Analysis_Assignment2

I have read and understood the course academic integrity policy.

Time out scheme:

Timeout scheme chosen for the experiments is 20 units. Since it is mentioned that the approximate time the packet takes to reach the receiver from sender is 5-time units that makes RTT 10 units. we can't take the timeout value to be too small (10-15units) because we don't want to make unnecessary retransmissions whenever a packet is delayed slightly due to congestion or queuing delays and processing the checksum and creating acks.

Also, we don't want to take high values of timeout, though it reduces unnecessary retransmission, because in our channel we are sending the data frequently and if some packet is lost we don't want to wait for too long as it limits the number if messages that we can transmit. Which decreases the throughput.

Timer implementation:

Maintain a buffer_list for packets that are sent to layer3. Create a list of size equal to the window size to store the time at which each packet is sent to layer 3corresponding to the entry in the buffer_list.

Whenever a packet is sent to layer 3, store the time (T_send) at the end of the list.
If the timer list is empty before sending the packet that means no packets are in queue, start the timer with timeout interval declared for the packet that is currently being sent.

If timer goes off:

Retransmit the packet for which timer went off. Update the time at which the packet is transmitted I the timer list, and starttimer for the packet which is next in the queue with value (timeoutvalue -timein link) calculate time in link by subtracting packet sent time from current simulator time.

If Ack is received successfully:

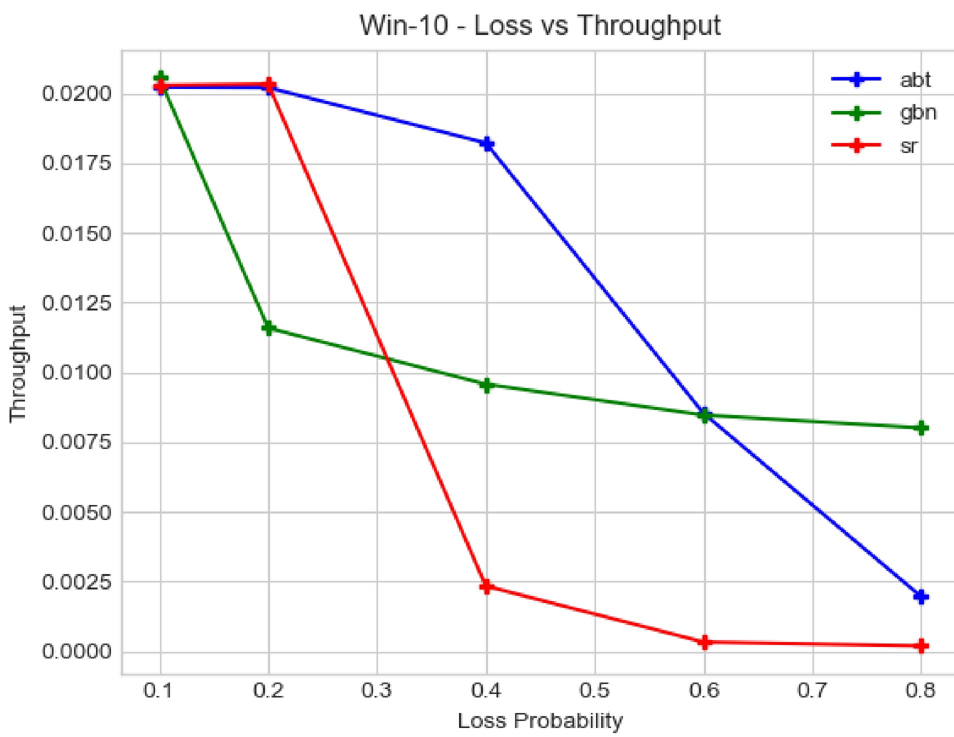
If the timer started is for the packet for which ack is received stop timer and starttimer for next successive packet in the window. If the ack received is not for the packet for which the timer is running remove timelist and buffer entries for the packet.

Experiments 1:

With loss probabilities: {0.1, 0.2, 0.4, 0.6, 0.8}, compare the 3 protocols' throughputs at the application layer of receiver B. Use 2 window sizes: {10, 50} for the Go-Back-N version and the Selective-Repeat Version.

Graph1:

window	loss	GBN	SR	ABT
10	0.1	0.020561	0.020289	0.020228
10	0.2	0.01159	0.020337	0.02021
10	0.4	0.009576	0.002338	0.018226
10	0.6	0.008467	0.000331	0.008532
10	0.8	0.008011	0.000195	0.001958

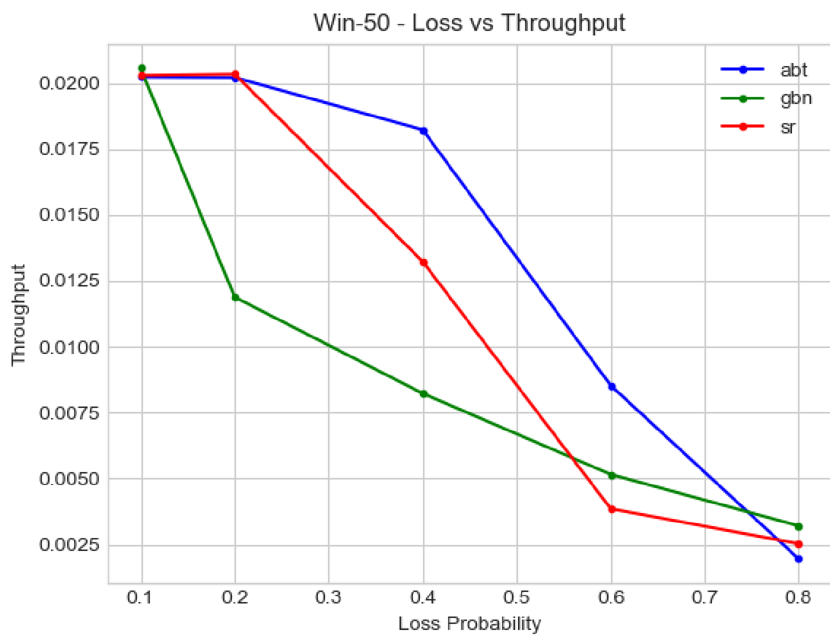


GBN performs better at lower window sizes when compared to ABT and SR in better channel conditions.

Expectation was that GBN and SR perform similarly in all channel conditons.

Graph 2:

window	loss	GBN	SR	ABT
50	0.1	0.020561	0.020289	0.020228
50	0.2	0.011882	0.020337	0.02021
50	0.4	0.00823	0.013229	0.018226
50	0.6	0.005162	0.003859	0.008532
50	0.8	0.00321	0.002533	0.001958



Observations:

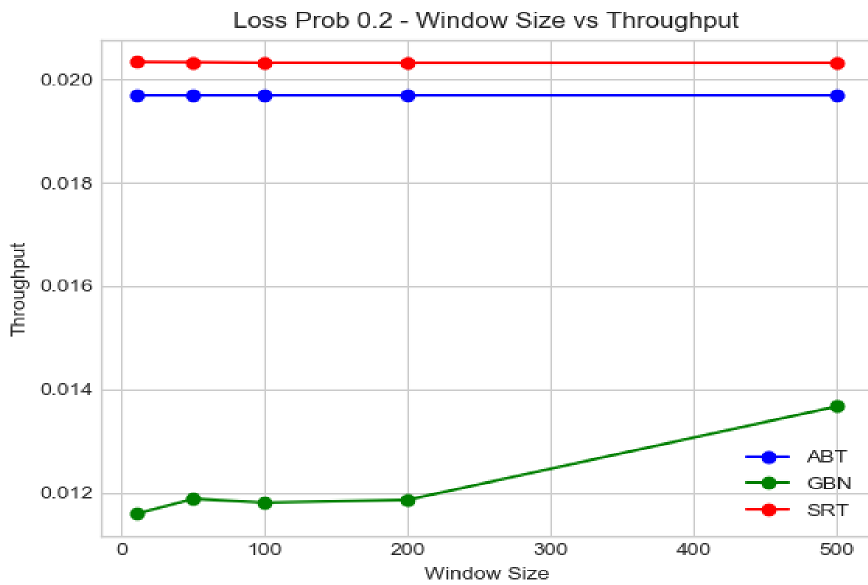
In ideal conditions GBN and SR perform similarly.
SR performs significantly better when the loss is low and GBN has a slightly better performance at higher losses or very bad channel conditions.
As the channel gets bad performance of SR deteriorates steeply when the window size is smaller when compared to larger window size
In bad channel conditions GBN has better throughput with small window sizes.

- Usually, in ideal conditions one would expect that GBN and SR would give much better performance over ABT. Which is not the case in this scenario that could be due to packets arriving less frequently due to which GBN and SR are not using the window size to its capacity.

Experiment 2:

Graph 1: ABR, GBN and SR performances with varying window sizes when loss probability 0.2.

window	loss	ABT	GBN	SR
500	0.2	0.019693	0.013673	0.020337
200	0.2	0.019693	0.011863	0.020332
100	0.2	0.019693	0.011811	0.020321
50	0.2	0.019693	0.011882	0.020320
10	0.2	0.019693	0.01159	0.020320

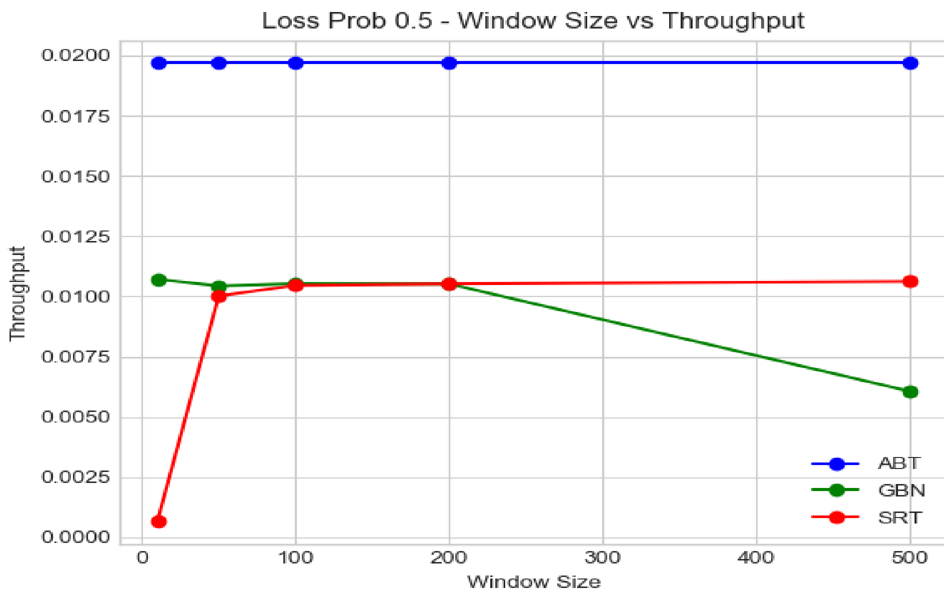


Selective repeat performs similarly at all the window sizes this is because the packets are not arriving as fast for SR to use the window size fully.

Where as GBN throughput increases with the window size.

Graph 2: ABR, GBN and SR performances with varying window sizes when loss probability 0.5.

window	loss	ABT	GBN	SR
500	0.5	0.00959	0.006052	0.010623
200	0.5	0.00959	0.010517	0.010525
100	0.5	0.00959	0.010529	0.010454
50	0.5	0.00959	0.010436	0.010023
10	0.5	0.00959	0.010714	0.000649



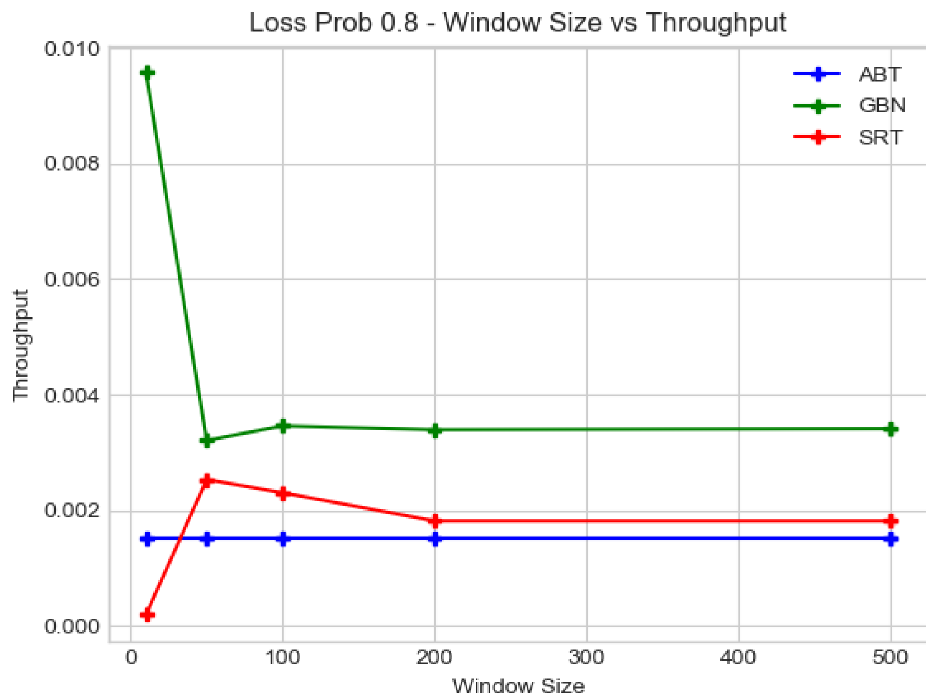
SR performance is low at window size 10, and increases significantly when the window size increases.

GBN performance slightly dips at very high window size.

Loss Probability 0.8:

Graph 3: ABR, GBN and SR performances with varying window sizes when loss probability 0.8.

window	loss	ABT	GBN	SR
500	0.8	0.001527	0.003413	0.001816
200	0.8	0.001527	0.003395	0.001817
100	0.8	0.001527	0.003462	0.002304
50	0.8	0.001527	0.00321	0.002533
10	0.8	0.001527	0.009576	0.000195

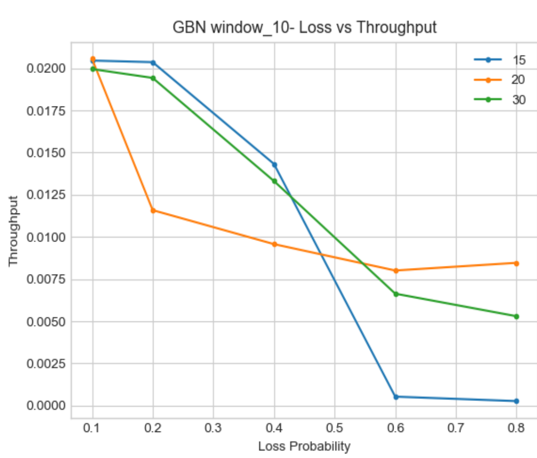


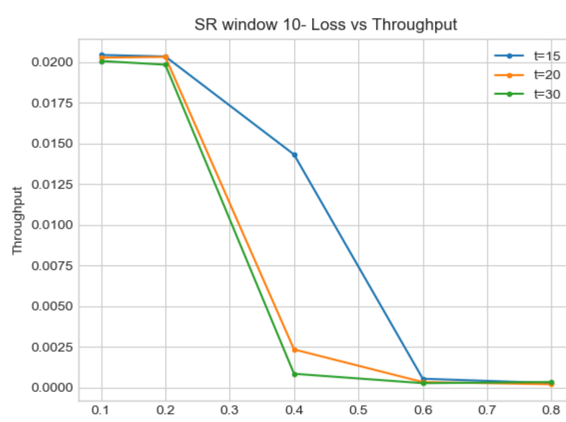
GBN performance is higher at smaller window sizes and falls sharply when the window size increases at higher losses.

SRs performance is doesn't vary much at higher loss and window sizes.

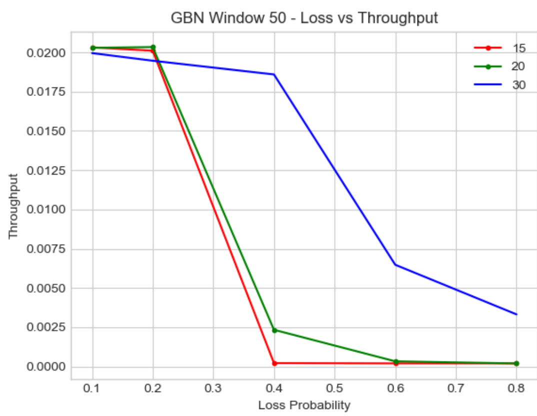
Performance comparison of ABR GBN and SR with different timeout values:

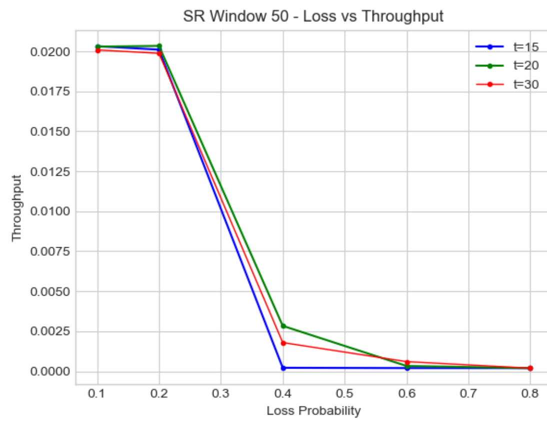
Loss probability vs Throughput variations for SR and GBN with different timeout intervals with window size 10.





Loss probability vs Throughput variations for SR and GBN with different timeout intervals with window size 50.





GBN's performance is better at higher timeout values. This is because when one packet loss occurs GBN retransmit all the remaining packets in the window for which a higher timeout helps. Whereas SR's throughput doesn't vary much with the timeout value.

For all the cases GBN and SR have similar performance under ideal conditions.

SR performs better when the channel has some loss and performance drops significantly when the loss increases. Whereas as GBN performance doesn't vary so drastically with the increase in loss probability.

Under all conditions SRs performance stays similar across all window sizes. Whereas GBNs performance increases when the window size increases.

Overall SR performs better under ideal conditions and when the loss is less among all the protocols. Performance of SR decreases significantly with the increase in loss probability.

GBN has similar performance to SR under ideal conditions and the performance decreases gradually with increase in loss probability, and increases with increase in window size.

ABT has similar performance as GBN and SR that is due to large time between messages and performance of ABT decreases gradually with increase in loss.