"I have read and understood the course academic integrity policy located under this link:

http://www.cse.buffalo.edu/faculty/dimitrio/courses/cse4589_f14/index.html#integrity"

2)

ABT: The timeout value that best suited my protocol implementation is 4.0. The reason behind choosing this value is that for the value higher than 4, with high loss probability the number of packets getting delivered to application layer of B was significantly less which was directly affecting the throughput. And for the value less than 4 the packets were getting retransmitted at the higher rate, which again was affecting throughput badly. The optimum timeout value came out to be 4 for my implementation which was giving me maximum throughput with minimum retransmission of packets.

GBN: The best timeout value that suited this protocol implementation was 10.0.For the value greater than 10 with less window size the retransmission rate was low, thus resulting in less packet delivery rate at receiver's end. And for larger window size number of retransmission was high resulting in less throughput. The timeout value 10 was optimum for this protocol implementation as it gave maximized throughput for all window size with maximum loss probability. However, throughput suffered badly for the loss probability of 0.8 with window sizes 200 and 500 because number of retransmission was too high with these window sizes and loss probability. Although a good throughput was obtained for all other loss probabilities.

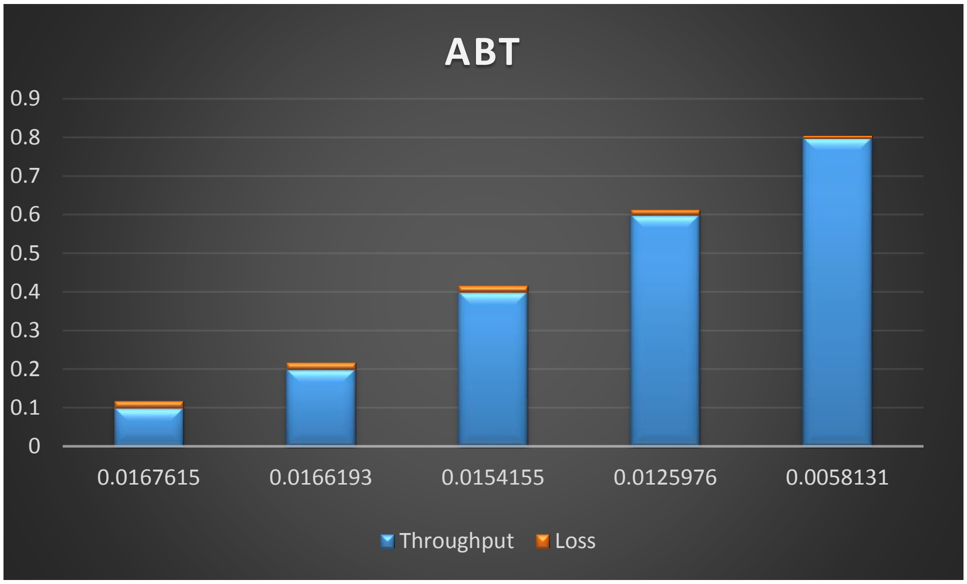
SR: The timeout value that best fit this protocol implementation is 40.0. As per my understanding as in SR protocol the receiver too buffers the packet and sends the acknowledgment for the packet received within the window, setting the higher value proved to be efficient as compared to above two protocols. As lower timeout value resulted in high number of transmission and higher timeout value resulted in less packet delivery late. The value 40.0 was optimum for higher throughput and packets delivered to application layer of B as compared to other values. Also as the window size increased this timeout value remained efficient same as for lower window size with all loss probabilities.

3)

For multiple timers in SR with the availability of single hardware timer, I started the timer at A's initialization with some small value in the second parameter of starttimer() function which will result the flow to go to timerinterrupt() function. In this function I am checking the timeout value of each packet. If this value is less than the (time - TIMEOUT) value, then the packet is retransmitted. And the packets whose value is not less than TIMEOUT value will not be transmitted as their timer is still running. 'time' here is the variable used in the simulator to generate event time and TIMEOUT value is 40.0. Note that the initial timer value of each packet is assigned with the variable 'time' which is used in the simulator for calculating the event time. As the packets whose timer expired are retransmitted again the timer starts for all those packets. In this way a single timer is used to implement individual timers of each packet.

4.2.1

(a) Alternating Bit Protocol

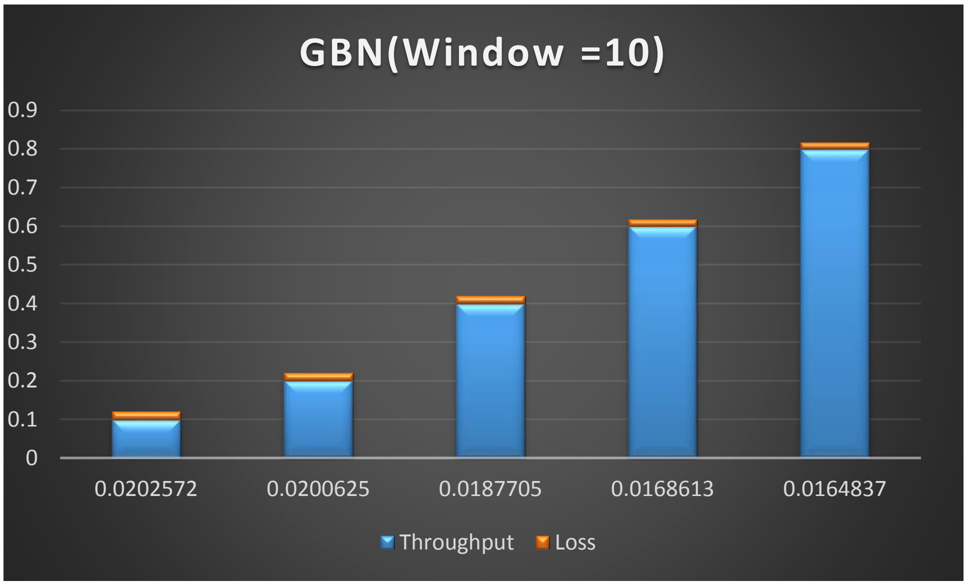


With increasing loss probability, loss of throughput was expected in ABT.

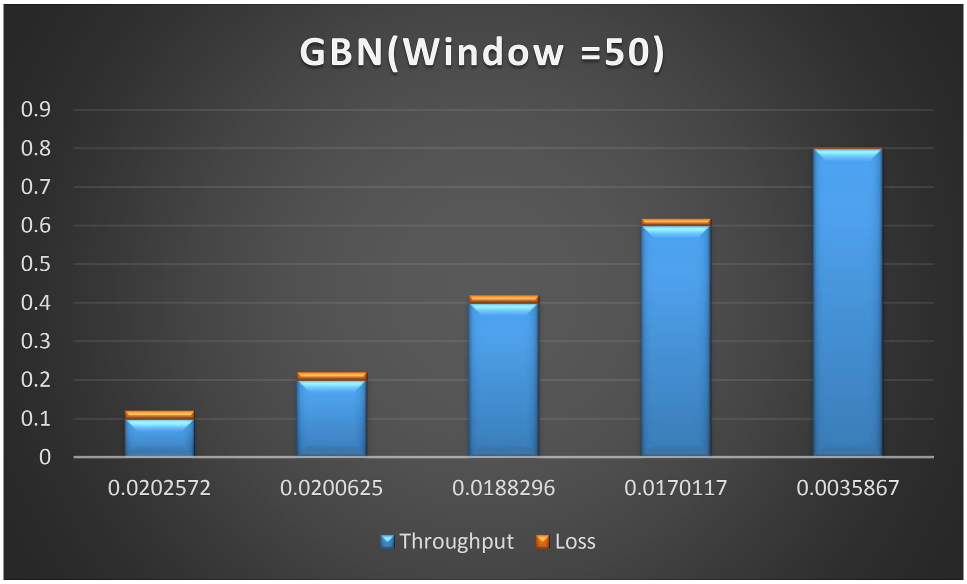
The graph above depicts that as the loss probability goes on increasing the throughput decreases.

The above measurements goes with what was expected.

(b) Go Back N Protocol

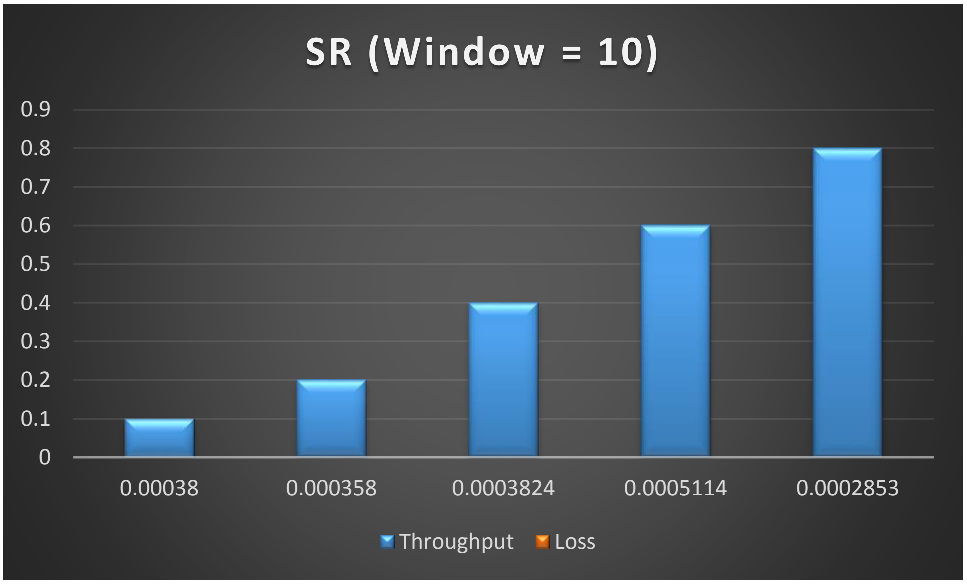


With increasing loss probability loss of throughput was expected at constant rate. But the value of throughput does not decrease significantly as expected. As we can see the value at loss 0.1 and 0.2 and at loss 0.6 and 0.8 are almost same. The measurements returned by protocol is appropriate as throughput is decreasing though not significantly.

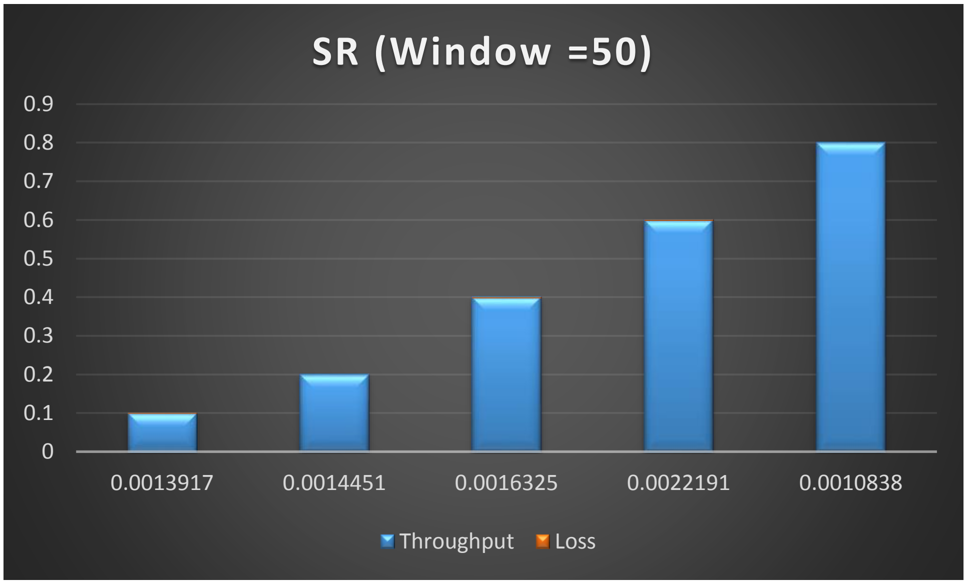


Loss of throughput with increasing loss probability was expected. The graph above depicts that the throughput is decreasing with increasing loss probability. The result obtained for loss probability 0.8 was abrupt than was expected as there is significant loss in throughput as compared to other throughputs at different loss probability. The measurement up-to loss probability 0.6 seems appropriate but at 0.8 its drastic change which was not expected.

(c) Selective Repeat Protocol



Loss of throughput with increasing loss probability was expected. The throughput decreases as expected but at loss probability 0.4 throughput is greater than at 0.2 and then again increases at 0.6. This was unexpected. Rest measurements are appropriate as per expectation.



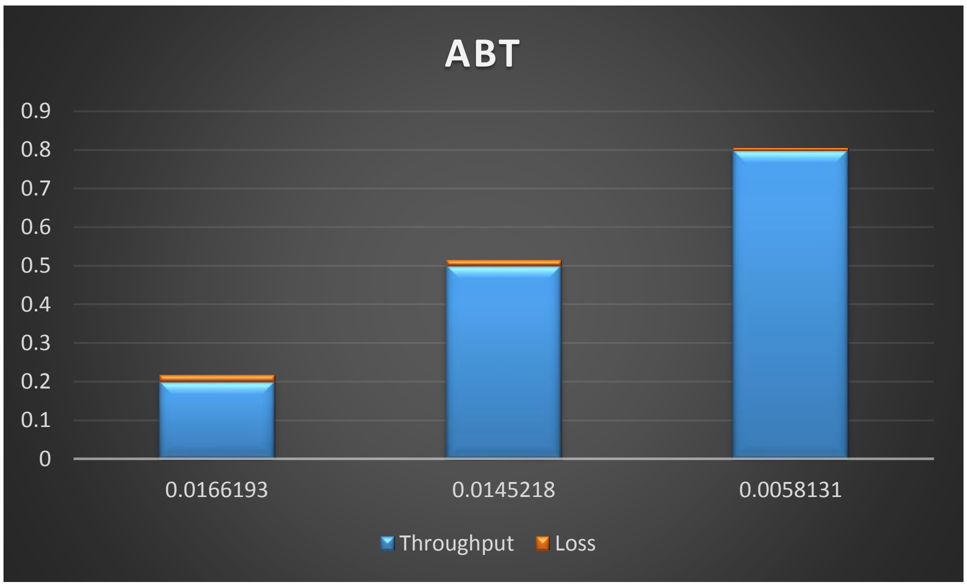
With the window size of 50, less decrease in throughput was expected.

The graph shows that the throughput decreases at very minute rate which was expected.

Although there was gradual decrease in throughput at loss probability 0.8.

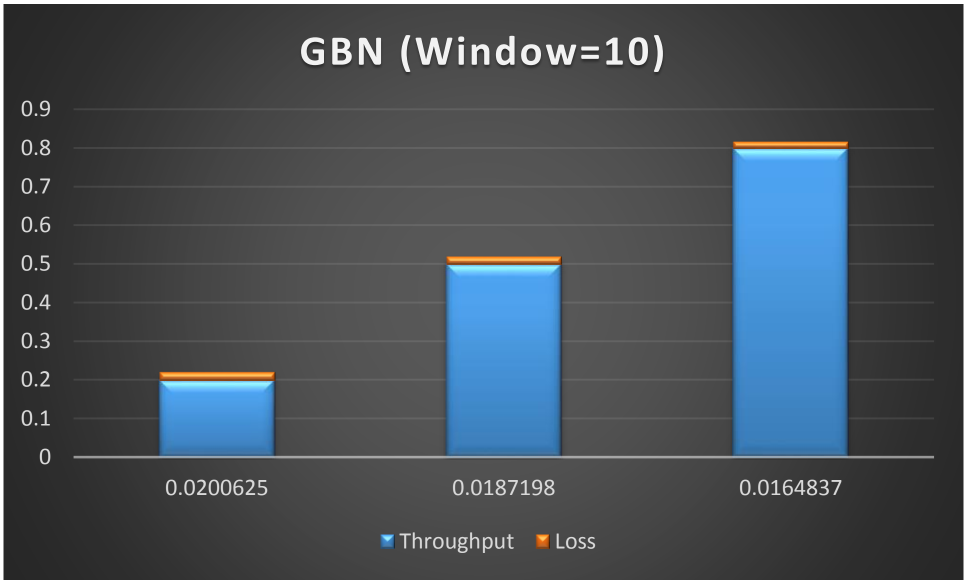
4.2.2

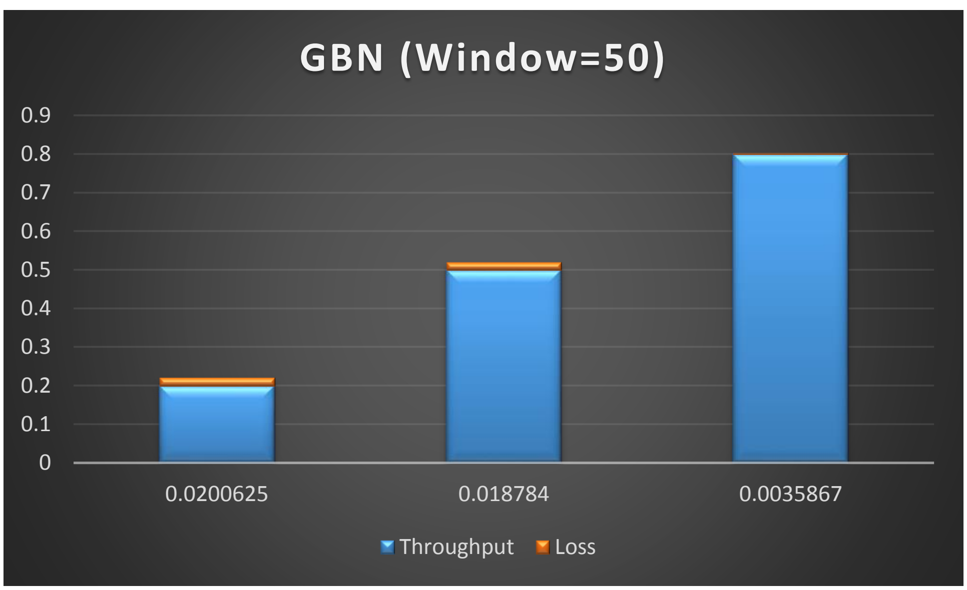
(A) Alternating bit Protocol

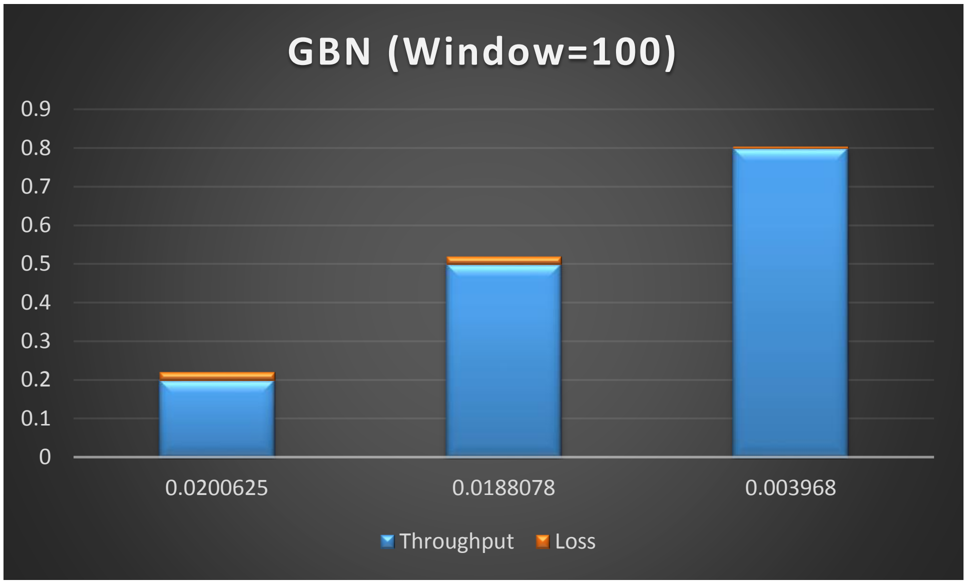


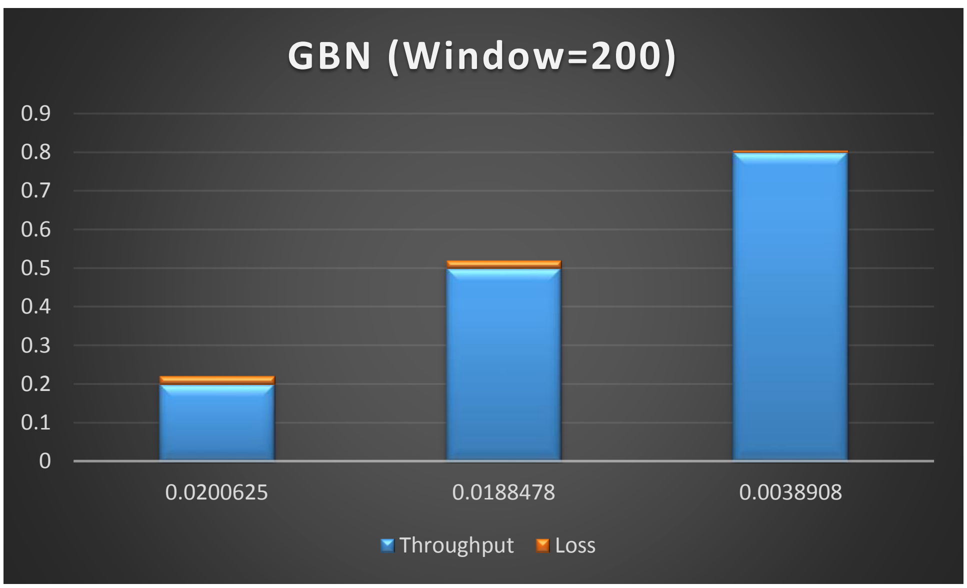
With ABT, loss of throughput with increasing loss probability was expected. The graph shows that the throughput decreases as loss probability increases from 0.2 to 0.5. But there is a gradual decrease in the throughput when loss probability switches from 0.5 to 0.8. The measurement at 0.8 loss probability seems disturbing.

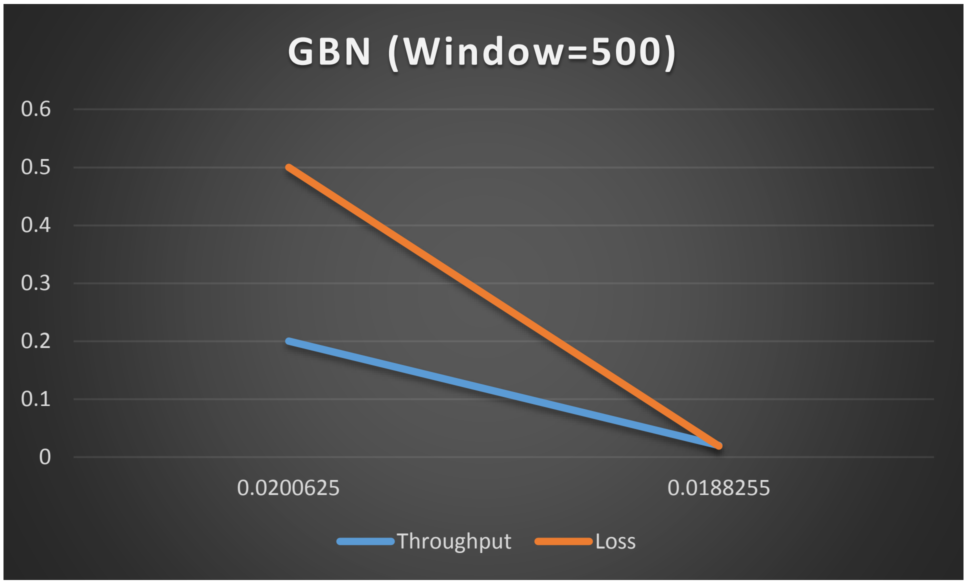
(B) Go Back N Protocol





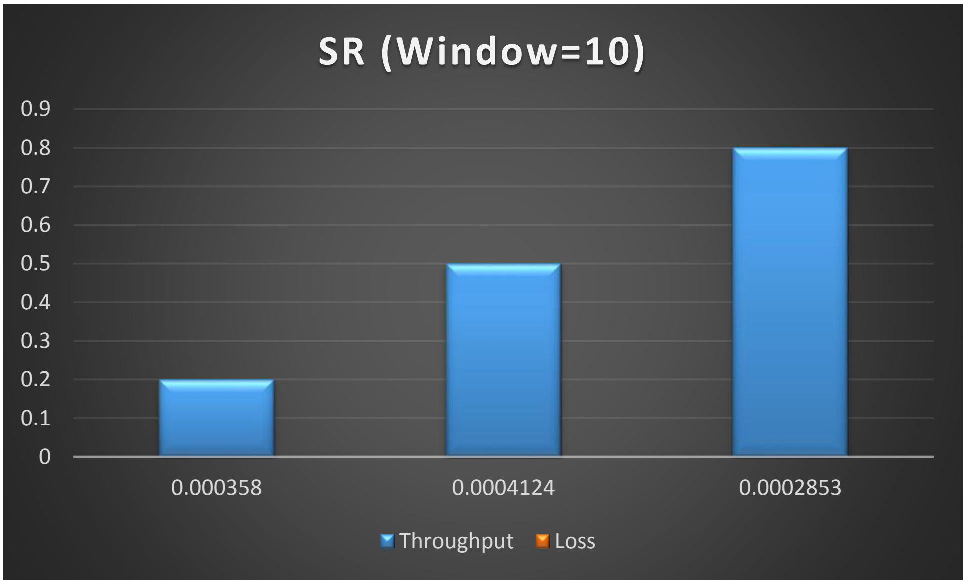

GBN (Window=100)
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.0200625
0.0188078
0.003968
Throughput
Loss

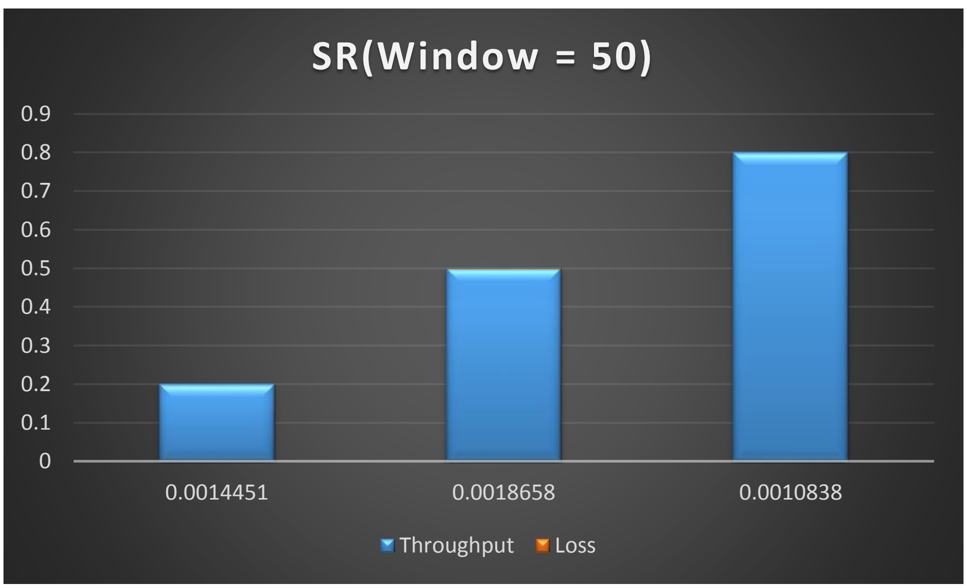

GBN (Window=200)
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.0200625
0.0188478
0.0038908
Throughput
Loss

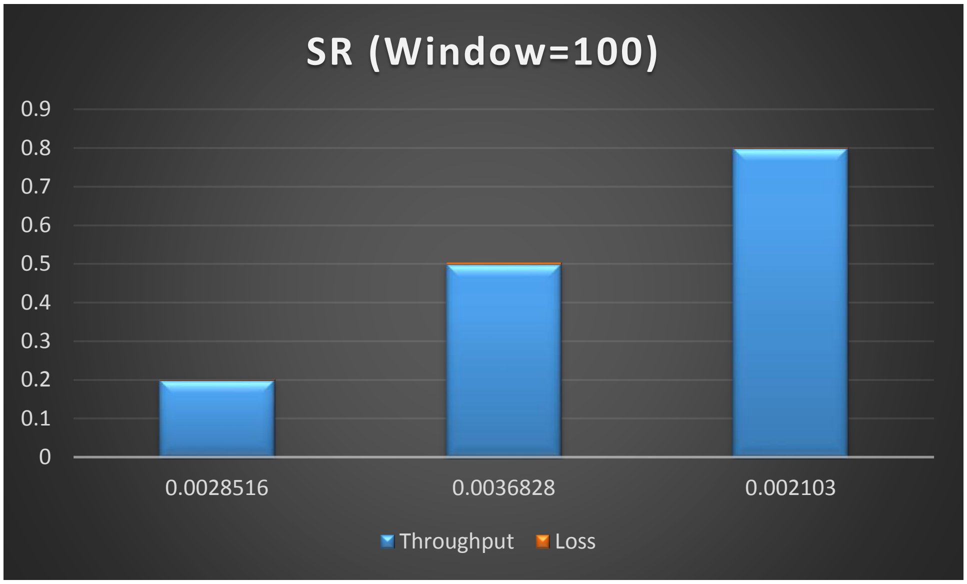


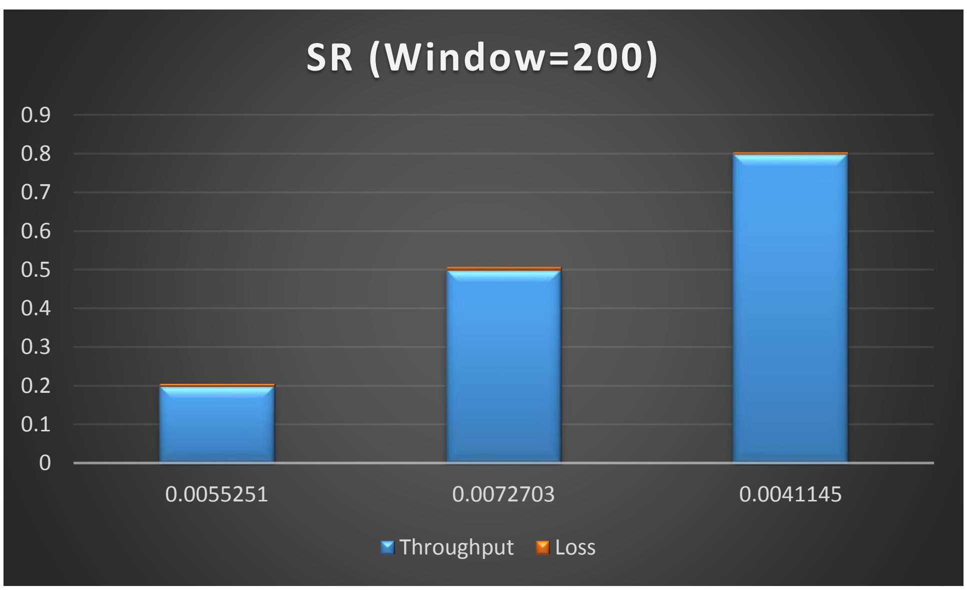
With increasing loss probability and increasing window size, number of retransmission increases in GBN which harshly affects the throughput. As we can see from the graph above the value of throughput at loss probability 0.8 gradually decreases. This is because of implementation of GBN which sends all packets in the window when timeout occurs. The loss of throughput was expected which we got as we can see from the graphs above. The measurements seems quite appropriate.

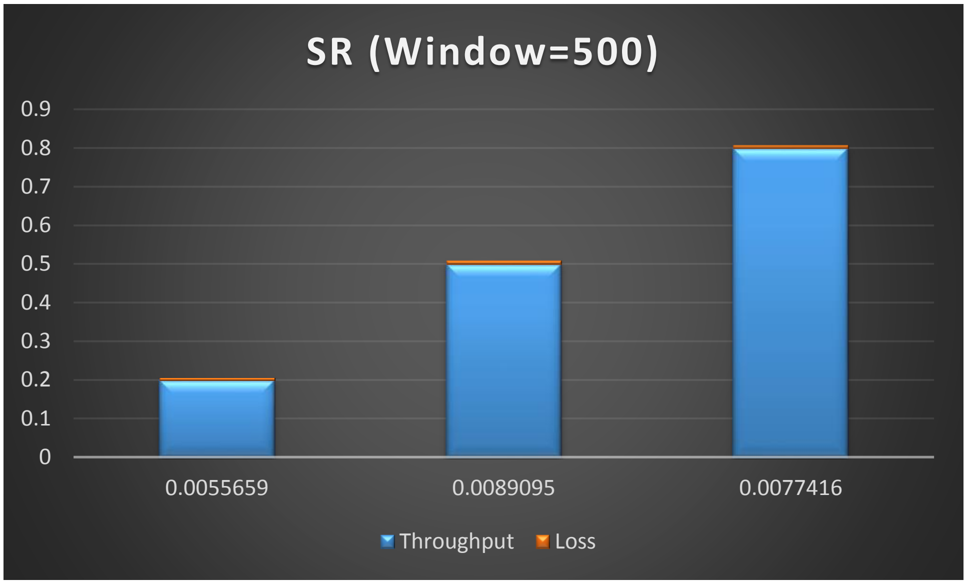
(C) Selective Repeat Protocol






SR (Window=100)
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.0028516
0.0036828
0.002103
Throughput
Loss


SR (Window=200)
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.0055251
0.0072703
0.0041145
Throughput
Loss



As Selective repeat buffers the acknowledgment of the packet and sends only the packet which is either lost or corrupted on the receiver's side, the number of retransmissions are less which results in good throughput. As the loss probability increases with window size the loss of throughput was expected but not a steep decrease as in case of GBN. The graph above depicts that throughput falls with increasing loss probability but does not decreases abruptly or steeply. This was expected from SR implementation and was retrieved.