PROGRAMMING ASSIGNMENT 2
REPORT

Academic Integrity: I have read and understood the course academic integrity policy.

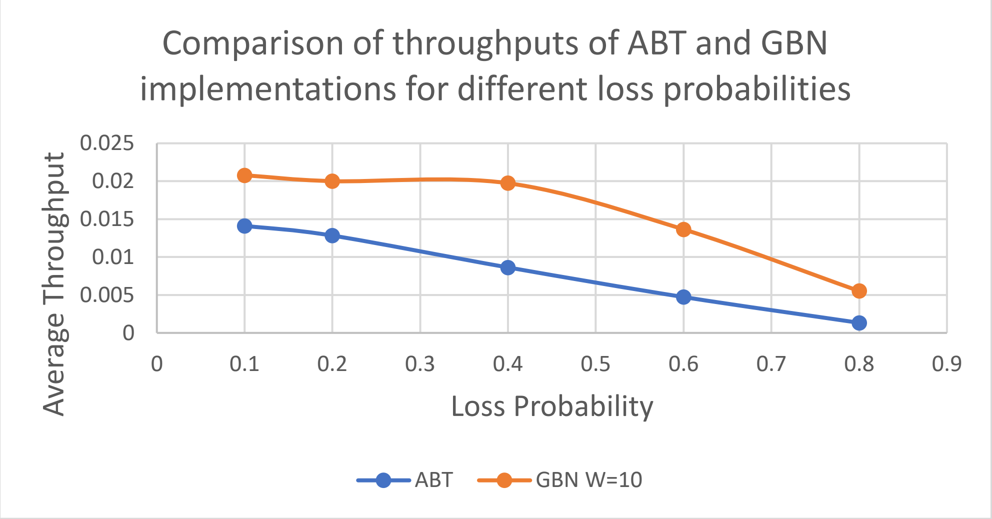
Timeout Values:

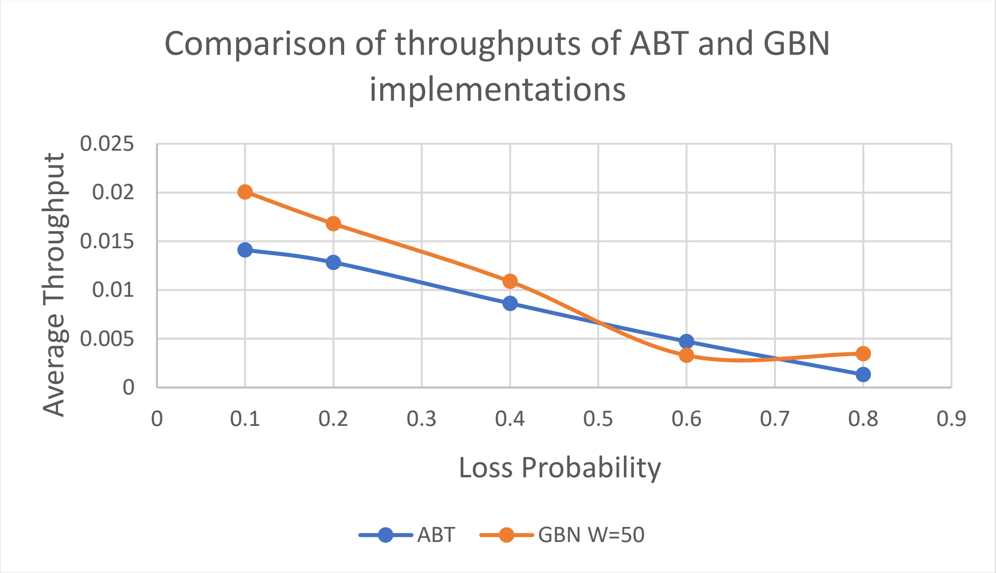
Alternating-Bit-Protocol: According to the assignment description, it takes 5 time units for a packet to travel from the sender to the receiver when no other packet is in the channel. That means the estimated RTT when the channel is empty is 10. So, theoretically one would expect an acknowledgment to reach the sender 10 time units after the packet is sent by it. However, practically, it is possible that packets may take longer than 5 time units to reach the receiver. In such a situation a timeout value of 10 would lead to unnecessary retransmissions. On the other hand, in a bid to avoid unnecessary retransmissions if I set the timeout value to a very high value, e.g. 50, the sender will wait too long for the acknowledgment to arrive before retransmitting the packet. This will lead to a sluggish performance and severely curtail the throughput of the implementation. Therefore, to find a middle ground while dealing with the two aforementioned scenarios I have set the timeout value for the ABT implementation to 20 time units.

Go-Back-N Protocol: In the GBN protocol multiple packets instead of one are sent by the sender on many an occasion, as a result of which there is more congestion in the channel as compared to ABT. This leads to an increase in the estimated RTT. Hence, I have increased the timeout value from 20 to 35 units for my GBN implementation.

Analysis:

Comparison of the Performance of the ABT and GBN Protocols



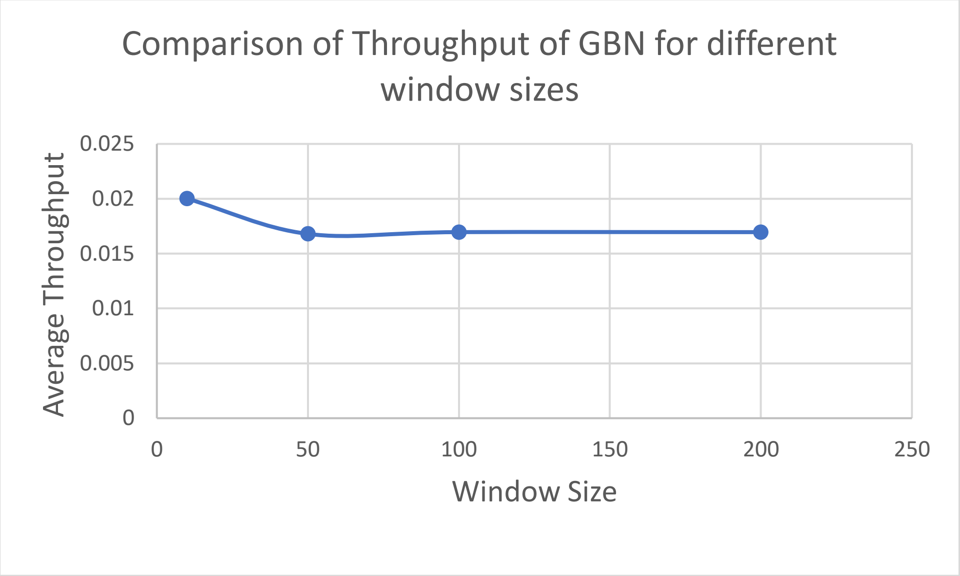


In the above graphs W represents the window size in the GBN implementation.

Observations: From both the graphs it is clear that the throughput of both the protocols decreases as the probability of the packets being lost increases. This is along expected lines. Moreover, the throughput of the GBN implementation is higher than that of ABT for almost all the loss probabilities, as expected(the use of windows leads to better utilisation of the channel bandwidth). However, when the window size is 50, the throughput for the GBN implementation experiences a sharp drop and falls even below the throughput of the ABT implementation for a loss probability of 6 packets on 10. This is unexpected behavior and cannot be explained by just inspecting the average throughput.

Performance of GBN Protocol for different window size

For loss probability of 0.2 :



Observations: From the graph it is clear that the throughput does not increase or decrease monotonically with increasing window sizes. The performance of the protocol cannot be simply judged from the average throughputs for different window sizes. Infact, there are significant variations in the throughput for different seeds for the same window size. Even though the loss probability for all runs is the same, it is possible that in one case the second packet is corrupted and in another, the second to last packet is corrupted. In the first case all but the first packets have to be retransmitted and in the second only the last two packets have to be retransmitted. This leads to a significant change in the throughput.