

Question 1 (6 points) Compare circuit-switching and packet-switching networks based on the following criteria:

(a) Reserving network resources ahead of data being sent: (2pts)

In circuit-switching networks, a physical circuit is established between the source and the destination. As part of setting up the circuit, resources are allocated for the connection. After the circuit is established, data can flow between the two end-points.

In packet-switching networks, there is no circuit establishment and therefore no resources are reserved. As such, data can flow without having to wait for a circuit to be set up.

(b) Utilizing network resources efficiently: (2pts)

In circuit switching networks, network resources may not be fully utilized, e.g., during “silent” times during calls. Since resources are reserved on a per-call basis, they cannot be utilized by other calls, even when they are idle.

In packets switching, resources are used on-demand, therefore, available resources will be used to fulfill demand.

(c) Guaranteeing uniform network quality of service during the connection: (2pts)

Since they reserve resources on a per-call basis, circuit-switching networks can guarantee uniform network quality of service for a connection, while packet-switching networks cannot.

Question 2 (6 points)

(a) What are “loss-intolerant” applications? Give an example of a loss-intolerant Internet application. (2pts)

Applications that don't tolerate loss of data. Example; Email, File transfer, Web.

(b) What are “delay intolerant” applications? Give an example of a delay-intolerant Internet application. (2pts)

Applications that don't tolerate delay in data delivery. Example; Internet telephony, video conferencing, ...etc.

(c) Based on your comparative analysis in Question 1, does the Internet adequately support loss-intolerant applications? And delay-intolerant applications? Explain. (2pts)

The Internet was designed to support loss-intolerant applications but was not designed to support delay-intolerant applications properly. This is because, as discussed in Question 1, the Internet, as a packet switching network, cannot provide delay bound guarantees.

Question 3 (6 points) Consider an Internet application running on two Internet hosts, Host1 and Host2, which are communicating using a path through router R. Assume that the average service time at R is 10 ms, packets are 2,000 bits long, and the propagation delay between Host1 and Host2 is 150ms.

(a) Based on the latency expression we covered in class, how is service time defined? Write the mathematical expression for it explaining its terms. (2 points)

The service delay is the sum of the delays for processing and queueing.

Processing: checking bit errors, determining output link

Queueing: waiting for output link availability

Total delay = processing + queue + transmission + propagation

Service time = processing + queue

(b) If the application running on Host1 and Host2 cannot tolerate latencies over 200ms, what is the minimum capacity of the link connecting the two hosts? Show your work. (4 points)

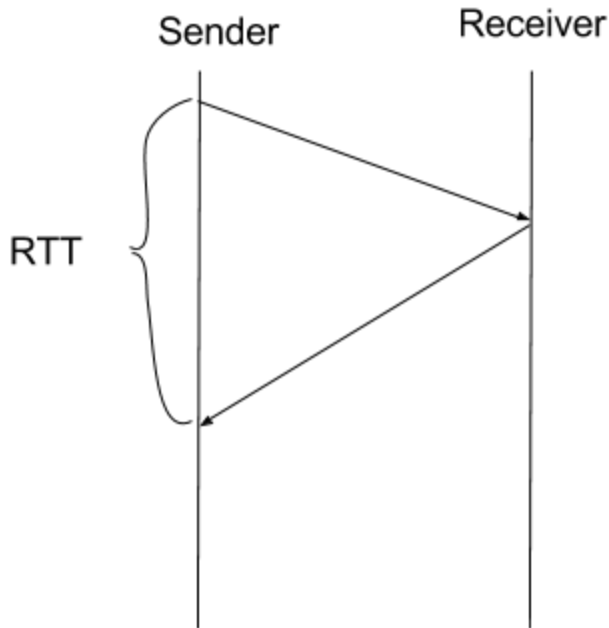
10ms service + 150ms propagation = 160ms -> 40ms for transmission

Transmission of 2000 bits in 40ms -> $2000 / .04 = 50 \text{ kb/s}$

Question 4 (4 points) Another way to measure end-to-end network latency is to use the round-trip time (RTT).

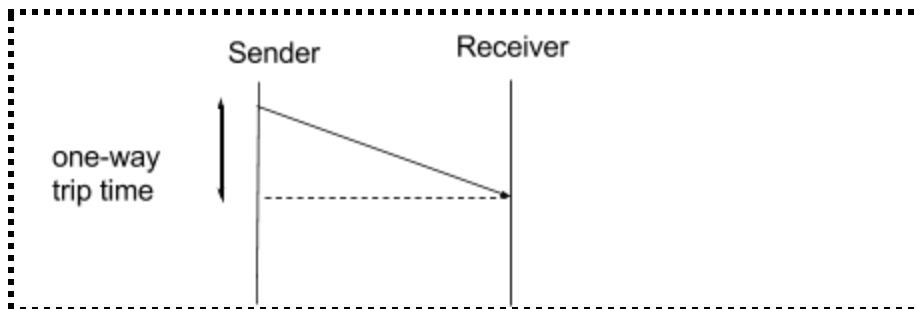
(a) What is the RTT? Use a time diagram to illustrate your explanation.

The time it takes for data to travel from sender to receiver and then back to the sender.



(b) What is the one-way trip time? Illustrate your answer with a time diagram.

It is the the time takes for data to be transmitted across a network from sender to receiver.



Question 5 (8 points) List one advantage and one disadvantage of:

(a) Peer-to-peer model for networked applications (when compared to the client-server model). Explain. (2 points)

Advantage: self-scalability”, new peers bring new service capacity, as well as new service demands

Disadvantage: Management and security, peers are unreliable and increase security risks.

(b) Web caching. Explain. (2 points)

Advantage: reduce response time for client request, save network- and server provider resources.

Disadvantage: on cache miss longer response delay and potential for out-of-date content.

(c) Layering. Explain. (2 points)

Advantage: It helps designing, maintaining, extending complex systems. Provide the ability to change the implementation of a service/function without affecting other layers of the model.

Disadvantage: Additional overhead; potential for duplication of functions.

(d) DNS (compared to a centralized name service). Explain. (2 points)

Advantage: Scalability and fault tolerance

Disadvantage: Complexity

Question 6 (14 points) TCP

(a) Is TCP a full-duplex protocol? Explain.

Yes since, in the same connection, data can flow in both directions.

(b) When Host1 initiates a TCP connection to Host2, it sends a SYN segment to Host2. As part of the SYN segment, it includes a sequence number SEQX. What is the purpose of sequence number SEQX?

SEQX will be the first sequence number Host 1 will use when sending data to Host2.

(c) Suppose that the Host1 application that initiated the TCP connection has data to send to Host2. Can Host1's SYN segment carry the data? Explain.

No, Host1 needs to wait for the 3-way handshake to complete before sending data to Host2.

(d) When Host2 replies to the SYN segment from Host1, it uses a SYN-ACK segment and also includes a sequence number SEQY . What is the purpose of SEQY ? Is SEQX = SEQY ? Explain.

No, SEQX is different from SEQY. SEQY will be the first sequence number Host2 will use when sending data to Host1.

(e) The original TCP protocol only retransmitted upon a timeout event. More modern variants of TCP use the Fast Retransmit mechanism. Explain how Fast Retransmit works and how it can improve TCP performance.

Fast retransmit is when TCP considers a packet lost after receiving a number (typically 3) of duplicate ACKs. This way TCP doesn't have to wait for the timeout event to trigger a retransmission. By retransmitting sooner, it can try to recover more quickly and avoid the timeout.

(f) TCP can "piggyback" acknowledgments on data segments. What field(s) in the TCP segment header are used to do that? Explain.

TCP's header acknowledgment field and the ACK flag.

(g) What is the main advantage of "piggybacked" acknowledgments?

It improves network resource usage by reducing the number of control packets that have to be sent.

Question 7 (12 points) The reliable data transfer, or rdt, protocols we covered in class provide reliable in-order data delivery with duplicate detection.

(a) What mechanism(s) used by rdt protocols cause duplicates to be generated? Explain. (2 points)

Duplicates are generated because of retransmissions. Consider a scenario in which a segment was transmitted but the segment got delayed inside the network. It eventually arrives at the receiver but in the meantime, the sender times out and retransmits the segment. The receiver would then receive a duplicate and, in order to guarantee duplicate-free delivery to the application, it needs to detect and get rid of duplicates.

(b) What mechanism(s) used by rdt protocols detect duplicates? Explain. (2 points)

Sequence numbers.

(c) rdt protocols can recover from two types of network losses. What mechanism(s) are used to recover from each type of loss? Explain. (3 points)

Checksums for data corruption, and timers for packets dropped in transit.

(d) What is the retransmission timeout, also known as RTO? (2 points)

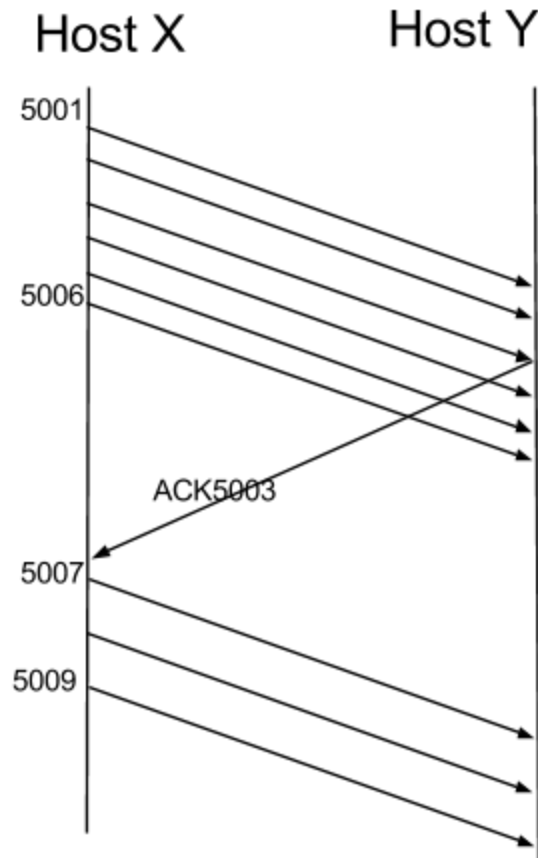
It's the amount of time the sender will wait for an acknowledgement for a transmitted segment before considering that the segment was lost and starting the retransmission of that segment.

(e) In class we discussed why retransmission timers are critical to the performance of reliable data transfer. Why are they set as a function of the RTT? (3 points)

Because the retransmission timer needs to reflect the network's current conditions. The sender shouldn't wait needlessly to retransmit the segment if it was lost, but it also shouldn't add unnecessary duplicate traffic if the segment is simply taking longer to be delivered by the network.

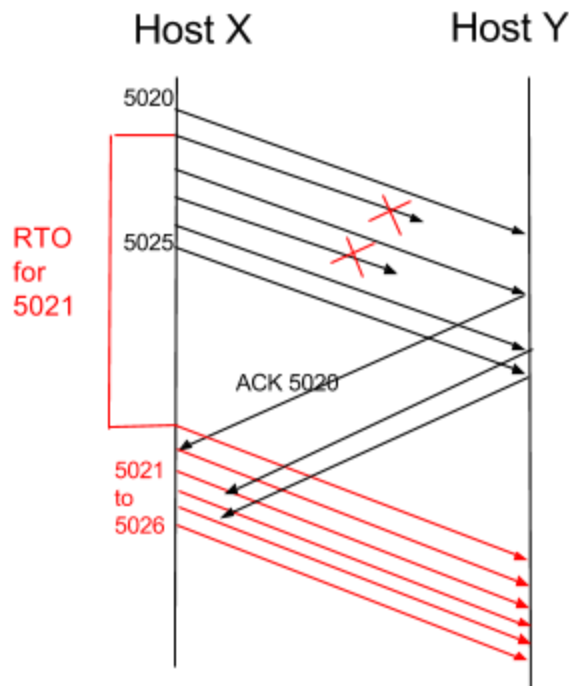
Question 8 (14 points) Two hosts X and Y are communicating over the Internet using TCP. Suppose that the TCP connection has already been opened.

(a) Assume host X's window size is 6 and segments 5001 through 5006 have already been sent. Host X then receives an ACK with ACK sequence number set to 5003. Segments 5007 through 5011 are waiting to be transmitted by host X. Draw the time diagram showing this scenario including which segments, if any, X can transmit after getting the ACK with ACK sequence number set to 5003, and explain why. (4 points)



When the sender receives the ACK for segment 5003, it means that the receiver received segments up to 5003 (5001-5003). This is because TCP uses cumulative acknowledgment. Because our window size is 6 and we have only 3 unacked segments, the sender will send segments 5007 to 5009.

(b) Some time later, X sends segments 5020 to 5025 (having received acknowledgments for all previously sent segments). Host Y receives segments 5020, 5022, 5024, and 5025. Assuming that segments that were not received were lost, draw the time diagram for this scenario, including the acknowledgments that host Y generates assuming it uses a Go-Back-N based variant of TCP. What does host Y do with segments received out of order and which are these segments? Explain. (4 points)



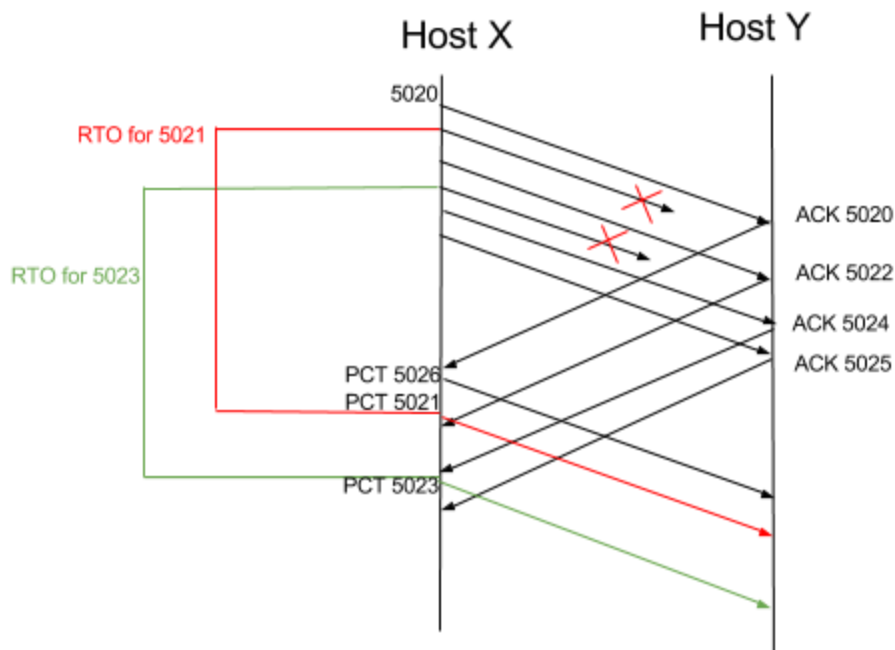
Because GBN uses cumulative acknowledgments, Host Y will send 3 ACKs for segment 5020. Only after the first ACK host X will send segment 5026, for the next 3 ACKs it won't send any new segments. Once a timeout event for segment 5021 is triggered, Host X will send all segments that have been previously sent but not yet acknowledged (5021 - 5026).

Host Y will discard any segments that were received out of order (5022, 5024, 5025, 5026).

(c) When X gets Y's acknowledgments, what will it do? Explain. (3 points)

Once host X receives the first ACK for packet 5020, it will compare the sequence number for the ACK with the next expected sequence number (5021). If they match, host X will restart the timer. Because, the next three ACKs (5020) don't match the next expected sequence number (5021), host X will not restart the timer which will lead to a timeout event and Host X will retransmit segments 5021-5026.

(d) Suppose the same situation described in (d), but both X and Y use a Selective-Repeat variant of TCP. What acknowledgments will Y send to X? What does Y do with out-of-order segments? Explain. And, what will X do when it gets Y's acknowledgments? Explain. (3 points)



Host Y will send ACK for segments 5020, 5022, 5024, and 5025. Host Y will buffer any out of order segments.

Explanation: Because in SR, the receiver individually acknowledges correctly received segments, Host Y will acknowledge the receipt of segments 5020, 5022, 5024, and 5025 even though they are received out of order. Host Y will buffer segments 5022, 5024, 5025, and 5026 until it receives segments 5021 and 5023.

Question 9 (20 points) You are studying for the CE 150/L midterm using one of the computers at the UCSC Science Library. During one of your breaks, you want to check

out the Soccer4All.org Web site for the latest videos of the Soccer World Cup 2018 Qualifiers.

(a) What would be the steps needed before your computer at ucsc.edu can issue a request for Soccer4All.org? Assume this is the first time anyone at ucsc.edu requests contents from that Web site and name resolution is done iteratively. (4 points)

Computer sends DNS request to local DNS name server at UCSC.

Local DNS name server sends DNS request to root name server.

Root name server responds with name server for .org domain.

Local DNS server sends DNS request to .org domain name server.

.org domain name server replies with the address for authoritative name server for soccer4all.org.

Local DNS server requests address from Soccer4All.org authoritative name server.

Soccer4All.org authoritative name server replies with Soccer4All.org Web server's IP address.

Local DNS name server forwards the IP address to client.

(b) Immediately after you downloaded videos, another friend, Gabriela, who is also at UCSC, wants to watch the videos as well. What are the steps needed before Gabriela's request to Soccer4All.org is issued? Explain. (4 points)

Computer will request the address to UCSC's local DNS name server which will reply with the proper IP address without the need to query other servers because it should have the answer in its cache.

(c) You finally manage to access the Soccer World Cup 2018 Qualifiers page from the Soccer4All.org site. The page has 10 videos embedded in it. The processing/service time within the network is 10ms each way, and the one-way propagation delay is 100ms. Assume that transmission delay is negligible. What would be the response time, i.e., the time it takes between when your browser requests the page and when the videos are delivered, if your browser uses non-persistent HTTP (with no parallel connections)? Explain. (4 points)

1 page with 10 videos means 11 objects. Non-persistent HTTP means that we will need 11 requests, each one with its own TCP connection. Each request will take $(100 + 10)ms * 2 * 2$, that is, two RTTs of 220ms each. So in total we will need 4840ms.

(d) What would be the response time if the browser uses persistent HTTP? Explain. (4 points)

In this case we will only need one connection, so $1 + 11$ RTTs, that is, one to establish the connection and one for each object requested, resulting in a total of 2640ms.

(e) It is now Gabriela's turn to download the videos. Assuming that the steps in (b) have already been executed and that you have already been able to download the videos, what is the average response time Gabriela will experience for each video in Seth's Web page if the ucsc.edu's cache hit ratio is 60%? Suppose that the delay to access an object within ucsc.edu's network is 20ms. Assume persistent HTTP is used. (4 points)

Considering the cache, the time to get any object would be $.6 * 20 + .4 (20 + 220) = 108$ ms
Then we need 220ms to start the connection, plus 108ms for each object. Total is 1408 ms.