Bryan Arnold

9/23/18

CSE 3300

Homework 1

1. To see whether sending the data with a friend is the faster way to get the data to WSU, we need to calculate the total time to send 62 gigabytes over a 1 Mbps transfer link. 1 Mbps is equal to sending 8 bits per second when doing data transfer. Now, if we convert this to MBps, we can more easily see how fast this data transfer link will take. 1 byte = 8 bits, so 1 bit = 0.125 bytes. So, the transfer speed in MBps would be 0.125 MBps. Now, there are 1,000 megabytes in a gigabyte, and there are 62 total gigabytes to send. So, in total, we would need to send 62 * 1,000 = 62,000 total megabytes. Since we know our data transfer speed in MBps, we can find the total time it would take to completely transfer the data: 62,000 * 0.125 = 7,750 seconds. Now, to convert seconds to days, all we need to do is 7,750 / 60 (seconds in a minute) / 60 (minutes in an hour) / 24 (hours in a day) = 0.0896 total days.

   Based on this calculation, we can easily tell that using this data transfer link would be a MUCH faster way to get the data to WSU. Sending the data along with a friend would take 7 days, opposed to 0.0896 days.

2. For the first packet switch, the total time to send the message from A to the first packet switch is the total message size divided by the transfer link speed. Message size = 5 * 10^6 bits, and transfer link = 2 * 10^6 bits. So, do the calculation: (5 * 10^6) / (2 * 10^6) = 2.5 seconds.

   If the switches use store-and-forward their packet switching, the total time to send the message from Host A to Host B would be time to send one switch multiplied by the total number of times the message moves to another switch. We already calculated the time for one switch to process, 2.5 seconds, so we just need to multiply by the total times the message hops over to the next destination, 4 times total. So: 2.5 * 4 = 10 seconds to send the message from A to B.

3. A) For the first packet to reach the first switch from A, we need to know the size of each packet, since they're segmented, as well as the transfer link speed, 1 Mbps. We find the size of each packet by dividing the total message size by the number of segmentations. This would be 6 * 10^6 / 300 = 20,000 bits, or 2 * 10^4 bits. Now, we can find the total time for this to be sent with 2 * 10^4 bits / 1 * 10^6 bits = 0.003333 seconds, or 3.333 milliseconds.

B) To find how long it takes the second packet to the second switch, we need to know how long is takes for the first packet to reach the third switch. When the first packet reaches the third switch, the second packet will arrive at the second switch. This is because the second packet cannot advance until the first packet has moved. So, we need to multiply the time it takes the first packet to move to the next switch, 0.003333 seconds or 3.3333 milliseconds, by 3, the number of hops the packet makes. 0.003333 * 3 = 0.009999 seconds or 9.999 milliseconds. The second packet will move into the second switch at the same time, so the second packet will reach the second packet in 0.009999 seconds or 9.999 milliseconds, so roughly 10 milliseconds.

4. A) To find the total number of users that can be supported, we need to know the total bandwidth used, 2 Mbps, and the total bandwidth for each individual user, 200 kps or 0.2 Mbps. Now, we divide the total bandwidth used by the individual bandwidth for each user: 2 * 10^6 bits / 0.2 * 10^6 bits = 10 total users.

   B) We need to set up a binomial distribution for this question. The total users we can choose from is 100, and we need to choose n of them at any given point. Next, we need to know the probability that a user is transmitting, which is given to us as 15% or 0.15. Now, we can set up the binomial distribution to find the total probability of n users are transmitting at the same time:

$$\mathrm{P} = \binom{100}{n}(0.15)^n(0.85)^{100-n}$$

   C) To find the probability of 51 or more users transmitting at the same time, we need to

   Use the binomial distribution we just created for n greater or equal to 51. Since we need to find a probability that's greater or equal to, we must subtract from the whole probability, which is 1:

$$P(51\ or\ more\ users) = 1 - \sum_{n=0}^{100} \binom{100}{n} (0.15)^n (0.85)^{100-n}$$

   Now, we can reduce this to a normal Z variable, look at the z values of a table, to get the probability. Use the central limit theorem to reduce to:

$$P(51\ or\ more\ users) = 1 - P\left(\sum_{n=1}^{100} Xj \le 50\right)$$

$$1 - P\left(\sum_{n=1}^{100} Xj \le 50\right) = 1 - P\left(\frac{\sum_{n=1}^{100} Xj - 40}{(100 * 0.15 * 0.85)^{\frac{1}{2}}} \le \frac{10}{(100 * 0.15 * 0.85)^{\frac{1}{2}}}\right)$$

$$1 - P\left(\sum_{n=1}^{100} Xj \le 50\right) = 1 - P\left(Z \le \frac{10}{3.571}\right) = 1 - P(Z \le 2.80)$$

$$1 - P\left(\sum_{n=1}^{100} Xj \le 50\right) = \ 1 - \ .99744 = 0.00256$$

From the calculations, the probability of 51 or more users transmitting at the same time is roughly 0.00256, or 0.256% chance.

5. A) The server was able to successfully find the document, indicated by the “200 OK” component of the string at the beginning of the text. The document reply was provided on Friday September 14^{th}, 2018 at 10:45:12 GMT.

   B) The document was last modified on Sunday August 12^{th}, 2017 at 15:28:23 GMT.

   C) The number of bytes in the document being returned is 8,733 bytes, indicated in the “Content-Length: 8733” component of the text.

   D) The first 5 bytes of the document is <!doc. the server agreed to keep the connection alive from the request, indicted in the “Connection: Keep-Alive” component of the text.

6. A) To find the total average response time, we need to know the time to transmit an object as well as the access delay. To find the time to transmit, we need the total size of the object being sent, 600,000 bits, as well as the request rate of the server. The request rate is 15, but to understand this problem, we need to convert this into bits transferred per request, which is just 15 * 10^6. Now, we can find the transmit time by dividing the object size by the request rate: 6 * 10^5 bits / 15 * 10^6 bits = 0.04 seconds.

   For the access delay, we are given the equation Δ/(1 – ΔB), where Δ is the average time required to send an object over the access link and B is the arrival rate of objects to the access link. We know Δ = 0.04, as well as B = 15 from the previous step of this question. Now, we just must do the calculation:

   0.04(1 – (0.04 * 15)) = 0.016 seconds. We also know the time to receive a response is five seconds, so all we do is add this value to 5 seconds to get the average response time:

   5 + 0.016 = 5.016 seconds average response time.

   B) Now that we have a miss rate, the traffic intensity is reduced by 50% since 50% are handled by another network. The changes the average access delayed we just calculated by changing the value of ΔB. ΔB is now (0.5) * ΔB. So, we need to recalculate the access delay the same way we did in the previous step:

   0.04(1 – (0.5)(0.04 * 15)) = 0.028 seconds. Half of the time the cache will fulfill a request with 0 response time, and the other half of the time the average response time will be the given 5 seconds plus the access delay we just calculated: 5 + 0.028 = 5.028 seconds. So, to find out the final response time we multiply the probabilities of each situation by the respective average response times:

(0 seconds wait)(0.5 chance) + (5.028 seconds wait)(0.5 chance) = 2.514 seconds total response time.