CSE 489/589
Programming Assignment 2
Reliable Transport Protocols

Submitted by: Hariny Ganapathy (50373513) and Vignesh Waran (50419520)

I have read and understood the course's academic integrity policy.

1. Timeout scheme

- ABT

  We have used timeout as 20, after trial and error we fixed on RTT as 20.

- GBN

  We have used a RTT of 30 after trial and error and found that an RTT of 30 gave the best results.

- SR

  In Selective repeat we have to maintain timer for each packet sent, therefore we need N packet for window of size N. A desired timeout will be more than the one in selective repeat. This is specially due to the retransmission and buffer management techniques used by these protocols.

2. IMPLEMENTATION:

   We are using array of packets, initially we are creating a pointer of packet. We allocate the memory in init method. We use get getwinsize() to get the window size specified and allocate that much memory and use pointer to point to that location. We increment pointer as follows pointer=(pointer+1)%window size A checksum is a small-sized datum derived from a block of digital data for the purpose of detecting errors which may have been introduced during its transmission. We defined a method calc checksum() which calculate check sum for
   the packet.

   struct node
   {

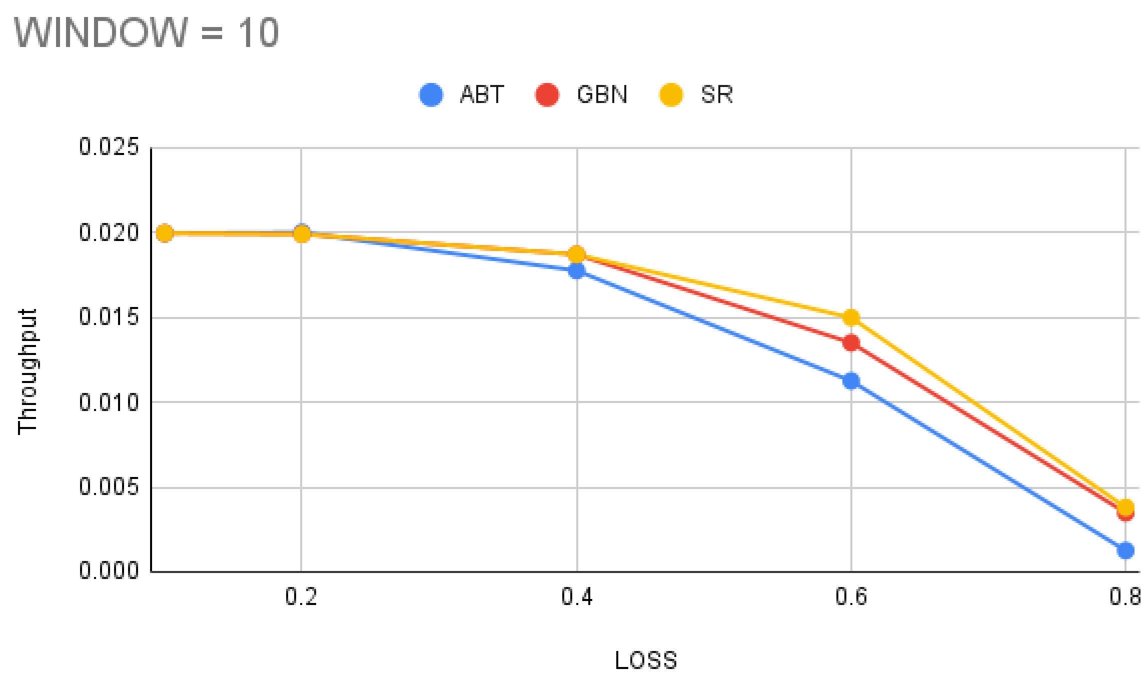
```
struct msg message; //store the message
struct node *next; //point to the next element
};
```

We defined two method to store and retrieve the message from the linked list. We defined append msg(message) to store the message and pop msg() to retrieve the message from the linked list.

3. EXPERIMENTS

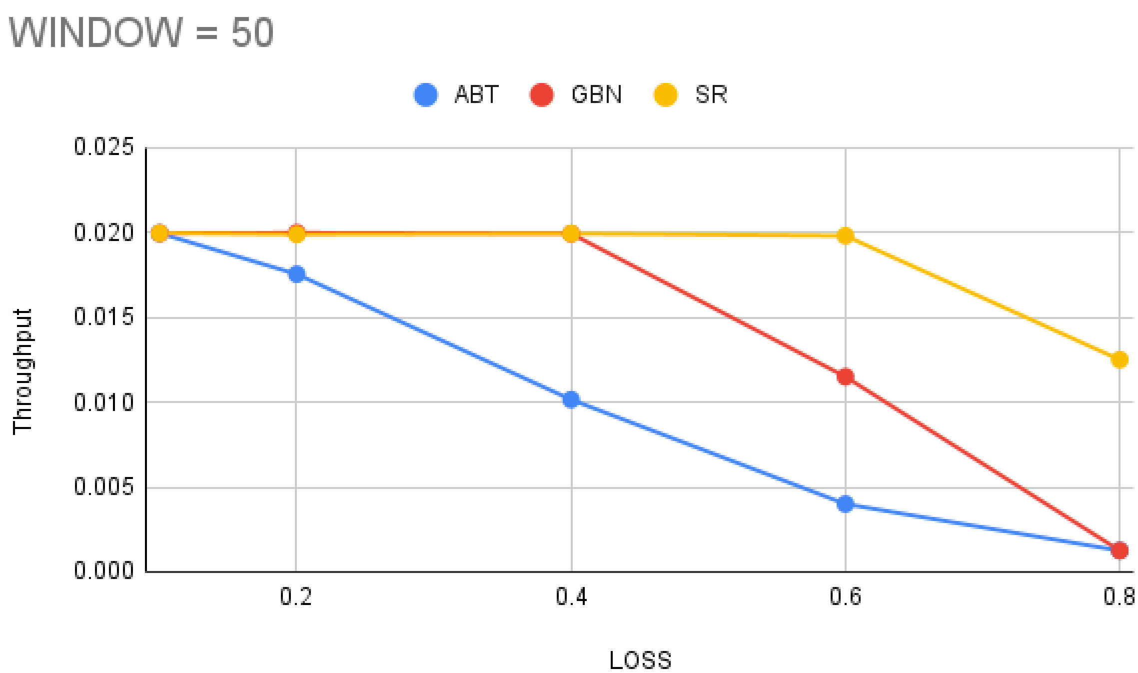
    1. Experiment 1 is conducted on constant values of 1000 messages to be sent by entity A, a corruption probability of 0.2. A mean time of 50 between message arrivals (from A's layer5). The experiments are done on the basis of different loss probabilities of 0.1, 0.2, 0.4, 0.6 and 0.8 and window size of 10 and 50 (for GBN and SR), comparing the 3 protocols (ABT, GBN and SR).

WINDOW SIZE = 10



As the loss probability increases, the throughput of all three protocols drops significantly. Also, the throughput drops pointedly after 0.4 loss probability
As the loss probability increases, less and less packets are able to
make their ways to the receiver side, resulting in the decrease in the throughput.
As for the steadiness of throughput from loss probability from 0.1 to 0.4 and then drastically drops may because the arriving packets number is greater than certain threshold t before 0.4, and throughput is determined by t, however, after 0.4, the arriving packets become less than t , and thus the throughput begins to drop. In the beginning, low loss rate makes gbn’s retransmission redundant, In the end gbn has a better performance since excessive retransmission makes the performance since most of the packets are expected to be lost.

WINDOW = 50

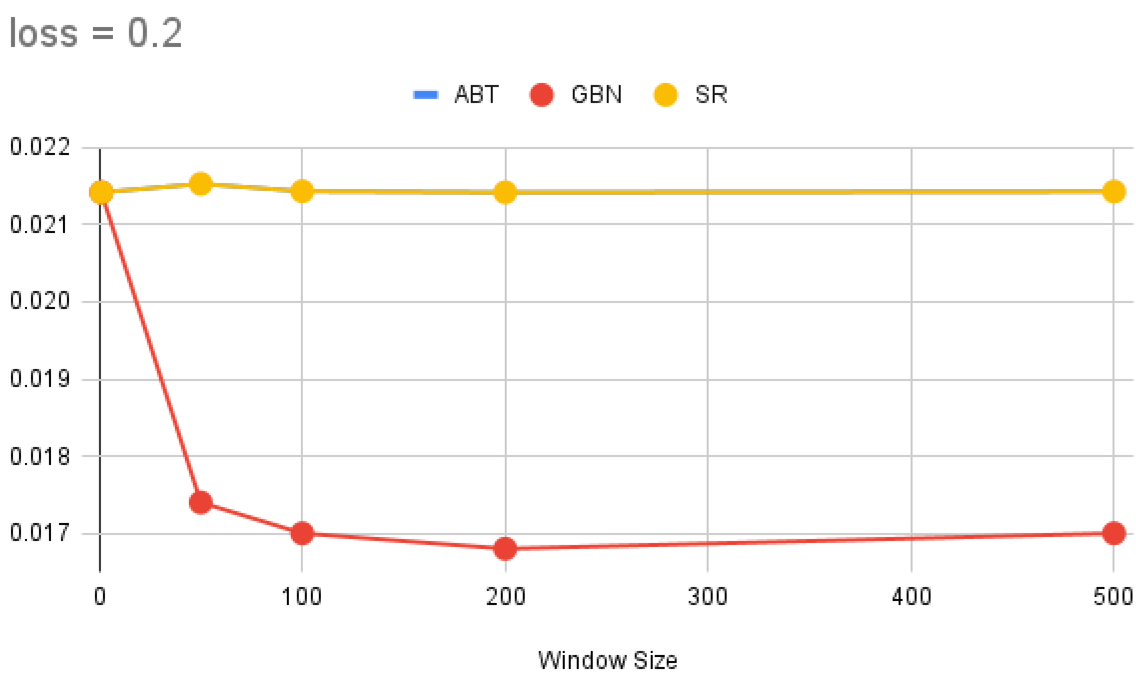


We see that GBN throughput decreases a lot as we increase the loss probability as window size is 50 we have to send lot of packet to B which decreases the throughput. In case of SR it performance remains the same.

2. EXPERIMENT 2

   Experiment 2 is conducted on constant values of 1000 messages sent by entity A, a mean time of 50 between message arrivals (from A's layer5) and a corruption probability of 0.2. The experiments are done on the basis of different window sizes: {10, 50, 100, 200, 500} for GBN and SR and the 3 loss probabilities: {0.2, 0.5, 0.8}, comparing the 3 protocols (ABT, GBN and SR).

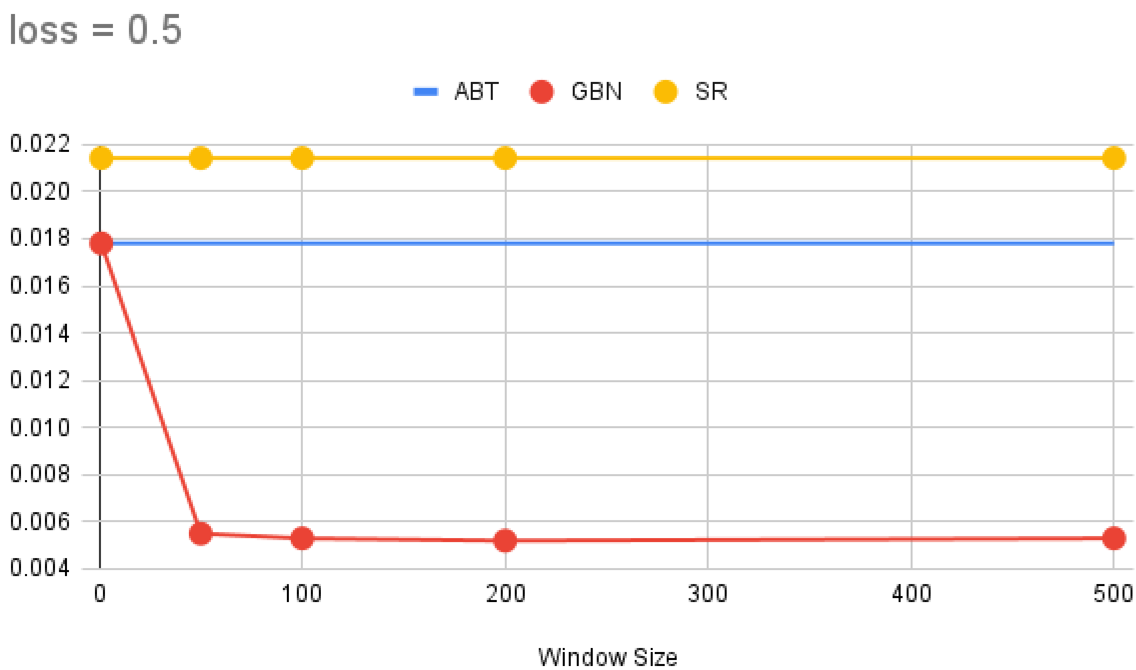
   LOSS PROBABILITY = 0.2



For abt and sr, the throughput stays in a high value constantly, however, for gbn, the throughput is not bad in the beginning, but it begins to drop instantly after the window size starts to expand.

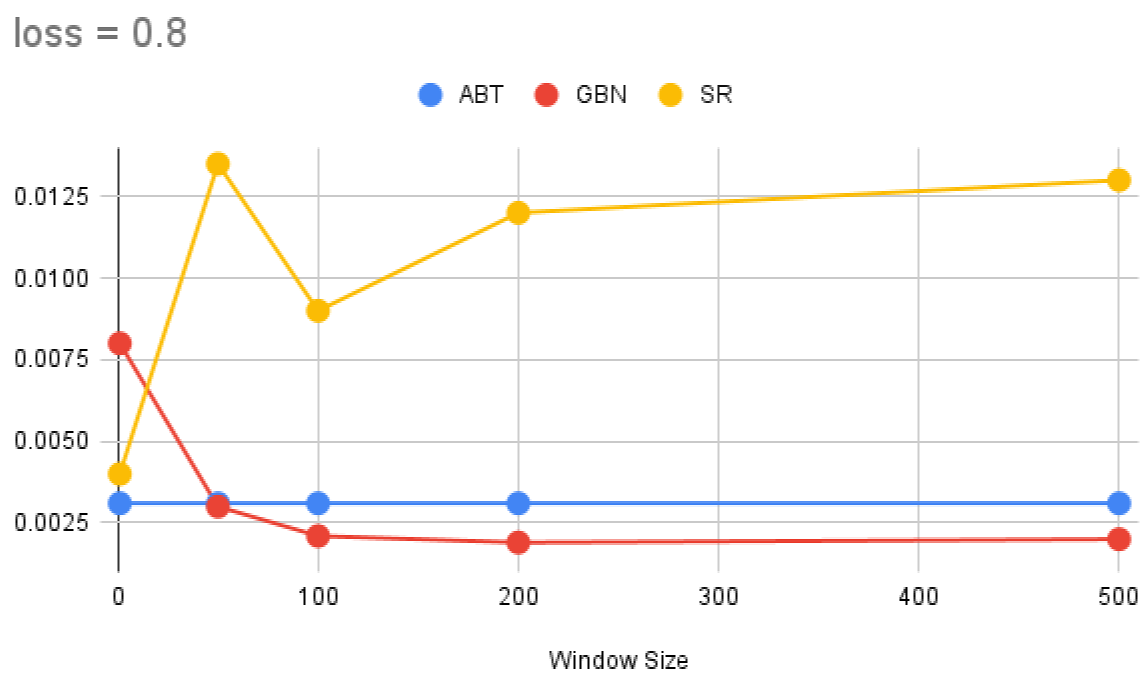
We can see at loss probability 0.2 we have same through put for variation of window sizes, we can conclude that the window sizes at low rate of loss probability window size does not affect the throughput.

LOSS PROBABILITY = 0.5



We can see abt and sr are almost immune to the variance of the window size, however the performance of abt is not as good as sr in all window sizes. As before, gbn once the window size starts to increase, the throughput drops drastically. Though we can see as due to large number of retransmission throughput of the GBN is decreased. For the difference between sr and abt, sr can send multiple packets simultaneously whereas the abt can be stuck in one state if the packet loses during transmission hence sr has a better throughput than abt. Similarly gbn still suffers from excessive retransmission and limit the throughput greatly.

LOSS PROBABLITY = 0.8



We can see at loss probability 0.8 GBN performance decreased dramatically as we are retransmitting same packet multiple times. Though SR initially due to high probability gives low throughput but catches on as we increase the window size, here ADT is not affected due to stop and wait. Since the loss probability is 0.8, which means most of the packets A sends will be lost, so when the window size is small, excessive retransmission like gbn actually improves the performance. The low throughput of abt is because the loss of packets may result in no progress for abt. Gbn, again, suffers from excessive retransmission when the window size is large enough.