EECS 461 Midterm Exam
February 1998

Your Name: Sample Solution

This is a **closed book** examination; you are permitted one 8"x11" page of notes. You have 80 minutes to answer as many questions as possible. The number in parentheses at the beginning of each question indicates the number of points given to the question; there are 60 points in all. We expect the exam to take roughly one point per minute; we included the extra time to help ensure that everyone has enough time to complete the exam.

Write all of your answers directly on this paper. Make your answers as concise as possible (you needn't cover every available nano-acre with writing). If there is something in a question that you believe is open to interpretation, then please go ahead and interpret **but** state your assumptions in your answer.

Problem 1: (10 points)

For each of the following statements, indicate in one sentence whether the statement is true or false, and **why**.

(a) The stop-and-wait protocol is always less efficient than the sliding window protocol if the window size is greater than one. Hint: does it matter if the link bandwidth times latency is smaller than the packet size?

False: on a simplex channel with a very low latency (consider an ethernet for example) stop and wait can achieve full link bandwidth.

Note that for a duplex channel sliding window *will* always be better, because it can be transmitting a second packet during the time that the ack for the first packet is being transmitted, so this answer was acceptable if it was properly explained.

(b) The TCP header has separate fields for the acknowledgment number and the data sequence number.

True, because TCP is a full duplex channel, and it can piggyback acks for one direction on data for the other direction.

c) One can completely solve the retransmission ambiguity problem by simply ignoring round-trip time estimates for retransmitted packets.

False: Correctly estimating round trip times also requires increasing the estimate when timeouts happen, or else too short a timeout will never be fixed.

(d) The TCP packet header includes an "advertised window" field to reduce congestion at routers.

False: the advertised window is for the receiver to throttle back the sender if it is sending faster than the receiver can handle. It has nothing to do with router congestion.

(e) It is impossible to guarantee that a remote procedure call will be executed exactly once, even with the assumption that machines that crash eventually resume.

False: by logging to disk before the RPC is sent and after it is received it is possible to make sure it is executed exactly once in all cases.

Problem 2: Project implementation (10 points)

(a) A common mistake in implementing the alternating bit protocol is to have the receiver only acknowledge packets whose sequence number is expected. Explain how this strategy will cause the protocol to fail.

If an ack is dropped, the sender will keep retrying until it gets an ack, but if the receiver doesn't resend an ack when it gets a duplicate, the sender will never realize the data got through.

(b) Some students implemented the "fast retransmit" TCP optimization for the first project assignment. With fast retransmit, TCP retransmits a missing packet as soon as it receives a "duplicate" acknowledgment, signifying that a packet was received out of order, indicating that there is a missing packet that might have been dropped. Why does this optimization usually work well for TCP in the Internet, and why did it not work very well for the simulated network in the project? Hint: the simulated network reorders virtually every packet.

On the internet, packets usually arrive in order unless a route has changed (infrequent). So if a packet arrives out of order, it usually means the prior packet has been dropped, so resending is probably a good idea.

In the project, packets always arrive out of order, often substantially so, so we can't conclude much of anything from out of order arrivals.

Problem 3: (10 points)

In the original HTTP (world wide web) protocol (version 1.0), a separate TCP connection was established between the client and server for every separate data object (e.g., text frame, picture, or advertisement) on a page. The HTTP protocol has recently been revised (version 1.1) to allow for "persistent connections", to keep the TCP connection between the client and server open until all data objects on a page have been transferred.

(a) List all the ways that persistent connections reduce the latency to retrieve the contents of a multi-object page from a server.

(1) No three-way handshake before requesting each object after the first

(2) Slow start only needed for first object; later objects can be transferred with a bigger window.

(3) Round trip time estimation will be more accurate, but that is only important if there are drops.

(b) Assuming a page has three objects (including the text frame) and each object takes 1 packet to request and 4 packets to reply, what is the minimal number of round trips that are required (for example, assuming no drops) before the entire page can be displayed using the original and revised protocols?

Problem 4: (10 points)

Several years after the ATM standard was defined, researchers discovered an unfortunate interaction between two of ATM's features. First, to transfer a TCP packet across an ATM network, the packet is fragmented into 53 byte cells and reassembled at the destination. Both the fragmentation and reassembly are done in hardware, along with a CRC to ensure that all the cells in the packet were delivered. The packet is only delivered to the TCP code on the destination machine if the CRC check succeeds; ATM relies on TCP to retransmit the packet if the CRC fails. Second, ATM permits switches to drop cells if the switch becomes congested; cells are dropped in round-robin fashion so that a cell is dropped from each virtual circuit before the second cell is dropped from any circuit.

(a) Why might this combination of features have lower throughput than a normal TCP/IP network where routers drop entire packets under congestion?

If we drop even one frame from a packet the entire packet will have to be resent. Round-robin is exactly the worst way to drop frames, because it will cause the maximum number of packet drops for any number of frame drops, since it will effect the maximum number of packets (in the minimum way possible, but that doesn't matter).

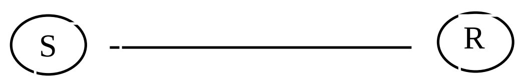
(b) Many ATM switches now implement "tail drop", so that if they drop one cell in a packet, they drop all successive cells from that packet before dropping any cells from other packets. Why does this improve performance if the network is congested?

Tail drop makes sense because it takes advantage of the understanding that once the first frame from a packet has been dropped the rest are useless. There is no cost to dropping them, and there is some benefit because it will reduce congestion an possibly spare other packets.

Note: the two parts to problem 4 were graded as a whole. The answer to part b was worth most of the points.

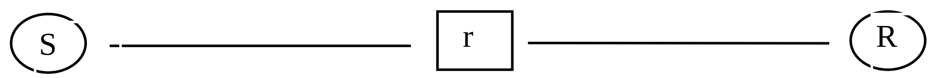
Problem 5: (10 points)

(a) Suppose we want to measure the bit-rate and delay of each link in a store and forward network path between two end points. First consider just one hop. We devise a scheme where on receipt of a special "probe" packet, the destination immediately transmits a response to the source. The source then uses the elapsed round-trip time to infer properties of the underlying network. If the source S sends a 1000-bit packet to the destination R, the response is received after 30 milliseconds (1 second = 1000 milliseconds). If S sends a 2000-bit packet to the destination R, the response is recieved after 40 milliseconds. Find the bit rate (bandwidth) and propagation delay (latency) of the link. Assume the rate and delay are the same in both directions and that the response packet is small enough that you can ignore its transmission delay (but not its propagation delay).



bandwidth = (2000 – 1000) bits / (40 – 30) ms = 100 Kb/s
round-trip = 30 ms – 1000 bits / 100 Kb/s = 20 ms
latency = 10 ms

(b) Now, consider a two-hop configuration. Again, we assume that this is a store and forward network. We note that routers will reply back to the source when a packet's hop count reaches 0 (analogous to the receiver's response message from above). In this case, if S sends a 1000-bit packet to R with a hop-count of 1, the control message from the intermediate router is received after 30 milliseconds, while if S sends a 2000-bit packet with a hop-count of 1, the control message is received after 40 milliseconds (as above). Likewise, if S sends a 1000-bit packet to R with a hop count of 2, the control message is received after 240 milliseconds, while if S sends a 2000-bit packet with a hop count of 2, the control message is received after 290 milliseconds. Compute the bit rate and propagation delay for each of the two links, and briefly explain how you could extend this algorithm to an arbitrary number of hops.



Link 1 is the same as part (a).

Link 2:

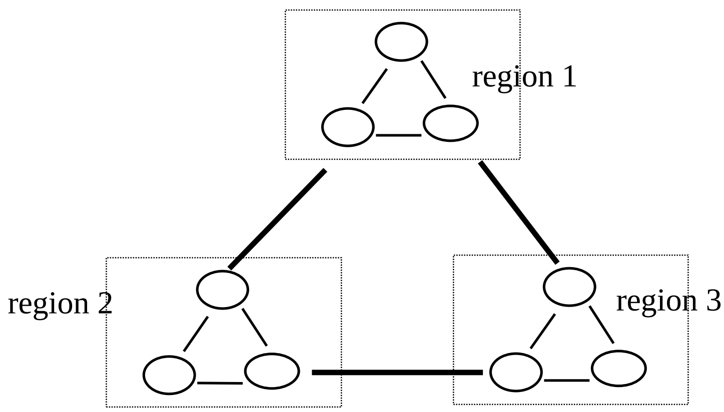
Bandwidth = 1000 bits / [(290-40) – (240-30)] = 25 KB/s
Round-trip = (240-30) – 1000 bits/25 Kb/s = 170 ms
Latency = 85 ms

The approach can be extended to any number of hops by induction. Solve for the kth link by determining time to send 1Kb and 2Kb probes to k-1th link (inductively), then subtract off those (the numbers for k-1) from the times for hop count = k.

Problem 6: (10 points)

For scalability, Internet routing is hierarchical, in that groups of routers and hosts are grouped into "autonomous regions", and a separate shortest path routing protocol is run between regions and inside each region. Inside each region, routes are chosen using the standard link state protocol. Between regions, routes are chosen to traverse the shortest number of **regions**; when a packet reaches a new region, it is routed along the shortest path within that region to the next autonomous region in its path. Although more scalable, hierarchical routing does not always yield the absolute shortest path, such as would always be found by running link state across the entire topology.

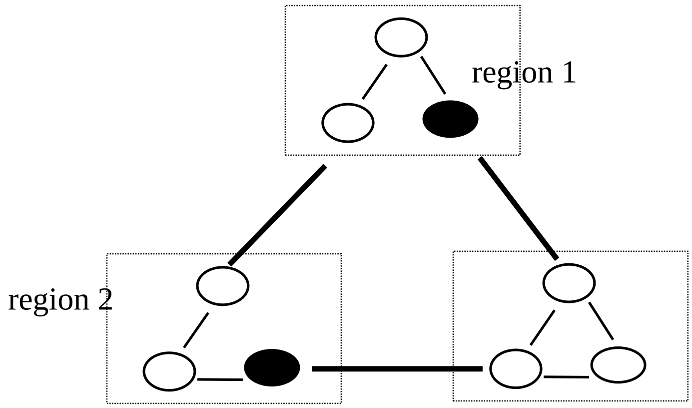
(a) Consider the following topology, where circles represent routers, light lines represent intra-region links, and dark lines represent links between regions. Assuming each link has equal cost, for this topology, are all hierarchical shortest routes the same as the absolute shortest routes? If so explain why; if not, give a counterexample.



In this topology the hierarchical routes are the shortest routes.

The top-level route is always a direct hop from the source domain to the destination domain. Similarly all intradomain routes are of length 1. So the greatest cost for any hierarchical route is 3. The only case in which a non-hierarchical route might be better would be if it found a shortcut through another domain, but this cannot happen because the least cost for any route which goes through a third domain is also 3 (one hop in, one inside, one out).

(b) Draw a topology for which your answer to (a) would be different. If you said that hierarchical routing for (a) always yielded the true shortest path, then draw a topology for which hierarchical routing sometimes produces a non-shortest route. If you said that (a) sometimes used a non-shortest route, then draw a topology for which hierarchical routing always produces the shortest route.



Consider removing the two links as shown, and routing between the shaded nodes. The hierarchical route will have length 5 (2 inside the source region, 1 to the destination region and 2 inside the destination) but there is a non-hierarchical route of length 3 through the other domain.

Bonus Question: (2 points)

A recent study showed that the only people who have an accurate view of their own abilities are clinically depressed people (I'm not making this up -- well, at least not entirely ;-). For example, 80% of Americans believe that their automobile driving skills are above average; clearly, we can't all be right. Estimate your score on this exam (not including the bonus points!); if you are correct, we'll need to cheer you up, so we'll give you a couple bonus points.

60