes pictures of its surroundings and sends these to Earth. How quickly after a picture is taken can it reach Mission Control on Earth? Assume that each image is 5 Mb in size.

5)

Calculate the latency (from first bit sent to last bit received) for:

(a) 1-Gbps Ethernet with a single store-and-forward switch in the path and a packet size of 5000 bits. Assume that each link introduces a propagation delay of 10 μs and that the switch begins retransmitting immediately after it has finished receiving the packet.

(b) Same as (a) but with three switches.

(c) Same as (b), but assume the switch implements "cut-through" switching; it is able to begin retransmitting the packet after the first 128 bits have been received.

6)

Calculate the delay × bandwidth product for the following links. Use one-way delay, measured from first bit sent to first bit received.

(a) 100-Mbps Ethernet with a delay of 10 μs.

(b) 100-Mbps Ethernet with a single store-and-forward switch like that of Exercise 16(b), packet size of 12,000 bits, and 10 μs per link propagation delay.

(c) 1.5-Mbps T1 link, with a transcontinental one-way delay of 50 ms.

(d) 1.5-Mbps T1 link between two groundstations communicating via a satellite in geosynchronous orbit, 35,900 km high. The only delay is speed-of-light propagation delay from Earth to the satellite *and back*.

7)

Hosts A and B are each connected to a switch S via 100-Mbps links as in Figure 1.21. The propagation delay on each link is

20 μs. S is a store-and-forward device; it begins retransmitting a received packet 35 μs after it has finished receiving it. Calculate the total time required to transmit 10,000 bits from A to B

(a) As a single packet.

(b) As two 5000-bit packets sent one right after the other.

8)

Suppose a host has a 1-MB file that is to be sent to another host. The file takes 1 second of CPU time to compress 50% or 2 seconds to compress 60%.

(a) Calculate the bandwidth at which each compression option takes the same total compression + transmission time.

(a) Explain why latency does not affect your answer.

9)

Assume you wish to transfer an n B file along a path composed of the source, destination, 7 point-to-point links, and 5 switches. Suppose each link has a propagation delay of 2 ms and a bandwidth of 4 Mbps, and that the switches support both circuit and packet switching. Thus, you can either break the file up into 1-KB packets or set up a circuit through the switches and send the file as one contiguous bitstream. Suppose that packets have 24 B of packet header information and 1000 B of payload, store-and-forward packet processing at each switch incurs a 1-ms delay after the packet had been completely received, packets may be sent continuously without waiting for acknowledgments, and circuit setup requires a 1-KB message to make one round trip on the path, incurring a 1-ms delay at each switch after the message has been completely received. Assume switches introduce no delay to data traversing a circuit. You may also assume that filesize is a multiple of 1000 B.

(a) For what filesize n B is the total number of bytes sent across the network less for circuits than for packets?

(b) For what filesize n B is the total latency incurred before the entire file arrives at the destination less for circuits than for packets?

(c) How sensitive are these results to the number of switches along the path? To the bandwidth of the links? To the ratio of packet size to packet header size?

(d) How accurate do you think this model of the relative merits of circuits and packets is? Does it ignore important considerations that discredit one or the other approach? If so, what are they?

Typo in problem. It should be 6 switches

10)

Consider a network with a ring topology, link bandwidths of 100 Mbps, and propagation speed 2×10^8 m/s. What would the circumference of the loop be to exactly contain one 1500-byte packet, assuming nodes do not introduce delay? What would the circumference be if there was a node every 100 m, and each node introduced 10 bits of delay?

11)

Consider a simple protocol for transferring files over a link. After some initial negotiation, A sends data packets of size 1 KB to B; B then replies with an acknowledgment. A always waits for each ACK before sending the next data packet; this is known as *stop-and-wait*. Packets that are overdue are presumed lost and are retransmitted.

(a) In the absence of any packet losses or duplications, explain why it is not necessary to include any "sequence number" data in the packet headers.

(b) Suppose that the link can lose occasional packets, but that packets that do arrive always arrive in the order sent. Is a 2-bit sequence number (that is, N mod 4) enough for A and B to detect and resend any lost packets? Is a 1-bit sequence number enough?

(c) Now suppose that the link can deliver out of order and that sometimes a packet can be delivered as much as 1 minute after subsequent packets. How does this change the sequence number requirements?

12)

Suppose the following sequence of bits arrives over a link:

01101011111010100111111011001111110

Show the resulting frame after any stuffed bits have been removed. Indicate any errors that might have been introduced into the frame.

13)

Suppose we want to transmit the message 11100011 and protect it from errors using the CRC polynomial $x^3 + 1$.

(a) Use polynomial long division to determine the message that should be transmitted.

(b) Suppose the leftmost bit of the message is inverted due to noise on the transmission link. What is the result of the receiver's CRC calculation? How does the receiver know that an error has occurred?

14)

Suppose we want to transmit the message 1011 0010 0100 1011 and protect it from errors using the CRC8 polynomial $x^8 + x^2 + x^1 + 1$.

(a) Use polynomial long division to determine the message that should be transmitted.

(b) Suppose the leftmost bit of the message is inverted due to noise on the transmission link. What is the result of the receiver's CRC calculation? How does the receiver know that an error has occurred?

15)

Suppose you are designing a sliding window protocol for a 1-Mbps point-to-point link to the moon, which has a one-way latency of 1.25 seconds. Assuming that each frame carries 1 KB of data, what is the minimum number of bits you need for the sequence number?

16)

Suppose you are designing a sliding window protocol for a 1-Mbps point-to-point link to the stationary satellite revolving around the Earth at an altitude of 3×10^4 km. Assuming that each frame carries 1 KB of data, what is the minimum number of bits you need for the sequence number in the following cases? Assume the speed of light is 3×10^8 m/s.

(a) RWS=1

(b) RWS=SWS

17)

Draw a timeline diagram for the sliding window algorithm with SWS = RWS = 3 frames, for the following two situations. Use a timeout interval of about $2 \times$ RTT.

(a) Frame 4 is lost.

(b) Frames 4 to 6 are lost.

18)

Draw a timeline diagram for the sliding window algorithm with SWS = RWS = 4 frames in the following two situations. Assume the receiver sends a duplicate acknowledgment if it does not receive the expected frame. For example, it sends DUPACK[2] when it expects to see Frame[2] but receives Frame[3] instead. Also, the receiver sends a cumulative acknowledgment after it receives all the outstanding frames. For example, it sends ACK[5] when it receives the lost frame Frame[2] after it already received Frame[3], Frame[4], and Frame[5]. Use a timeout interval of about $2 \times$ RTT.

(a) Frame 2 is lost. Retransmission takes place upon timeout (as usual).

(b) Frame 2 is lost. Retransmission takes place either upon receipt of the first DUPACK or upon timeout. Does this scheme reduce the transaction time? (Note that some end-to-end protocols, such as variants of TCP, use similar schemes for fast retransmission.)

19)

Suppose that we attempt to run the sliding window algorithm with SWS = RWS = 3 and with MaxSeqNum = 5. The Nth packet DATA[N] thus actually contains $N \bmod 5$ in its sequence number field. Give an example in which the algorithm becomes confused; that is, a scenario in which the receiver expects DATA[5] and accepts DATA[0]—which has the same transmitted sequence number—in its stead. No packets may arrive out of order. Note that this implies MaxSeqNum ≥ 6 is necessary as well as sufficient.

20)

Consider the sliding window algorithm with SWS = RWS = 3, with no out-of-order arrivals and with infinite-precision sequence numbers.

(a) Show that if DATA[6] is in the receive window, then DATA[0] (or in general any older data) cannot arrive at the receiver (and hence that MaxSeqNum = 6 would have sufficed).

(b) Show that if ACK[6] may be sent (or, more literally, that DATA[5] is in the sending window), then ACK[2] (or earlier) cannot be received.

These amount to a proof of the formula given in Section 2.5.2, particularized to the case SWS = 3. Note that part (b) implies that the scenario of the previous problem cannot be reversed to involve a failure to distinguish ACK[0] and ACK[5].

21)

Suppose that we run the sliding window algorithm with SWS = 5 and RWS = 3, and no out-of-order arrivals.

(a) Find the smallest value for MaxSeqNum. You may assume that it suffices to find the smallest MaxSeqNum such that if DATA[MaxSeqNum] is in the receive window, then DATA[0] can no longer arrive.

(b) Give an example showing that MaxSeqNum − 1 is not sufficient.

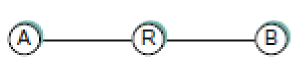
(c) State a general rule for the minimum MaxSeqNum in terms of SWS and RWS.

22)

Suppose A is connected to B via an intermediate router R, as shown in Figure 2.37. The A–R and R–B links each accept and transmit only one packet per second in each direction (so two packets take 2 seconds), and the two directions transmit independently. Assume A sends to B using the sliding window protocol with SWS = 4.

(a) For Time = 0, 1, 2, 3, 4, 5, state what packets arrive at and leave each node, or label them on a timeline.

(b) What happens if the links have a propagation delay of 1.0 second, but accept immediately as many packets as are offered (i.e., latency = 1 second but bandwidth is infinite)?



23)

Suppose A is connected to B via an intermediate router R, as in the previous problem. The A–R link is instantaneous, but the R–B link transmits only one packet each second, one at a time (so two packets take 2 seconds). Assume A sends to B using the sliding window protocol with SWS = 4. For Time $= 0, 1, 2, 3, 4$, state what packets arrive at and are sent from A and B. How large does the queue at R grow?

24)

Consider the situation in the previous exercise, except this time assume that the router has a queue size of 1; that is, it can hold one packet in addition to the one it is sending (in each direction). Let A's timeout be 5 seconds, and let SWS again be 4. Show what happens at each second from Time $= 0$ until all four packets from the first window-full are successfully delivered.

Note: Problems 1 through 24 were taken from the reference Text by Peterson and Davie (Posted)

25)

Frames of 1000 bits are sent over a 1-Mbps channel using a geostationary satellite whose propagation time from the earth is 270 msec. Acknowledgements are always piggybacked onto data frames. The headers are very short. Three-bit sequence numbers are used. What is the maximum achievable channel utilization for

(a) Stop-and-wait?

b) Go-back-N

c) Selective Reject

26)

The distance from earth to a distant planet is approximately 9×10^{10} m. What is the channel utilization if a stop-and-wait protocol is used for frame transmission on a 64 Mbps point-to-point link? Assume that the frame size is 32 KB and the speed of light is 3×10^{8} m/s.

27)

A channel has a bit rate of 4 kbps and a propagation delay of 20 msec. For what range of frame sizes does stop-and-wait give an efficiency of at least 50%?

28)

Ten signals, each requiring 4000 Hz, are multiplexed onto a single channel using FDM. What is the minimum bandwidth required for the multiplexed channel? Assume that the guard bands are 400 Hz wide.

29)

Consider an error-free 64-kbps satellite channel used to send 512-byte data frames in one direction, with very short acknowledgements coming back the other way. What is the maximum throughput for window sizes of 1, 7, 15, and 127? The earth-satellite propagation time is 270 msec.

30)

A 100-km-long cable runs at the T1 data rate. The propagation speed in the cable is 2/3 the speed of light in vacuum. How many bits fit in the cable?

Note: a T1 line has a rate of 1.5Mbps

31)

Consider the use of 1000-bit frames on a 1-Mbps satellite channel with a 270-ms delay. What is the maximum link utilization for

a. Stop-and-wait flow control?
b. Continuous flow control with a window size of 7?
c. Continuous flow control with a window size of 127?
d. Continuous flow control with a window size of 255?

32)

Two neighboring nodes (A and B) use a sliding-window protocol with a 3-bit sequence number. As the ARQ mechanism, go-back-N is used with a window size of 4. Assuming A is transmitting and B is receiving, show the window positions for the following succession of events:

a. Before A sends any frames
b. After A sends frames 0, 1, 2 and receives acknowledgment from B for 0 and 1
c. After A sends frames 3, 4, and 5 and B acknowledges 4 and the ACK is received by A

33)

Find the number of the following devices that could be accommodated by a T1-type TDM line if 1% of the T1 line capacity is reserved for synchronization purposes.

a. 110-bps teleprinter terminals
b. 300-bps computer terminals
c. 1200-bps computer terminals
d. 9600-bps computer output ports
e. 64-kbps PCM voice-frequency lines

How would these numbers change if each of the sources were transmitting an average of 10% of the time and a statistical multiplexer was used?

34)

Ten 9600-bps lines are to be multiplexed using TDM. Ignoring overhead bits in the TDM frame, what is the total capacity required for synchronous TDM? Assuming that we wish to limit average TDM link utilization to 0.8, and assuming that each TDM link is busy 50% of the time, what is the capacity required for statistical TDM?

35)

Say you want to transfer 1Kbyte (i.e. 8000 bits) frames over a 0.5 Mbps channel with a one-way delay of 32 ms, using the sliding window algorithm, with SWS = RWS.

a) Compute the value of SWS that will ensure high channel utilization.

b) Suppose the sender will <u>time-out</u> at 1.5*RTT, and an ACK frame is sent <u>for each correctly received, in-order frame</u> (Ignore the transmission time of the ACK frame but do not ignore the propagation delay). **Sketch a timeline diagram for the following case**, labeling each data frame and ACK as F_1, F_2, … and ACK_1, ACK_2, … etc. (assume that there are enough bits in the sequence field in the header of the frame such that there is no need to wrap-around sequence numbers). Clearly label the time axis assuming the first frame F_1 is transmitted at t = 0. <u>No ACKs</u> are sent for <u>out-of-order frames</u>.

<u>**CASE**</u>: F_2 and F_4 are lost (remember, frame lost means it did not reach the receiver)