Gabriel Borges

Professor Wei

CSE3300 – Assignment01

9/2/16

Assignment01

1. Problem R13

1. With circuit switching, the maximum supported users is two. The way to find this answer, is to take the total bandwidth available, and divide that by the amount of bandwidth that each user uses. So 2Mbps / 1Mbps = 2 users total.
2. There will be no queuing delay for packet switching of two or fewer users, because packet switching uses the maximum bandwidth available on the link. Because each user on the link only uses 1Mbps, two users will meet the full amount available on the link (2Mbps). Therefore, because the limit of the link is not exceeded with two or fewer people transmitting simultaneously, there will be no queuing delay. This changes when more than two people transmit, as n-number of people would be transmitting with a total bandwidth of n-Mbps. Whenever n is three or more, that will exceed the total amount of bandwidth available on the link.
3. The probability that a given user is transmitting is 0.2. This is given in the problem where it says that each user transmits only 20 percent of the time.
4. The probability that n users are transmitting simultaneously can be determined by taking the probability that any one user is transmitting (0.2) raised to the n power. Therefore, the probability that three users are transmitting simultaneously is 0.2^3, which equals 0.008. The fraction of time that the queue will start growing (i.e. there will begin to be a queue delay on the link) was determined above in part b. Whenever there are three or more users transmitting, there will begin to be a queuing delay (which increases the more users there are transmitting simultaneously). Therefore, the fraction of time during which the queue grows Is equal to the probability (0.008).

2. Problem P2

The general formula for finding the delay of a given packet across N links (assuming N-1 routers between the source and end-point) is given to be N (L / R). If we use an example where there are two packets, we know that the second packet reached the end-point at time N (L / R) + (L / R). The second packet needs to travel through one more router than the first one, as each router can only process one packet a time, and given in the question, the packets are sent back to back. I.E. for three packets, this would change to be N (L / R) + 2 \* (L / R) as the third packet will still need to travel through two routers by the time the first is done. As an aside, we keep the N (L / R) as we want to retain the generalization for N number of links. This can then be generalized for P number of packets, by multiplying the extra time (the second L/R) by P-1. Therefore the equation becomes N (L / R) + (P – 1)(L / R), or (N + (P – 1)) \* (L / R).

3. Problem P6

1. Propagation delay is given in the book to be “the distance between two routers divided by the propagation speed.” Therefore, the dprop in terms of m and s is m/s, where m is the distance and s is the speed.
2. The transmission time of a packet is given to be the length of a given packet (denoted L) divided by the transmission rate (denoted R). Therefore, the transmission time in terms of L and R is given to be dtrans = L / R.
3. An expression for the total delay of transmission and propagation delay is to add the two delays together (the time to transmit the packet and the time to propagate it). The total delay = (m/s) + (L / R).
4. If the time is at the total transmission delay time of the router, then the entire packet has been processed, meaning the last bit of the packet has just left the router (or Host A).
5. If the propagation delay is greater than the transmission delay, then at the total delay transmission time, the entire packet has been processed through the first router and the last bit of the packet has just left the router (established in question four). However, since the total amount of time to propagate the packet to the next router is greater than the total amount of time it takes to transmit it, the first bit will still be en-route to the destination. i.e. if the transmission time is 5ms, and the propagation time is 6ms (which meets the conditions), and the current time is 5ms, the first bit will not yet have reached it’s destination (it takes 6ms).
6. Here we have the reverse scenario to that described in part e. The propagation time is less than the transmission time. This means that after the packet has been fully transmitted (the last bit has left the router), and we’re at the transmission delay time, the first bit will have already arrived at its destination as the transmission time is less than the total time necessary to propagate it to the destination.
7. Well, we know that dtrans is given as L/R and dprop is given as m/s. Plugging in numbers and setting the equations equal to one other we get m/2.5\*10^8 = 120/56. An m that would satisfy these two equations to be equal is ~535714285.71, or 5.35x108 if we can round and don’t care about significant digits.

4. Problem P10

Assuming no queuing delays, the total end-to-end delay for the packet is the number of nodes multiplied by dproc, dtrans and dprop all in parenthesis, generalized by equation 1.2 in the book, where it states dend-end=N(dproc + dtrans + dprop). Dtrans, of course, still equals L/R here where L is the packet size. Now, given the values for the packet size being 1,500 bytes, a propagation speed on all three links as 2.5x108m/s, a transmission rate of all three links being 2Mbps, a packet switch processing delay being 3msec, and the three link lengths being 5,000km, the second being 4,000km and the third being 1,000km, the total end to end delay with these values is calculated as follows. First, we can calculate the propagation delay. The propagation delay is defined as the d/s, where d is the distance between router A and router B and s is the propagation speed of the link. Calculating the propagations for the three distances (the speed is all the same) we get (5,000,000m / 2.5x108m/s) + (4,000,000m / 2.5x108m/s) + (1,000,000m / 2.5x108m/s), which ends up as (10,000km / 2.5x108m/s) total propagation delay. We can then calculate the transmission delay. For each link, the transmission rate is 2Mbps. To calculate the transmission delay we do L/R. So 1,500 bytes (0.012Mbps) / 2Mbps. So the total transmission delay here is 0.006 seconds per switch, or 0.018 for all three switches. The processing delay for the three switches is given as 0.003 seconds per switch, or 0.009 total. Adding all these delays together we get 40ms for the propagation delay, 18ms for the transmission delay and 9ms for the processing delay or 67ms total end to end delay.

5. Problem P23

1. The packet arrival time at the destination (the amount of time that elapses from when the last bit of the first packet arrives until the last bit of the second packet arrives) is equal to the total time it takes to propagate and transmit the first packet minus the total time it takes to transmit and propagate (and process both packets) through the entire link. If the packets are “identical” in terms of size, and the other delays are also identical, this will be essentially the total time it takes the second packet to transmit and propagate from the second link.
2. Yes, it is possible that the second packet queues at the input queue of the second link, as it’s possible due to the bottleneck that the router has not finished transmitting the first packet (resulting in the need for the second packet to queue. If there is a delay T before the second packet is sent, T must be larger than the time it takes for the first packet to transmit to the destination (i.e. greater than the bottleneck).

6. Problem P24

The best option here is to use FedEx overnight delivery. The reason, is that 40 terabytes divided by 100Mbps (megabits per second) results in 3,200,000 seconds. In hours, that around 888 or around 37 days to transmit the data over the link. In fact, if you wanted to save some money, you could do 3-5 day normal ground shipping and it would still arrive there significantly faster than transmitting the data over the link.

7. Problem P25

1. Propagation delay is calculated by d/s where d is the distance and s is the propagation speed. So 2Mbps x (20,000,000m / 2.5x108) = 0.16Mb bandwidth-delay product.
2. The total number of bits that can be sent in the link at any given time is 0.16Mb as calculated above. 800,000 bits is equivalent to 0.8Mb. So the entire message will not fit in the link, and the maximum number of bits present in the link will be 0.16Mb.
3. Based on the two above questions, the bandwidth-delay product means that for files that are sent all at once, it may not be able to fit that entire product on the link. It also prevents a larger amount of data from being sent over the link (in a speedy manner) as in problem b, the total message could not be sent on the link at once, causing a delay.
4. Well, if we divide the total distance (20,000,000 meters) by the bandwidth-delay product (160000 bits), we get an answer that the “width” of each bit, or how far each bit is spread out on the link to be 125 meters (which yes, is longer than a football field, which is technically 109.73 meters end zone to end zone[[1]](#footnote-2)).
5. A general expression for the width of a bit on the link (as calculated in part d) can be described as the length of the link m divided by the bandwidth-delay product. Now the bandwidth-delay product can described as the link speed multiplied by the distance divided by the propagation speed. Using the terms given, this equation then becomes .

8. Problem P28

1. To calculate the speed it takes to send the file, we need to add the transmission and the propagation delay together. Once again, the propagation delay is calculated by dividing the distance by the propagation speed (d/s). The transmission delay is calculated by the length of the packet divided by the transmission rate of the router (L/R). So in this case, the propagation delay can be calculated by doing 20,000,000m / 2.5x108 = 0.08s. The transmission delay is calculated by taking the 800,000 bits and dividing that by the link speed which is 2e+6 bits per second, which gives 0.4 seconds. In total, that’s 480ms total delay (or time spent) to deliver the packet to it’s destination.
2. The result here is that the total time from above carriers over (480ms). We have to add to that value, the time it takes to send the acknowledgement packet 20 times. Given that the transmission time for this packet is negligible, all we care about is the propagation time. To propagate a single packet back, it would take 0.08s (80ms). Multiplying that by 20, we get an additional 1600ms. Adding that to our original value of 480ms we get a total of 2080ms.
3. The value of b is *significantly* longer than that of a. Needing to propagate a packet back that distance over and over increases the total amount of time needed to transmit the packet between the two destinations.

9. Problem P31

1. The total time to move the message from source to destination if you ignore propagation, queuing and processing delays leaves the total time as just the transmission delay. This total time is 8x106 bits / 2x106 = 4 seconds total (given the standard formula for transmission delay being that you divide the packet length by the link speed). In figure 1.27, there are two routers that the packet needs to move through, so all in all, this value needs to be multiplied by three. So the total time it would take is 12 seconds.
2. The time it takes for a 10,000 bit packet to move from the first switch to the second switch is 10,000 / 2x106, or 5ms (calculated in the same way as part a with the simple difference being the length of the packet). The time that the second packet arrives at the first switch is 10ms (as it is the second packet in the queue, and the first one arrived 5ms beforehand.
3. The total amount of time it took to move the file from source host to destination essentially decreases by a third (to 4 seconds). The reason for this is that packets can be moving simultaneously.

10. Wireshark Lab

1. Three protocols that appear in the protocol column in the unfiltered packet-listing window are TCP, UDP and HTTP.
2. Less than one millsecond between the get message and the OK reply.
3. The internet address (IP?) of gai.cs.umass.edu is 10.6.10.6. The (current) IP of my PC is 128.119.245.12.
4. **Packets printed below:**
5. **PACKET ONE:**

378 2.294234723 10.6.10.6 128.119.245.12 HTTP 598 GET /wireshark-labs/INTRO-wireshark-

1. file1.html HTTP/1.1
2. Frame 378: 598 bytes on wire (4784 bits), 598 bytes captured (4784 bits) on interface 0
3. Linux cooked capture
4. Internet Protocol Version 4, Src: 10.6.10.6, Dst: 128.119.245.12
5. Transmission Control Protocol, Src Port: 42754 (42754), Dst Port: 80 (80), Seq: 1, Ack: 1, Len: 530
6. Hypertext Transfer Protocol
7. GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1\r\n
8. Host: gaia.cs.umass.edu\r\n
9. Connection: keep-alive\r\n
10. Cache-Control: max-age=0\r\n
11. Upgrade-Insecure-Requests: 1\r\n
12. User-Agent: Mozilla/5.0 (X11; Linux x86\_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/53.0.2785.92 Safari/537.36\r\n
13. Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,\*/\*;q=0.8\r\n
14. DNT: 1\r\n
15. Accept-Encoding: gzip, deflate, sdch\r\n
16. Accept-Language: en-US,en;q=0.8\r\n
17. If-None-Match: "51-53c858ad97a05"\r\n
18. If-Modified-Since: Thu, 15 Sep 2016 05:59:01 GMT\r\n
19. \r\n
20. [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html]
21. [HTTP request 1/1]
22. [Response in frame: 383]
23. 383 2.307869804 128.119.245.12 10.6.10.6 HTTP 309 HTTP/1.1 304 Not Modified
24. Frame 383: 309 bytes on wire (2472 bits), 309 bytes captured (2472 bits) on interface 0
25. Linux cooked capture
26. Internet Protocol Version 4, Src: 128.119.245.12, Dst: 10.6.10.6
27. Transmission Control Protocol, Src Port: 80 (80), Dst Port: 42754 (42754), Seq: 1, Ack: 531, Len: 241
28. Hypertext Transfer Protocol
29. HTTP/1.1 304 Not Modified\r\n
30. Date: Fri, 16 Sep 2016 03:21:48 GMT\r\n
31. Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.1e-fips PHP/5.4.16 mod\_perl/2.0.9dev Perl/v5.16.3\r\n
32. Connection: Keep-Alive\r\n
33. Keep-Alive: timeout=5, max=100\r\n
34. ETag: "51-53c858ad97a05"\r\n
35. \r\n
36. [HTTP response 1/1]
37. [Time since request: 0.013635081 seconds]
38. [Request in frame: 378]

**PACKET TWO:**

378 2.294234723 10.6.10.6 128.119.245.12 HTTP 598 GET /wireshark-labs/INTRO-wireshark-

file1.html HTTP/1.1

Frame 378: 598 bytes on wire (4784 bits), 598 bytes captured (4784 bits) on interface 0

Linux cooked capture

Internet Protocol Version 4, Src: 10.6.10.6, Dst: 128.119.245.12

Transmission Control Protocol, Src Port: 42754 (42754), Dst Port: 80 (80), Seq: 1, Ack: 1, Len: 530

Hypertext Transfer Protocol

GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1\r\n

Host: gaia.cs.umass.edu\r\n

Connection: keep-alive\r\n

Cache-Control: max-age=0\r\n

Upgrade-Insecure-Requests: 1\r\n

User-Agent: Mozilla/5.0 (X11; Linux x86\_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/53.0.2785.92 Safari/537.36\r\n

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,\*/\*;q=0.8\r\n

DNT: 1\r\n

Accept-Encoding: gzip, deflate, sdch\r\n

Accept-Language: en-US,en;q=0.8\r\n

If-None-Match: "51-53c858ad97a05"\r\n

If-Modified-Since: Thu, 15 Sep 2016 05:59:01 GMT\r\n

\r\n

[Full request URI: http://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html]

[HTTP request 1/1]

[Response in frame: 383]

383 2.307869804 128.119.245.12 10.6.10.6 HTTP 309 HTTP/1.1 304 Not Modified

Frame 383: 309 bytes on wire (2472 bits), 309 bytes captured (2472 bits) on interface 0

Linux cooked capture

Internet Protocol Version 4, Src: 128.119.245.12, Dst: 10.6.10.6

Transmission Control Protocol, Src Port: 80 (80), Dst Port: 42754 (42754), Seq: 1, Ack: 531, Len: 241

Hypertext Transfer Protocol

HTTP/1.1 304 Not Modified\r\n

Date: Fri, 16 Sep 2016 03:21:48 GMT\r\n

Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.1e-fips PHP/5.4.16 mod\_perl/2.0.9dev Perl/v5.16.3\r\n

Connection: Keep-Alive\r\n

Keep-Alive: timeout=5, max=100\r\n

ETag: "51-53c858ad97a05"\r\n

\r\n

[HTTP response 1/1]

[Time since request: 0.013635081 seconds]

[Request in frame: 378]

1. http://hypertextbook.com/facts/2001/NinTam.shtml [↑](#footnote-ref-2)