

CS176A Example Problems

Calculate the total time required to transfer a 1.5 MB file in the following cases, assuming RTT of 80ms, a packet size of 1KB and an initial 2XRTT of “handshaking” before it is sent.

- The b/w is 10Mbps, and the data packets can be sent continuously.**
- The b/w is 10Mbps, but after we finish sending each data packet, we must wait one RTT before sending the next**
- The link allows infinitely fast transmits, but limits bandwidth such that only 20 packets can be sent per RTT zero transmit time as in (c), but during the first RTT, we can send one packet during the 2nd RTT we can send 2 packets, during the 3rd we can send 4 = 2³-1 and so on**

Solution:

We will count the transfer as completed when the last data bit arrives at its destination

a) 1.5MB = 12,582,912 bits, 2 initial RTTs (160ms) + (12,582,912)/(10,000,000) bps (transmit) + RTT/2 (propagation) = 1.458 secs (approx)

b) To the above, we add the time for 1499 RTTs for a total of 1.46 + 119.92 = 121.38 secs

c) This is 74.5 RTTs, plus the two initial RTTs, for 6.12 secs

d) Right after the handshaking is done, we send one packet. One RTT after handshaking, we send 2 packets. At n RTTs past the handshaking, we have sent $1+2+4+\dots+2^n = 2^{(n+1)} - 1$. At n =10, we have thus been able to send all 1500 pkts, the last batch arrives 0.5 RTT later. Total time is 2 + 10.5 RTTs or 1 sec.

Suppose a 128 kbps p2p link is set up between earth and a rover on mars. The distance from the earth to mars (when they are the closest together) is approximately 55Gm, and data travels over the link at the speed of light 3×10^8 m/s

- Calculate the minimum RTT for the link**
- Calculate the delay X bandwidth product of the link**
- A camera on the rover takes pictures of its surroundings and sends these to the earth. How quickly can it reach Mission Control on Earth? Assume that each image is 5Mb in size**

Solution:

a) Propagation Delay of the link is $55 \times 10^9 / (3 \times 10^8) = 184$ secs, Thus RTT = 368 secs

b) The delay X bandwidth product for the link is the RTT X bandwidth = 23.5Mb

c) After a picture is taken, it must be transmitted on the link and completely propagated, before Mission Control can interpret it. Transmit delay for 5Mb of data is 29 secs

Hence, total time = transmit delay + propagation delay = 223 secs

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

- 1 Gbps Ethernet with a single store and forward switch in the path, and a packet size of**

5000bits. Assume that each link introduces a propagation delay of 10 micro second and that the switch begins retransmitting immediately after it has finished receiving the packet.

- b) Same as a) but with 3 switches**
- c) Same as b) but assume the switch implements "cut-through" switching. It is also able to begin retransmitting the packet after the first 128 bits have been received.**

Solution:

a) For each link it takes $1 \text{ Gbps} / 5 \text{ Kb} = 5 \mu\text{s}$ to transmit the packet on the link, after which it takes an additional $10 \mu\text{s}$ for the last bit to propagate across the link. Thus for a LAN with only with only one switch that starts forwarding only after receiving the whole packet, the total transfer delay is the transmit delays + two propagate delays = $30 \mu\text{s}$.

b) For 3 switched and thus 4 links, the total delay is 4 transmit + 4 prop. Delay = $60 \mu\text{s}$

c) For "cut-through", a switch needs to only decode the 128 bits before it begins to forward. This takes 128 ns. This delay replaces the switch transmit delays in the previous answer for a total delay of one transmit delay + 3 cut through decoding delays + 4 propagation delays = $45.384 \mu\text{s}$

For the following, assume that no data compression is done. Calculate the bandwidth necessary for transmitting in real time.

- a) HDTV high-resolution video at resolution of 1920×1080 , 24 bits/pixel, 30 frames / second.**
- b) POTS (Plain Old Telephone Service) voice audio of 8-bits samples at 8 KHz**
- c) GSM mobile voice audio of 260-bit samples at 50 Hz.**
- d) HDCD high-definition audio of 24-bit samples at 88.2 KHz**

- a) $1920 * 1080 * 24 * 30 \approx 115 \text{ Gbps}$
- b) $8 * 8000 = 61 \text{ Kbps}$
- c) $260 * 50 = 13 \text{ Kbps}$
- d) $24 * 882000 \approx 2.1 \text{ Mbps}$

Consider an application that transmits data at a steady rate (for example, the sender generates an N-bit unit of data every k time units, where k is small and fixed). Also, when such an application starts, it will continue running for relatively longer period of time. Answer the following questions briefly justifying your answer:

- a) Would a packet switched network or a circuit switched network more appropriate for this application? Why?**
- b) Suppose that a packet switched network is used and the only traffic in this network comes from such applications described above. Further assume that the sum of the application data rates is less than the capacities of each and every link. Is some form of congestion control needed? Why?**

a) A circuit-switched network would be well suited to the application described, because the application involves long sessions with predictable smooth bandwidth requirements. Since the transmission rate is known and not bursty, bandwidth can be reserved for each application session circuit with no significant waste. In addition, we need not worry greatly about the overhead costs of setting up and tearing down a circuit connection,

which are amortized over the lengthy duration of a typical application session.

b) Given such generous link capacities, the network needs no congestion control mechanism. In the worst (most potentially congested) case, all the applications simultaneously transmit over one or more particular network links. However, since each link offers sufficient bandwidth to handle the sum of all of the applications' data rates, no congestion (very little queuing) will occur.

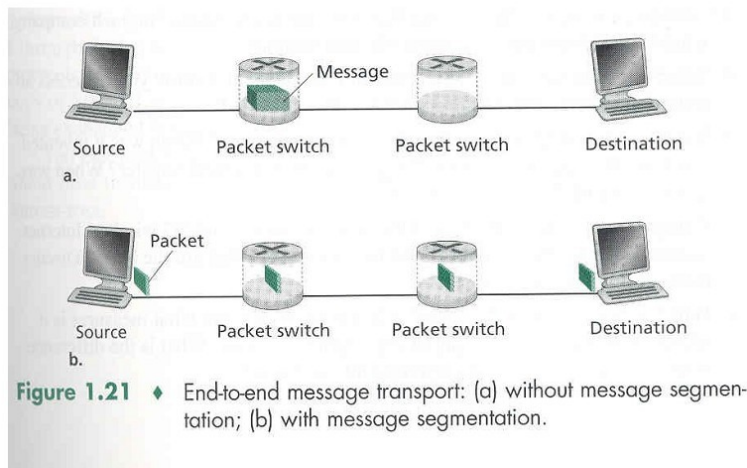
In modern packet switched networks, the source host segments long application layer messages (for example an image or a music file) into smaller packets and sends the packets into the network. The receiver then reassembles the packets back into the original message. We refer to this process as message segmentation. Fig illustrates the end to end transport of a message with and without segmentation. Consider a msg that is 7.5×10^6 bits long that is to be sent from the source to the destination as shown in the figure. Suppose that each link in the figure is 1.5 Mbps. Ignore propagation, queuing and processing delays.

a) consider sending the message from source to destination without message segmentation. How long does it take to move the message from the source host to the first packet switch? Keeping in mind that each packet switch uses a store and forward packet switching. What is the total time to move the message from the source host to the destination host?

b) Now suppose that the message is segmented into 5,000 packets, with each packet being 1500 bits long. How long does it take to move the the first packet from the source host to the first packet switch?

c) How long does it take to move the file from the source host to the destination host when message segmentation is used? Compare with part A and comment.

d) Discuss the drawbacks of message segmentation.



a) Time to send message from source host to first packet switch =

$7.5 \times 10^6 \text{ sec} = 5 \text{ sec}$. With store-and-forward switching, the total time to move

1.5×10^6 message from source host to destination host = $5 \text{ sec} \times 3 \text{ hops} = 15 \text{ sec}$

b) Time to send 1st packet from source host to first packet switch = $1.5 \times 10^3 \text{ sec} = 1 \text{ m sec}$. Time at which 2nd packet is received at the first switch = 1.5×10^6 time at which 1st packet is received at the second switch = $2 \times 1 \text{ m sec} = 2 \text{ m sec}$

c) Time at which 1st packet is received at the destination host = $1 \text{ m sec} \times 3 \text{ hops} = 3 \text{ m sec}$. After this, every 1msec one packet will be received; thus time at which last (5000th) packet is received = $3 \text{ m sec} + 4999 * 1 \text{ m sec} = 5.002 \text{ sec}$. It can be seen that delay in using message segmentation is significantly less (almost 1/3rd).

d) Drawbacks:

- i. packets have to be put in sequence at the destination.
- ii. Message segmentation results in many smaller packets. Since header size is usually the same for all packets regardless of their size, with message segmentation the total amount of header bytes is more.