

Computer Communications and Networks (COMN)

2022/23, Semester 1

Assignment 2 Worksheet

Forename and Surname:	Nikita Peleshatyi
Matriculation Number:	2150635

Question 1 – Number of retransmissions and throughput with different retransmission timeout values with stop-and-wait protocol. For each value of retransmission timeout, run the experiments for **5 times** and write down the **average number of retransmissions** and the **average throughput**.

Retransmission timeout (ms)	Average number of retransmissions	Average throughput (Kilobytes per second)
5	2844	78.29
10	1326	75.58
15	107	69.97
20	98	68.70
25	96	67.09
30	100	63.15
40	98	57.84
50	93	57.01
75	94	48.74
100	92	44.25

Question 2 – Discuss the impact of retransmission timeout value on the number of retransmissions and throughput. Indicate the optimal timeout value from a communication efficiency viewpoint (i.e., the timeout that minimizes the number of retransmissions while ensuring a high throughput).

Short timeout -> not enough time for a packet to be delivered and for its ACK to travel back, causing a large number of retransmissions. It increases the throughput as there's very little delay between distinct packets, however risks flooding the network with retransmissions hence large number of them.

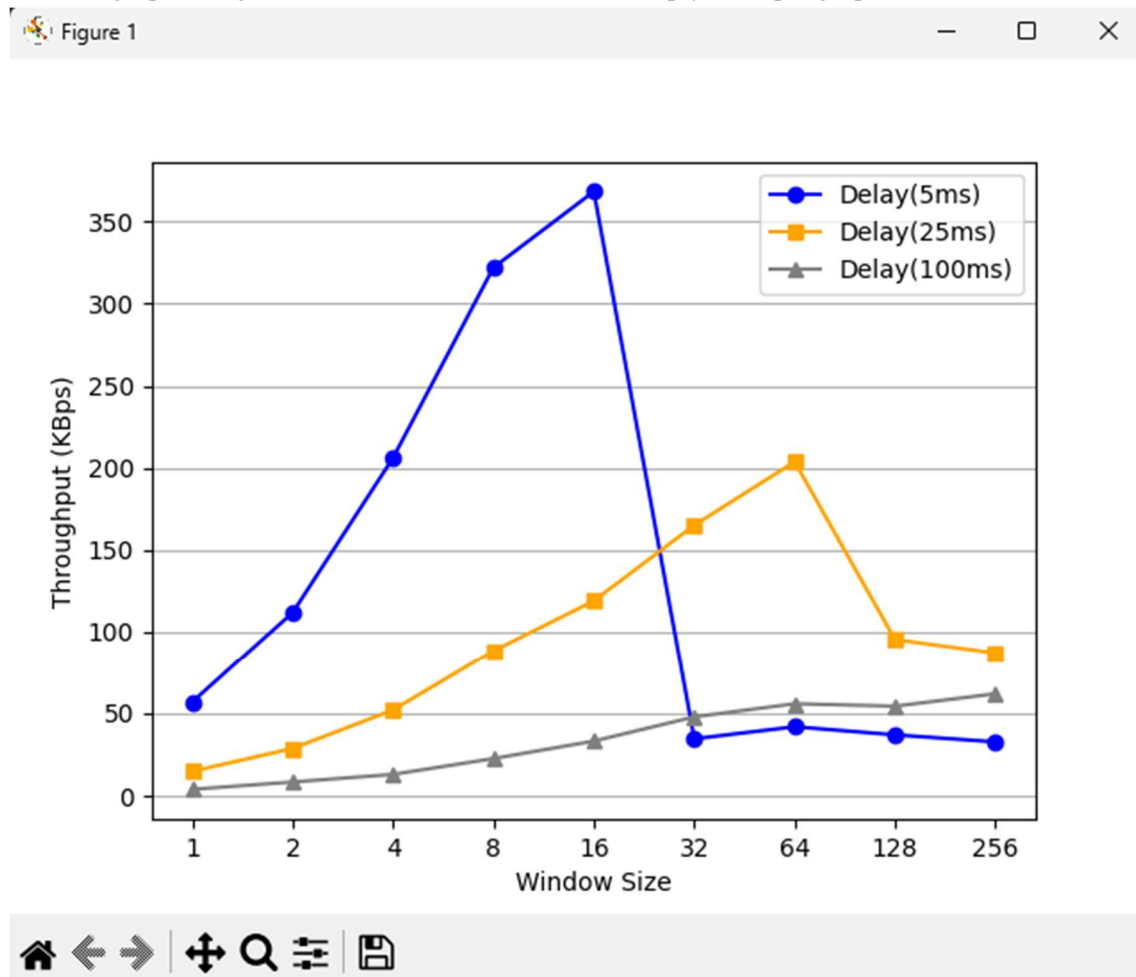
Longer timeout -> less retransmissions since we give ACKs more time to get back to the sender, however this stalls the program in general and reduces the throughput accordingly.

For a 5ms one-way propagation delay, 15ms seems to be the optimal value providing the highest throughput with the lowest number of retransmissions. This makes sense theoretically and is verified by experiments. It takes 10ms for the packet and its ACK to make their way and an additional 5ms buffer for processing ensures the optimal setting for this data transfer. Values below 15 have an explosive growth in the number of retransmissions with low increase in throughput, while values above 16 significantly cut the throughput at a negligible reduction in the number of retransmissions.

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

Window Size	Average throughput (Kilobytes per second)		
	Delay = 5ms	Delay = 25ms	Delay = 100ms
1	56.94	14.89	4.00
2	111.74	28.74	8.45
4	206.7	52.09	13.05
8	321.99	88.34	22.65
16	368.32	118.97	33.24
32	34.63	165.07	47.90
64	41.91	203.72	55.85
128	37.01	95.28	54.39
256	32.74	87.05	61.92

Create a graph using the results from the above table (empty example graph shown below):



Question 4 – Discuss your results from Question 3.

The higher the delay, the lower the throughput as can be seen in the graph - 5ms for max tput and 100ms for min. Larger window size tends to increase the throughput as more packets can be in-transit at any time up until a certain value between 16 and 32, where it peaks and then suddenly drops to low throughput values, where all delays seem to converge to a range values between 50 and 100 KBps. Optimal window size is 16.

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

Window Size	Average throughput (Kilobytes per second)
	Delay = 25ms
1	15.58
2	27.18
4	54.37
8	90.72
16	165.13
32	278.57

Question 6 - 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.

Under similar conditions (25ms one-way delay, 10Mbps bandwidth, 5% packet loss, 60ms timeout and varied window sizes), both GBN and Selective repeat perform closely similarly on low sizes (1,2,4,8), however as the window size becomes larger, Selective Repeat outperforms GBN and shows a much higher – almost a 1.5 increase – which is thanks to the selective repeat algorithm that instead of retransmitting the entire window (16/32 packets in this case) only selectively resends presumably lost packets, allowing the window to move faster without flooding the network with duplicate packets, resulting in a higher throughput value.

Question 7 – Experimentation with *iperf*. For each value of window size, run the experiments for **5 times** and write down the **average throughput**.

Window Size (KB)	Average throughput (Kilobytes per second)
	Delay = 25ms
1	16.02
2	27.28
4	52.04
8	74.3
16	91.4
32	168.34

Question 8 - 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.

At lower window sizes, iperf seems to perform worse than both Selective Repeat and GBN, albeit not too far off. But as the window size increases, it tends to be similar to GBN, hinting that perhaps iperf's implementation of TCP involves a Go Back N implementation rather than a Selective Repeat