Computer Communications and Networks (COMN) 2018/19, Semester 2

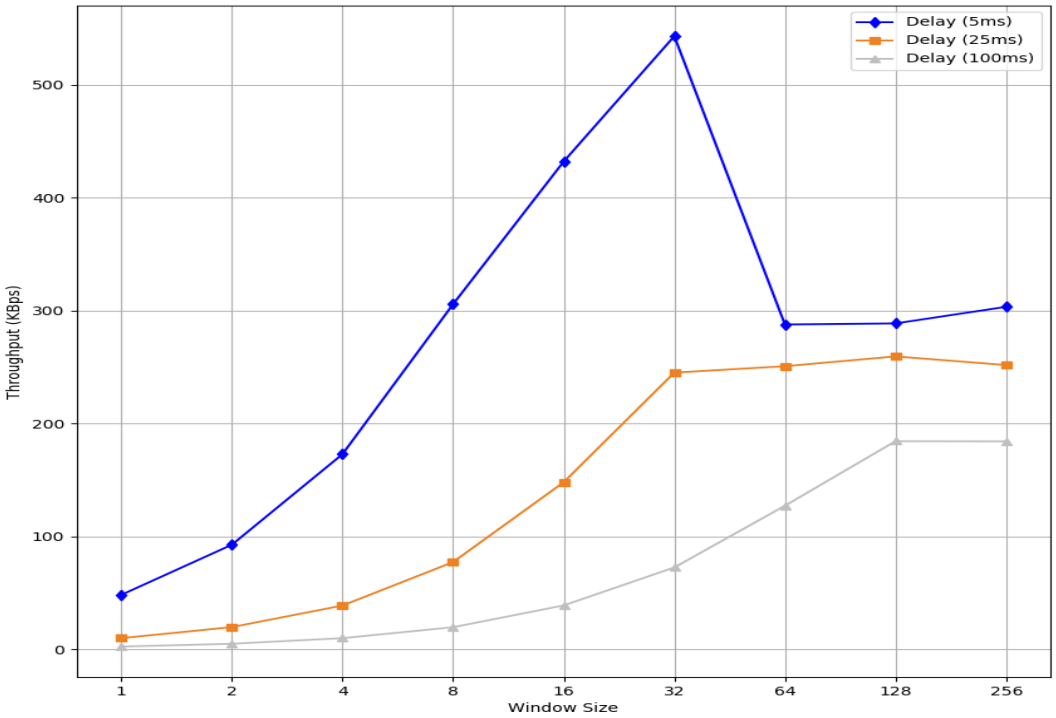
Assignment Part 2 Results Sheet

Forename and Surname:	Akshay Chandiramani
Matriculation Number:	s1558717

Question 1 – Experimentation with Go-Back-N. For each value of window size, run the experiments for **5 times** and write down **average throughput**.

	Average throughput (Kilobytes per second)		
Window Size	**Delay = 5ms**	**Delay = 25ms**	**Delay = 100ms**
1	48.27	9.93	2.49
2	92.69	19.73	4.97
4	172.91	38.81	9.93
8	305.79	77.34	19.63
16	432.19	148.04	38.92
32	543.25	245.19	72.57
64	287.83	250.84	127.25
128	288.76	259.54	184.43
256	303.47	251.86	184.29

Create a graph as shown below using the results from the above table:



Question 2 – Discuss your results from Question 1.

For a 5ms propogation delay, the througput peaks around a window size of 32 with a 40ms timeout. This makes sense as the the packet loss rate is only 0.5%, so a large window of UnAcked packets with a retransmission timeout much greater than the propogation delay allows a faster throughput. If the window is too small, fewer UnAcked packets are allowed and this may result in an increased number of retransmissions. However, having a too large window for such a small progogation delay decreases the throughput as a possibly large number of packets have to be retransmitted if a packet is lost. However, if we increase the propogation delay the throughput is better for a larger window size like 128 or 256, because fewer packets will have to be retransmitted with a retransmission timeout (as fewer packets are sent in that time). A 15ms retransmission timeout proved to be ideal for the 25ms and 100ms propogation delay.

Question 3 – Experimentation with Selective Repeat. For each value of window size, run the experiments for **5 times** and write down **average throughput**.

	Average throughput (Kilobytes per second)
Window Size	**Delay = 25ms**
1	19.47
2	29.37
4	48.09
8	86.31
16	160.31
32	250.23

Question 4 - Compare the throughput obtained when using “Selective Repeat” with the corresponding results you got from the “Go Back N” experiment and explain the reasons behind any differences.

The throughput for Selective repeat is slightly higher than that for Go Back N because selective repeat is an algorithm that significantly reduces the number of packets retrasmitted as only the packets that have not been acknowledged are retrasnmitted vs all packets (after the oldest packet that timedout) being retrasnmitted. However, the improvement is only marginal as the propogation delay is quite significant and therefore packets are sent slowly and the number of packets needed to be retransmitted in the Go Back N algorithm (if a packet is lost) are quite low for this case.

Question 5 – Experimentation with *iperf*. For each value of window size, run the experiments for **5 times** and write down **average throughput**.

Window Size (KB)	Average throughput (Kilobytes per second) Delay = 25ms
1	54.17
2	66.87
4	78.55
8	83.61
16	84.25
32	74.13

Question 6 - Compare the throughput obtained when using “Selective Repeat” and “Go Back N” with the corresponding results you got from the *iperf* experiment and explain the reasons behind any differences.

Iperf relies on TCP which is a connection oriented protocol, which makes a handshake before sending packets. Selective repeat and Go Back N are based on UDP, a protocol that doesn’t make connections before sending packets. For smaller window sizes, TCP is much faster because it as it is much more reliable, makes much better use of the available bandwith and will result in lesser retranmissions because of packet loss. However, at a higher window size, TCPs congestion control comes into play and it buffers a lot of data to send bigger packets, vs UDP that will keep pushing out packets whenever possible.