CS433/533: Computer Networks

Midterm 1

2/25/2003

7:00-9:00pm

YOUR NAME ___________________________________

TODAY'S DATE ________________________________

Problem 1:

Problem 2:

Problem 3:

Problem 4:

Total:

1. [35 points] Provide ***short*** answers to the following questions.

a. [4 points] **Network architecture**
Some people refer to the Internet architecture as the hour-glass architecture. Briefly state why it has such a name.

b. [8 points] **Caching**
b.1) [4 points] Why does a Freenet node save a copy of a document when it relays the document to the receiver?

b.2) [4 points] Briefly compare how HTTP, DNS, and Freenet maintain cache freshness.

c. [4 points] **Peer-to-peer networks**
Some people say that Napster is a client-server application while some others say that it is a peer-to-peer application. What is your opinion?

d. [4 points] **Parallel stop-and-wait protocol**
A parallel stop-and-wait protocol runs several instances of the stop-and-wait protocol concurrently between a sender and a receiver. This approach is especially appealing to some transaction-based applications, such as telephone signaling or bank transfers, where the sender sends many small, independent requests to the receiver. Briefly describe why a parallel stop-and-wait protocol can be better than a sliding window protocol such as TCP for these applications.

e. [15 points] **TCP connection management**
e.1) [3 points] Suppose a client C repeatedly connects via TCP to the same port on a server S. How does the server determine which connection an arrived packet belongs to?

e.2) [4 points] Assume that it is C who initiates the active close. How many TCP connections can C make before it ties up all its available ports in TIME_WAIT state? Assume client ephemeral ports are in the range 1024-5119, and that TIME_WAIT lasts 60 seconds.

e.3) [4 points] For HTTP, it is typically the server who makes the active close. Is there any disadvantage of server closing first?

e.4) [4 points] TCP starts a timer when there are outstanding packets starting from its sending base. However, connection management packets such as SYN can also be lost. How does TCP treat the timeout of connection management packets and the normal data packets uniformly?

2. [30 points] **Sliding window protocols and reliable transport to the moon**
Suppose you are hired by NASA to design a sliding window protocol for a 1-Mbps (10^6 bit-per-second) point-to-point link to the moon, which has a one-way propagation delay of 1.25 seconds. Assume that each packet carries 1 KB (10^3 bytes). Also assume that sequence numbers are given to packets instead of bytes.

a) [4 points] What is the window size (in packets) to fully utilize the link?

b) [4 points] How many bits do you need for a sequence number if you use go-back-n? How about selective repeat?

c) [4 points] Assume that the drop probability of each data packet is p. Suppose you are transmitting the packet at the head of your sliding window. What is the expression of p_k, the probability that you need to transmit the packet exactly k times, where $k \geq 1$?

d) [4 points] Please derive the average number of times a packet is transmitted for each successful transmission by Selective Repeat. Hints: $(1 + p + p^2 + \ldots) = 1/(1-p)$; $(1+ 2p + 3p^2 + 4p^3+\ldots) = 1/(1-p)^2$. The expression is 1/(1-p).

e) [3 points] For GBN with window size *w*, packet drop probability *p*, consider the packet at the head of the window at the sender. Suppose you send this packet *k* times. When the packet is finally received at the receiver, how many packets have you sent in total from the time you first sent this packet?

f) [3 points] Derive the expression for the average number of packets is transmitted for each successful transmission by go-back-n. Hint: (1-p+pw)/(1-p)

g) [4 points] Suppose *p* is 0.0004 (0.04%) for the link from earth to the moon. Compare the efficiency of go-back-n and selective repeat. Which one do you recommend to NASA, go-back-n or selective repeat?

h) [4 points] To avoid writing a new program from scratch, you decide to use TCP for your reliable data transfer. The loss rate *p* is the same as in g). Does this suggestion work well? If yes, support your claim. If no, please state why?

3. [18 points] **TCP with different parameters**
TCP increases its window size by *1/cwnd* when an ACK is received, and cuts its window to half when it detects a packet loss. We can extend TCP congestion control to use different parameters.

a) [5 points] Suppose your new implementation does not increase window size by *1/cwnd*, but *a/cwnd*, for each ACK; suppose also that your implementation cuts the window size to *b* times of the current window (for example TCP cuts to half, i.e., b = ½), where *a* and *b* are parameters. What is the long-term average throughput of your new congestion control scheme? Assume no slowstart, delayed ACK or timeout. Hint: see the derivation for TCP.

b) [5 points] Suppose a TCP flow and your flow using parameters *a* and *b* are competing on the same path from a source to a destination. Thus the two flows will experience the same loss rate. You can choose a large value of *a* to receive more bandwidth than the competing TCP flow. However, consciousness limits you to receive only twice the bandwidth received by the TCP flow. Suppose you have to set b = ½, the same as TCP does. What value should you set *a* to be?

c) [4 points] Another approach to giving a flow more bandwidth is to change a router so that the router gives that flow higher priority. How do you argue for the approach of using different *a* and *b* parameters to achieve different throughput?

d) [4 points] The approach of using different *a* and *b* parameters will not always give a higher throughput under all scenarios. Give a scenario where more aggressive parameters will not help much.

4. [17 points] **TCP and HTTP**
Consider a user who accesses the Internet using a connection with a link speed of *4Mbps*. *For ease of calculation, you can assume 1K = 1000, and 1M = 1,000,000.*

a) [5 points] Assume that this is the first time that the user visits the home page of Yale. The user types in the URL http://www.yale.edu/index.html. Please list three types of application servers that this transaction may involve.

b) [12 points] Consider the latency of the TCP connection to download the web page, assuming the web server uses HTTP/1.1 persistent connection. Assume that the round-trip time between the host and the web server is 12ms; the size of the web page is 4K bytes; the web page contains 5 images of 1K bytes each
at the end of the web page; MSS size of TCP is 1K bytes; TCP/IP header is 40 bytes. For simplicity, you can ignore the transmission time of all short packets (e.g. ACK, HTTP request). Draw the space-time diagram for the TCP transaction, assuming no packet loss and no delayed ACK. Please label the delays on the diagram.