**[F21] CS 3640: Introduction to Networks and Their Applications**

**Group Members: Johan Avila, Arjun Kanwar, Reese Thornburg, Reagan Reid**

Assignment 2/Part 2: Theoretical Understanding of the Link Layer [40 points]

1. Calculating queue characteristics [10 points]
   1. Every second, a 100 bits arrive uniformly until .1 seconds have passed (i.e., each bit arrives 1/1000th of a second after the previous). The departure rate can never exceed 750bps. What is the maximum occupancy and average occupancy of the queue? What is the maximum delay and average delay due to queueing? Explain your work.

To calculate this, first we must find the traffic intensity. I = La/R. L in this case is 100, and a is assumed to be 1. R in this case is 750. I = 100/750, making I = .13. To find the delay we use I(L/R)(1-I), which is .13(100/750)(1-.13), which gives us 0.01508, the average delay. The average occupancy of the queue is 0 since the queue never fills up. The theoretical max is 750 since that’s the departure rate.

* 1. What happens when 100 bits arrive all at once at the start of each second? What is the average and maximum occupancy of the queue? What is the average and maximum delay due to queueing? Explain your work.

When 100 bits arrive all at once, a goes from 1 to 100, making the intensity equation I = 100(100)/750, or 1.3. This is bad as traffic will build up, but since only 100 bits are being transferred, the delay doesn’t build up after the first transfer. The delay is .0507 for this case.

* 1. What happens if a 1000 bits arrive at the queue every second?

The arrival rate outweighs the departure rate, causing an exponential buildup in the queue. Every second, another 250 bits is added to the queue, which will cause massive slowdown and eventually packet loss.

1. Calculating queue characteristics at extremes [10 points]
   1. Consider a queue with a continuous time arrival process that is constrained so that in any time interval t1 to t2 (where t2 – t1 = t > 0) no more than (s+rt) bits can arrive, where s and r are non-negative constants. The output link operates at a constant rate of 2r. What is an upper bound on the **long-term average rate** that bits can arrive to the queue? Explain your work.

Looking at these variables, it can be deduced that the higher r is, the more bits can arrive and depart. In order for the queue to not overflow the input must remain less than or equal to the output. Therefore, the long term average rate of bits that can arrive in a queue can be represented by the function 2r where r >= 1.

* 1. Consider a queue with a continuous time arrival process that is constrained so that in any time interval t1 to t2 (where t2 – t1 = t > 0) no more than (s+rt) bits can arrive, where s and r are non-negative constants. The output link operates at a constant rate of 2r. **What is the maximum occupancy of the queue**? Explain your work.

In order to get the max occupancy, s must be equal to r. Represented as a function, s+rt would be r(t+1) assuming s = r. Therefore, this would be the max occupancy as it fills the queue without causing an overflow.

1. Calculating end-to-end delays [10 points]
   1. We wish to send a message of size 150,000 bytes over the network. There are four hops, each of length 200km and running at 100 Mb/s. However, before sending we split the message into 1500 byte packets. Assume that packet transmission doesn’t begin until all bits in the packet have arrived. What is the end-to-end delay of the message? Use speed of light in copper as 2\*10^8 m/s, and round your answer to the nearest integer millisecond and explain your work.

To get the end-to-end delay, we must sum the transmission, propagation, processing, and queueing delays. The transmission delay is 1500/100, giving us 15ms. Each hop also has a propagation delay of 1ms each. Each packet would take 16\*4ms since there are 4 hops, meaning each packet takes 64ms to arrive. Since there are 10 packets, this would give us a total end-to-end delay of 640ms.

* 1. What happens if we do not break this down into 1500 byte packets and bits may leave as they arrive.

If not broken down, transmission and propagation will have to be done as is. This makes the transmission delay 1500ms. Multiplying this by four and adding the propagation delay results in 6004ms.

4. Hubs and switches [10 points]

Here are two snippets of traffic flows observed on 4 hosts (h10, h11, h00, and h20). Which snippet came from a network connected by a hub? Which snippet came from a network connected by a switch?

Snippet A is a switch since it does a broadcast and then paths to another value. Snippet B is a hub since some lines only perform broadcasts without an intended receiver, which only hubs can do.

Note: You will need to understand the ARP protocol, tshark commands and trace formats (see [here](https://www.wireshark.org/docs/wsug_html_chunked/AppToolstshark.html) to understand what the commands are doing and use tshark to understand the outputs).

snippet A

H10 > sudo tshark -a duration:8 arp

1 0.000000000 2a:ed:41:12:eb:fd → Broadcast ARP 56 Who has 10.0.0.4? Tell 10.0.0.1

2 0.000153585 8e:97:6c:4e:5f:b7 → 2a:ed:41:12:eb:fd ARP 42 10.0.0.4 is at 8e:97:6c:4e:5f:b7

H11 > sudo tshark -a duration:8 arp

1 0.000000000 2a:ed:41:12:eb:fd → Broadcast ARP 56 Who has 10.0.0.4? Tell 10.0.0.1

2 0.000156023 8e:97:6c:4e:5f:b7 → 2a:ed:41:12:eb:fd ARP 42 10.0.0.4 is at 8e:97:6c:4e:5f:b7

H00 > sudo tshark -a duration:8 arp

1 0.000000000 2a:ed:41:12:eb:fd → Broadcast ARP 56 Who has 10.0.0.4? Tell 10.0.0.1

2 0.000417378 8e:97:6c:4e:5f:b7 → 2a:ed:41:12:eb:fd ARP 42 10.0.0.4 is at 8e:97:6c:4e:5f:b7

H20 > sudo tshark -a duration:8 arp

1 0.000000000 2a:ed:41:12:eb:fd → Broadcast ARP 56 Who has 10.0.0.4? Tell 10.0.0.1

2 0.000013853 8e:97:6c:4e:5f:b7 → 2a:ed:41:12:eb:fd ARP 42 10.0.0.4 is at 8e:97:6c:4e:5f:b7

snippet B

H10 > sudo tshark -a duration:8 -f arp

1 0.000000000 d6:6e:6c:bc:b0:ef → Broadcast

ARP 56 Who has 10.0.0.4? Tell 10.0.0.1

H11 > sudo tshark -a duration:8 -f arp

1 0.000000000 d6:6e:6c:bc:b0:ef → Broadcast

ARP 56 Who has 10.0.0.4? Tell 10.0.0.1

H00 > sudo tshark -a duration:8 -f arp

1 0.000000000 d6:6e:6c:bc:b0:ef → Broadcast ARP 56 Who has 10.0.0.4? Tell 10.0.0.1

2 0.082946897 de:32:b2:1f:6e:55 → d6:6e:6c:bc:b0:ef ARP 42 10.0.0.4 is at de:32:b2:1f:6e:55

H20 > sudo tshark -a duration:8 -f arp

1 0.000000000 d6:6e:6c:bc:b0:ef → Broadcast ARP 56 Who has 10.0.0.4? Tell 10.0.0.1

2 0.000017022 de:32:b2:1f:6e:55 → d6:6e:6c:bc:b0:ef ARP 42 10.0.0.4 is at de:32:b2:1f:6e:55

Part 2 was worked on by Johan Avila and Arjun Kanwar

Johan - Questions 1, 3, and 4

Arjun - Question 2