PA2 Design

Team member work:

Minghui Yang finished the receiver part of 2 parts, Yinan An finished the sender part of 2 parts.

Timer Operation:

Stop timer means to add

Every time the sender sends a packet, it will start timer, although timer has already existed. And every time the sender receives a uncorrupt packet, it will stop timer. Also, if a packet is retransmitted, the timer will first stop and then start again.

Justification for retransmission timer: A short transmission timeout will cause some packets that are not corrupt and lost but are still on the way to retransmit, and a long transmission timeout will wait for too much time before retransmitting, which will affect the performance. Since it usually takes 5 time units for a packet to transmit from one side to the other, the RTT is 10 time units, so we set our timeout value 30 and 100 during the tests.

Average RTT and average communication time: We keep a map to record the first send time of every packets. And we calculate and record the interval time that a packet was first sent until sender received its ack. We keep a set storing all the packets that has been retransmitted. And if the packet is a retransmitted packet, we will consider it into our average communication time. And if it is not a retransmitted packet, we will consider it into our average communication time as well as average RTT.

SR with cumulative ACK Design

1. General description

Checksum: The checksum is set as the sum of sequence number and ack number, added to a character-by-character sum of payload field.

limited sequence number: The sequence number of SR is set twice as the window size

Tradeoff: We set the limited sequence number and reuse it in order to limiting the length of sequence number size. Also, we could add sack in SR so that the sender could retransmit more unacked packets when receiving duplicate ACK.

sender(A):

When the sender receives a message from layer 5, it will first check if the window is full, and if it is, the sender will buffer the message for sending later, else it will send the packet generated from the message immediately to layer 3.

When the sender receives an ack packet from layer 3 and it is a corrupt packet, the sender will not take any actions and wait for timeout. If the ack packet is duplicate, it will retransmit the first unacked packet, whose sequence number is the window base. And in order to avoid duplicate transmission shown as the Figure1 below, we ignore all the duplicate acks after it has been duplicated for once. And if the ack packet is not duplicate, that is, cumulative, the sender will slide the window and check if there are available messages in the buffer waiting to be sent and send them. When the sender meets time interruption, it will retransmit the first unacknowledged packet in its window.
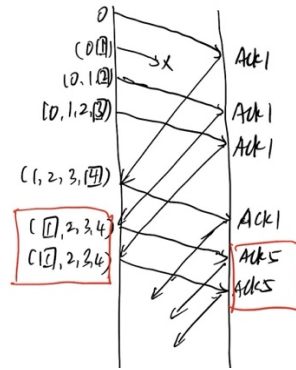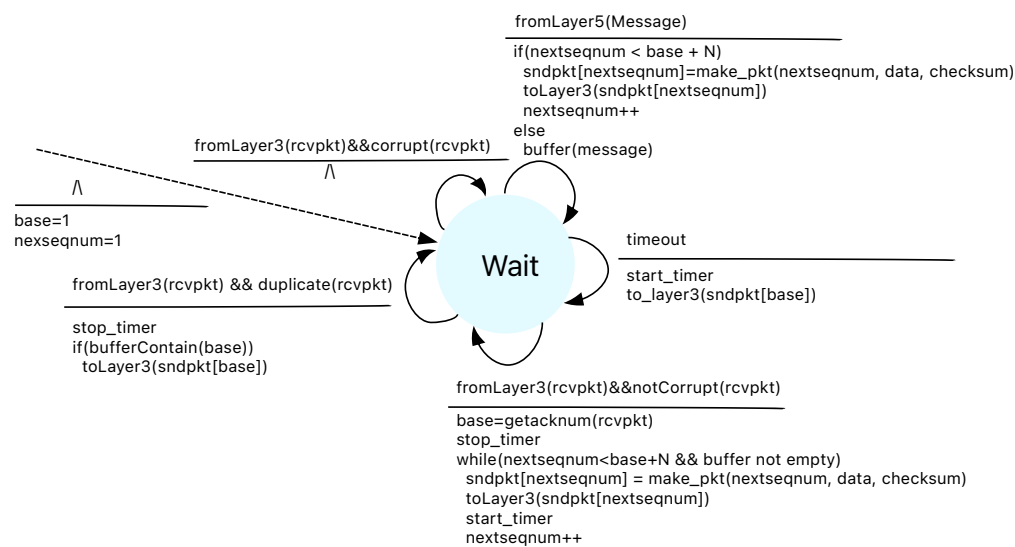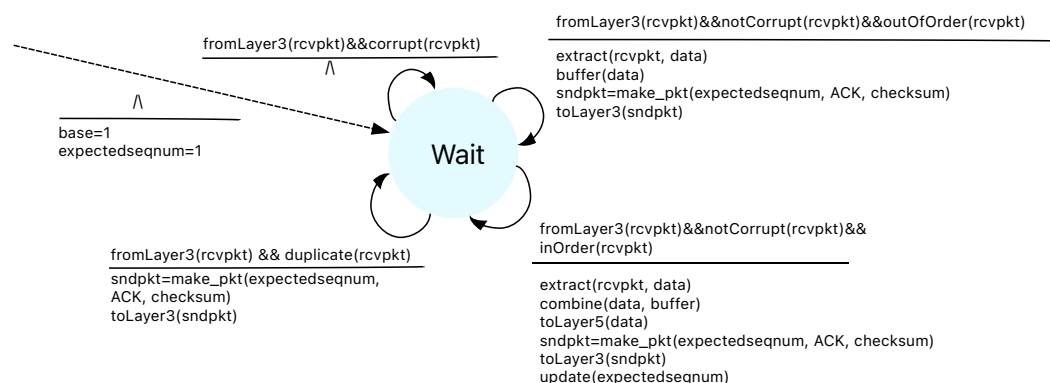
Figure1

Receiver(B):

If the receiver receives a corrupt packet, it will drop it. If it receives an uncorrupt packet, it will send back cumulative ack to the sender. Besides, if the received packet is out of order, the receiver will buffer it for later delivery to layer5, and if the packet is in order, the receiver will check if there's undelivered data in the buffer and deliver all the data to the layer5.

2. FSM
sender:


```
fromLayer5(Message)
if(nextseqnum < base + N)
  sndpkt[nextseqnum]=make_pkt(nextseqnum, data, checksum)
  toLayer3(sndpkt[nextseqnum])
  nextseqnum++
else
  buffer(message)
```

```
fromLayer3(rcvpkt)&&corrupt(rcvpkt)
Λ
```

```
Λ
base=1
nexseqnum=1
```

```
fromLayer3(rcvpkt) && duplicate(rcvpkt)
stop_timer
if(bufferContain(base))
  toLayer3(sndpkt[base])
```

Wait

```
timeout
start_timer
to_layer3(sndpkt[base])
```

```
fromLayer3(rcvpkt)&&notCorrupt(rcvpkt)
base=getacknum(rcvpkt)
stop_timer
while(nextseqnum<base+N && buffer not empty)
  sndpkt[nextseqnum] = make_pkt(nextseqnum, data, checksum)
  toLayer3(sndpkt[nextseqnum])
  start_timer
  nextseqnum++
```

receiver:


```
fromLayer3(rcvpkt)&&notCorrupt(rcvpkt)&&outOfOrder(rcvpkt)
extract(rcvpkt, data)
buffer(data)
sndpkt=make_pkt(expectedseqnum, ACK, checksum)
toLayer3(sndpkt)
```

```
fromLayer3(rcvpkt)&&corrupt(rcvpkt)
Λ
```

```
Λ
base=1
expectedseqnum=1
```

Wait

```
fromLayer3(rcvpkt) && duplicate(rcvpkt)
sndpkt=make_pkt(expectedseqnum,
ACK, checksum)
toLayer3(sndpkt)
```

```
fromLayer3(rcvpkt)&&notCorrupt(rcvpkt)&&
inOrder(rcvpkt)
extract(rcvpkt, data)
combine(data, buffer)
toLayer5(data)
sndpkt=make_pkt(expectedseqnum, ACK, checksum)
toLayer3(sndpkt)
update(expectedseqnum)
```

GBN with SACK option Design

1. General descriptions

Checksum: The checksum is set as the sum of sequence number and ack number, added to a character-by-character sum of payload field, and added to 5 integers in sack field.

limit sequence number: Unlike the traditional GBN whose limit sequence number is windowsize + 1, therefore, the limit sequence number is set as **windowsize+sacksize+1**

sender(A):

When the sender receives a message from layer 5, it will first check if the window is full, and if it is, the sender will buffer the message for sending later, else it will send the packet generated from the message immediately to layer 3.

When the sender receives an ack packet from layer 3 and it is a corrupt packet, the sender will not take any actions. If the ack packet is duplicate, it will retransmit all the unacknowledged packets with a sequence number from window base to the last number of Sack. We don't make it retransmit all the unacknowledged packets in the window in order to avoid infinite duplicate sending problems shown as Figure2. And if the ack packet is not duplicate, that is, cumulative, the sender will slide the window and check if there are available messages in the buffer waiting to be sent and send them. When the sender meets time interruption, it will retransmit the first unacknowledged packet in its window.
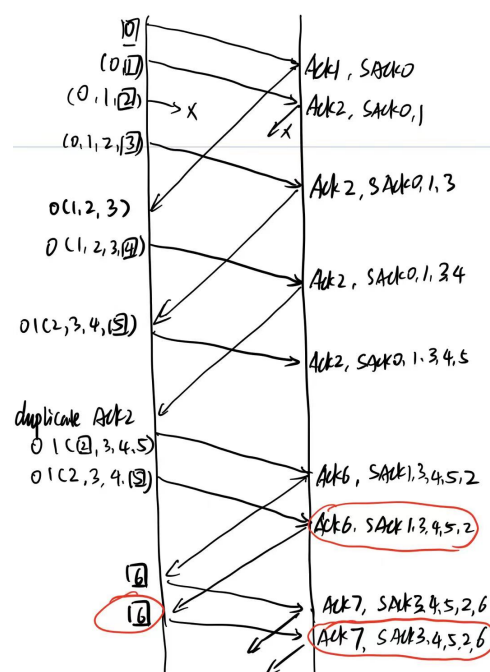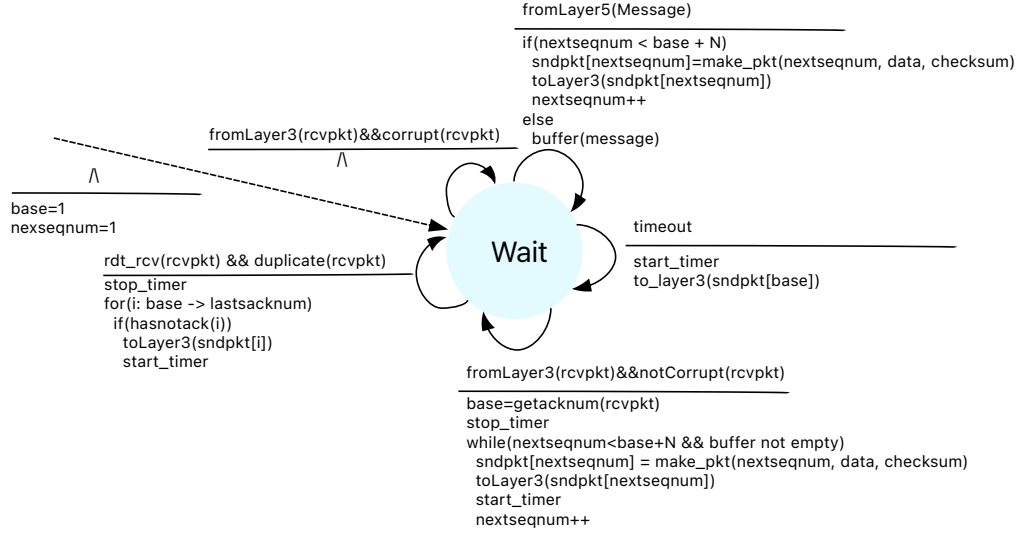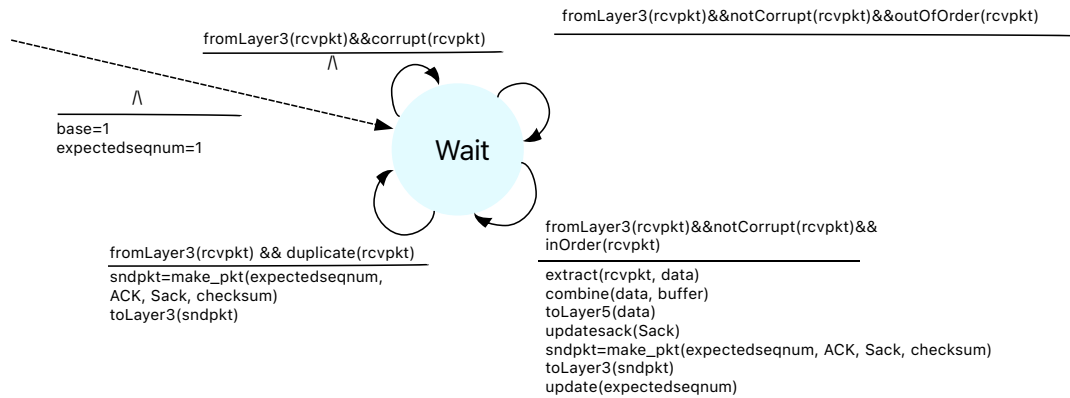

Figure2

receiver(B):

Every time that the receiver receives a packet that is not corrupt, it will update its Sack and next expected sequence number (set as ack) accordingly, and then send the ack packets to the sender. If the received packet is out of order, the receiver will buffer it for later delivery to layer5. And if the packet is in order, the receiver will check if there's undelivered data in the buffer and deliver all the data to the layer5.

2. FSM

sender:


```
                                              fromLayer5(Message)
                                              _____
                                              if(nextseqnum < base + N)
                                                sndpkt[nextseqnum]=make_pkt(nextseqnum, data, checksum)
                                                toLayer3(sndpkt[nextseqnum])
                                                nextseqnum++
                                              else
                                                buffer(message)

        fromLayer3(rcvpkt)&&corrupt(rcvpkt)
        _____
                        Λ

        Λ
    _____
    base=1                                                          timeout
    nexseqnum=1                              Wait                    _____
                                                                    start_timer
    rdt_rcv(rcvpkt) && duplicate(rcvpkt)                            to_layer3(sndpkt[base])
    _____
    stop_timer
    for(i: base -> lastsacknum)
      if(hasnotack(i))
        toLayer3(sndpkt[i])
        start_timer

                        fromLayer3(rcvpkt)&&notCorrupt(rcvpkt)
                        _____
                        base=getacknum(rcvpkt)
                        stop_timer
                        while(nextseqnum<base+N && buffer not empty)
                          sndpkt[nextseqnum] = make_pkt(nextseqnum, data, checksum)
                          toLayer3(sndpkt[nextseqnum])
                          start_timer
                          nextseqnum++
```

receiver:

```
                                    fromLayer3(rcvpkt)&&notCorrupt(rcvpkt)&&outOfOrder(rcvpkt)
                                    _____
        fromLayer3(rcvpkt)&&corrupt(rcvpkt)
        _____
                        Λ

        Λ
    _____
    base=1                              Wait
    expectedseqnum=1

        fromLayer3(rcvpkt) && duplicate(rcvpkt)         fromLayer3(rcvpkt)&&notCorrupt(rcvpkt)&&
        _____         inOrder(rcvpkt)
        sndpkt=make_pkt(expectedseqnum,                 _____
        ACK, Sack, checksum)                            extract(rcvpkt, data)
        toLayer3(sndpkt)                                combine(data, buffer)
                                                        toLayer5(data)
                                                        updatesack(Sack)
                                                        sndpkt=make_pkt(expectedseqnum, ACK, Sack, checksum)
                                                        toLayer3(sndpkt)
                                                        update(expectedseqnum)
```

Result Analysis

Confidence intervals:

We use *ThreadLocalRandom.current().nextInt()* to generate 100 different seeds for every combination of paramaters, and calculate average value of average communication time, we then compute the 90% confidence intervals, they are shown as below.

① The average value and confidence intervals of average communication time of 100 runs given different loss rate in SR.

M	N	O	P	Q
loss rate	aver commu time	stdev commu time	lower	upper
0.05	14.05659763	0.311503067	13.99554303	14.11765223
0.1	17.55323047	0.493558564	17.45649299	17.64996795
0.15	21.93365319	0.748649036	21.78691798	22.08038841
0.2	27.25989822	0.108012345	27.2387278	27.28106864
0.25	33.88254732	1.450761336	33.5981981	34.16689654
0.3	43.0676588	1.906998004	42.69388719	43.4414304
0.35	55.61293635	3.851617733	54.85801928	56.36785343
0.4	74.95345013	5.801319575	73.81639149	76.09050876
0.45	108.3638419	11.38205102	106.1329599	110.5947239
0.5	170.9535894	22.00529261	166.640552	175.2666267

② The average value and confidence intervals of average communication time of 100 runs given different corruption rate in SR.

P	Q	R	S	T
corrupt rate	aver commu tim	stdev commu time	lower	upper
0.05	14.09095008	0.351908958	14.0219759	14.1599242
0.1	17.68000831	0.575909385	17.5671301	17.7928866
0.15	22.07769992	0.964201483	21.8887164	22.2666834
0.2	27.25210573	1.114331724	27.0336967	27.4705148
0.25	34.0000009	1.523270879	33.7014398	34.298562
0.3	42.95160351	2.471561351	42.4671775	43.4360295
0.35	55.75459952	3.989470161	54.9726634	56.5365357
0.4	75.28919792	5.687625052	74.1744234	76.4039724
0.45	108.0092134	11.49017762	105.757139	110.261288
0.5	173.8330147	24.62774824	169.005976	178.660053

③ The average value and confidence intervals of average communication time of 100 runs given different loss rate in GBN.

P	Q	R	S	T
loss rate	aver commu	stdev commu	lower	upper
0.05	14.1301274	0.35728272	14.0601	14.2001548
0.1	17.6950996	0.46735913	17.6034972	17.786702
0.15	22.0061131	0.77885705	21.8534571	22.1587691
0.2	27.4136634	1.08827382	27.2003617	27.6269651
0.25	33.979664	1.43840586	33.6977365	34.2615916
0.3	42.8235119	2.08190449	42.4154586	43.2315652
0.35	54.5228923	3.30919935	53.8742893	55.1714954
0.4	73.1061137	6.19954681	71.8910025	74.3212249
0.45	107.723675	13.6216833	105.053825	110.393525
0.5	185.571387	35.5030325	178.612792	192.529981

④ The average value and confidence intervals of average communication time of 100 runs given different corruption rate in GBN

Q	R	S	T	U
corrrupt rate	aver commu	stdev commu	lower	upper
0.05	14.0872649	0.31858022	14.0248232	14.1497067
0.1	17.5992527	0.47518612	17.5061162	17.6923892
0.15	21.9810861	0.72751908	21.8384923	22.1236798
0.2	27.3218387	1.12050294	27.1022201	27.5414573
0.25	34.2226675	1.66391973	33.8965392	34.5487957
0.3	43.2278246	2.42102396	42.7533039	43.7023453
0.35	55.1577296	3.86422176	54.4003422	55.9151171
0.4	73.3795535	7.31415537	71.9459791	74.813128
0.45	106.769629	15.1811759	103.794119	109.74514
0.5	187.851459	38.7419988	180.258027	195.444891

Figure and Function (SR)

As the loss/corrupt rate becomes higher, the average communication time will increase, because higher loss/corrupt rate will cause more packets retransmission.
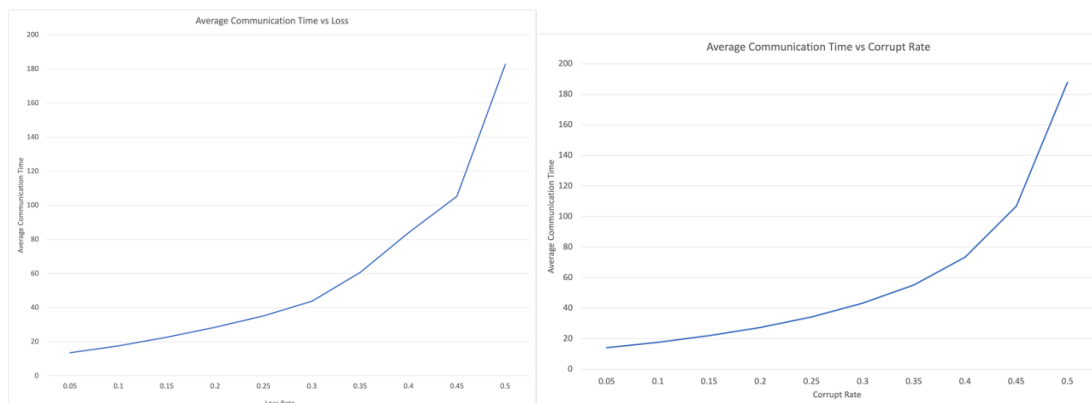
As the loss/corrupt rate becomes higher, the average RTT will increase, because higher loss/corrupt rate will cause more packets retransmission.
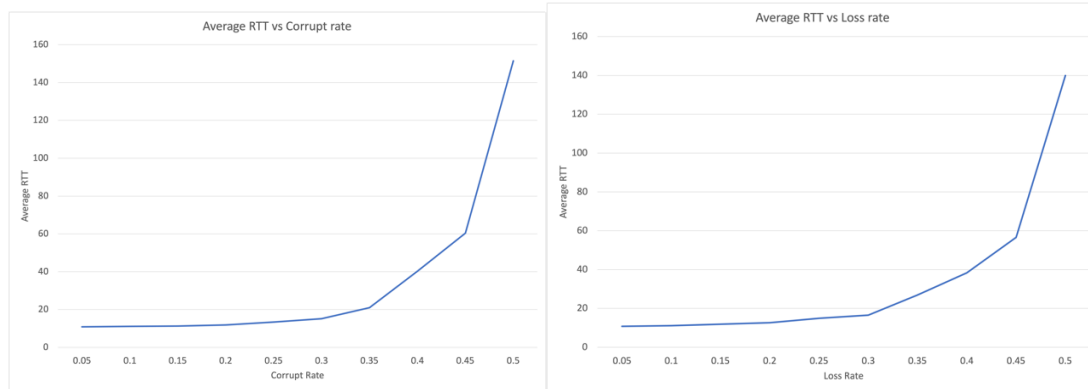
When the loss/corruption rate increase, the retransmission time will increase.
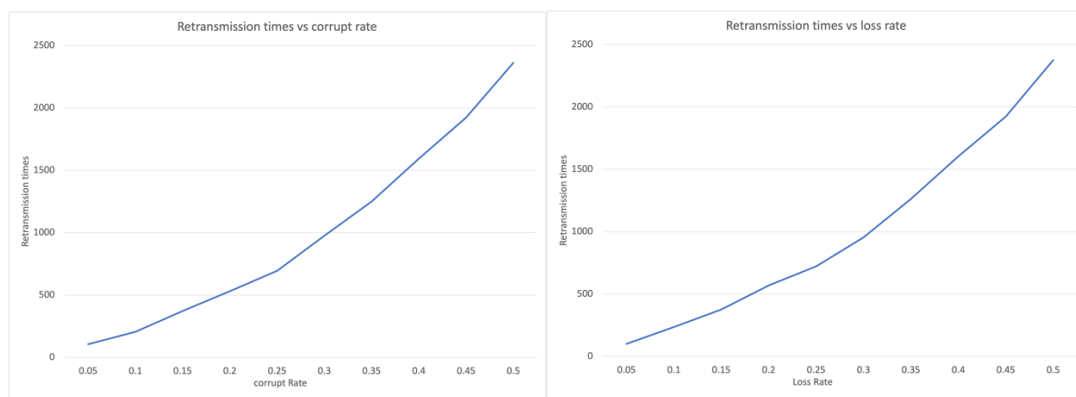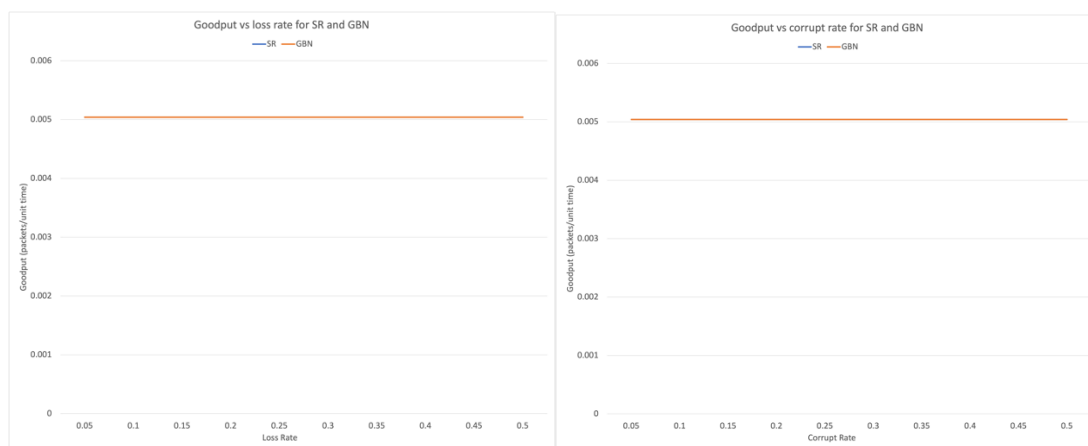

Figure and Function (GBN)

Compare GBN with SR, the goodput maintains the same although changing loss and corrupt rate. That's because the total time is a fix value, and the total origin packets for A is also a fix value.

Compare GBN with SR, the throughput will increase for GBN with Sack if the loss/corruption rate is high, that's because GBN will retransmit more unacked packets when receiving duplicate ACK than SR.

Throughput vs corrupt rate for SR and GBN

Throughput vs loss rate for SR and GBN
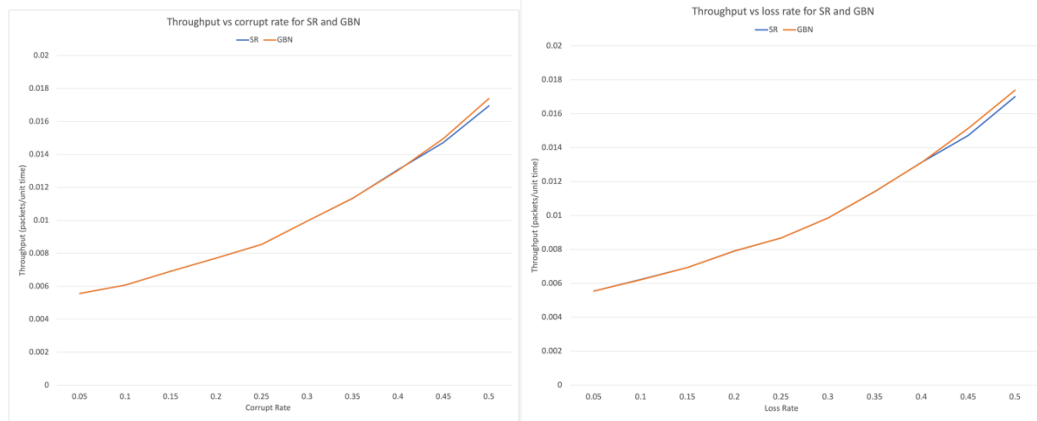
The average packet delay will be higher for GBN than SR if the corruption/loss rate is high. Because the number of retransmitted packets increase, it will cause the packet delay increasing.

Average packet delay vs corrupt rate for SR and GBN

Average packet delay vs loss rate for SR and GBN
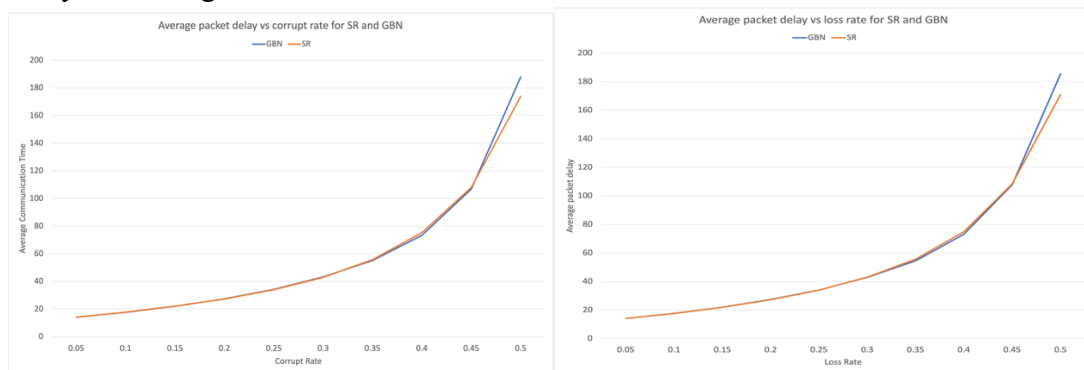
Output Trace (SR with Cumulative ACK)

Case 1: no loss no corruption

-- * Network Simulator v1.0 * --

Enter number of messages to simulate (> 0): [10] 1000

Enter packet loss probability (0.0 for no loss): [0.0] 0.0

Enter packet corruption probability (0.0 for no corruption): [0.0] 0.0

Enter average time between messages from sender's layer 5 (> 0.0): [1000] 200

Enter window size (> 0): [8] 8

Enter retransmission timeout (>0.0) [15.0] 30

Enter trace level (>= 0): [0] 3

Enter random seed: [0] 1234

generateNextArrival(): called

generateNextArrival(): time is 0.0

generateNextArrival(): future time for event 1 at entity 0 will be 43.4339426862392

EVENT time: 43.4339426862392 type: 1 entity: 0

generateNextArrival(): called

generateNextArrival(): time is 43.4339426862392

generateNextArrival(): future time for event 1 at entity 0 will be 67.36045411542099

toLayer3: seqnum: 0 acknum: -1 checksum: 1939 payload: aaaaaaaaaaaaaaaaaaaa

toLayer3: scheduling arrival on other side

startTimer: starting timer at 43.4339426862392

EVENT time: 45.66060365611743 type: 2 entity: 1

toLayer3: seqnum: 0 acknum: 1 checksum: 1 payload:

toLayer3: scheduling arrival on other side

EVENT time: 47.569475386822106 type: 2 entity: 0

stopTimer: stopping timer at 47.569475386822106

EVENT time: 67.36045411542099 type: 1 entity: 0

generateNextArrival(): called

generateNextArrival(): time is 67.36045411542099

generateNextArrival(): future time for event 1 at entity 0 will be 83.31553086947233

toLayer3: seqnum: 1 acknum: -1 checksum: 1960 payload: bbbbbbbbbbbbbbbbbbbb

toLayer3: scheduling arrival on other side

startTimer: starting timer at 67.36045411542099

EVENT time: 68.78865932187871 type: 2 entity: 1

toLayer3: seqnum: 0 acknum: 2 checksum: 2 payload:

toLayer3: scheduling arrival on other side

EVENT time: 70.75603503524887 type: 2 entity: 0

stopTimer: stopping timer at 70.75603503524887

EVENT time: 83.31553086947233 type: 1 entity: 0
generateNextArrival(): called
generateNextArrival(): time is 83.31553086947233
generateNextArrival(): future time for event 1 at entity 0 will be 238.65474410229803
toLayer3: seqnum: 2 acknum: -1 checksum: 1981 payload: cccccccccccccccccccc
toLayer3: scheduling arrival on other side
startTimer: starting timer at 83.31553086947233

EVENT time: 87.99688711203343 type: 2 entity: 1
toLayer3: seqnum: 0 acknum: 3 checksum: 3 payload:
toLayer3: scheduling arrival on other side

EVENT time: 91.66637165440838 type: 2 entity: 0
stopTimer: stopping timer at 91.66637165440838

EVENT time: 238.65474410229803 type: 1 entity: 0
generateNextArrival(): called
generateNextArrival(): time is 238.65474410229803
generateNextArrival(): future time for event 1 at entity 0 will be 529.581591235084
toLayer3: seqnum: 3 acknum: -1 checksum: 2002 payload: dddddddddddddddddddd
toLayer3: scheduling arrival on other side
startTimer: starting timer at 238.65474410229803

EVENT time: 241.6328928495132 type: 2 entity: 1
toLayer3: seqnum: 0 acknum: 4 checksum: 4 payload:
toLayer3: scheduling arrival on other side

EVENT time: 249.96951200903345 type: 2 entity: 0
stopTimer: stopping timer at 249.96951200903345
……
==============STATISTICS====================
Number of original packets transmitted by A:1000
Number of retransmissions by A:0
Number of data packets delivered to layer 5 at B:1000
Number of ACK packets sent by B:1000
Number of corrupted packets:0
Ratio of lost packets:0.0
Ratio of corrupted packets:0.0
Average RTT:10.76163435773795
Average communication time:10.76163435773795
===

Case 2: ack lost, a later accumulative ack moves the sender window by more than 1

-- * Network Simulator v1.0 * --

Enter number of messages to simulate (> 0): [10] 1000

Enter packet loss probability (0.0 for no loss): [0.0] 0.3

Enter packet corruption probability (0.0 for no corruption): [0.0] 0.0

Enter average time between messages from sender's layer 5 (> 0.0): [1000] 200

Enter window size (> 0): [8] 8

Enter retransmission timeout (>0.0) [15.0] 30

Enter trace level (>= 0): [0] 3

Enter random seed: [0] 1234

generateNextArrival(): called

generateNextArrival(): time is 0.0

generateNextArrival(): future time for event 1 at entity 0 will be 43.4339426862392

......

EVENT time: 1627.3689992980742 type: 1 entity: 0

generateNextArrival(): called

generateNextArrival(): time is 1627.3689992980742

generateNextArrival(): future time for event 1 at entity 0 will be 1801.1291848506119

toLayer3: seqnum: 8 acknum: -1 checksum: 2107 payload: iiiiiiiiiiiiiiiiiii

toLayer3: scheduling arrival on other side

startTimer: starting timer at 1627.3689992980742

EVENT time: 1631.3911557359538 type: 2 entity: 1

toLayer3: seqnum: 0 acknum: 9 checksum: 9 payload:

toLayer3: scheduling arrival on other side

EVENT time: 1639.8258308664203 type: 2 entity: 0

stopTimer: stopping timer at 1639.8258308664203

EVENT time: 1801.1291848506119 type: 1 entity: 0

generateNextArrival(): called

generateNextArrival(): time is 1801.1291848506119

generateNextArrival(): future time for event 1 at entity 0 will be 1866.9148838770714

toLayer3: seqnum: 9 acknum: -1 checksum: 2128 payload: jjjjjjjjjjjjjjjjjjjj

toLayer3: packet being lost

startTimer: starting timer at 1801.1291848506119

EVENT time: 1831.1291848506119 type: 0 entity: 0

toLayer3: seqnum: 9 acknum: -1 checksum: 2128 payload: jjjjjjjjjjjjjjjjjjjj

toLayer3: packet being lost

stopTimer: stopping timer at 1831.1291848506119

stopTimer: Warning: Unable to cancel your timer

startTimer: starting timer at 1831.1291848506119

EVENT time: 1861.1291848506119 type: 0 entity: 0
toLayer3: seqnum: 9 acknum: -1 checksum: 2128 payload: jjjjjjjjjjjjjjjjjjjjj
toLayer3: scheduling arrival on other side
stopTimer: stopping timer at 1861.1291848506119
stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 1861.1291848506119

EVENT time: 1866.9148838770714 type: 1 entity: 0
generateNextArrival(): called
generateNextArrival(): time is 1866.9148838770714
generateNextArrival(): future time for event 1 at entity 0 will be 2218.2317575609604
toLayer3: seqnum: 10 acknum: -1 checksum: 2149 payload: kkkkkkkkkkkkkkkkkkkkk
toLayer3: scheduling arrival on other side
startTimer: starting timer at 1866.9148838770714
startTimer: Warning: Attempting to start a timer that is already running

EVENT time: 1868.514908291879 type: 2 entity: 1
toLayer3: seqnum: 0 acknum: 10 checksum: 10 payload:
toLayer3: packet being lost

EVENT time: 1875.67210913419 type: 2 entity: 1
toLayer3: seqnum: 0 acknum: 11 checksum: 11 payload:
toLayer3: scheduling arrival on other side
....
===============STATISTICS=====================
Number of original packets transmitted by A:1000
Number of retransmissions by A:954
Number of data packets delivered to layer 5 at B:1000
Number of ACK packets sent by B:1354
Number of corrupted packets:0
Ratio of lost packets:0.2883917775090689
Ratio of corrupted packets:0.0
Average RTT:16.469906417455793
Average communication time:43.76293848558422
==
ack corrupted, a later accumulative ack moves the sender window by more than 1
-- * Network Simulator v1.0 * --
Enter number of messages to simulate (> 0): [10] 1000
Enter packet loss probability (0.0 for no loss): [0.0] 0.0
Enter packet corruption probability (0.0 for no corruption): [0.0] 0.3

Enter average time between messages from sender's layer 5 (> 0.0): [1000] 200
Enter window size (> 0): [8] 8
Enter retransmission timeout (>0.0) [15.0] 30
Enter trace level (>= 0): [0] 3
Enter random seed: [0] 1234

......
EVENT time: 8191.9804071169165 type: 2 entity: 1
toLayer3: seqnum: 0 acknum: 10 checksum: 10 payload:
toLayer3: scheduling arrival on other side

EVENT time: 8197.87002166814 type: 2 entity: 0
stopTimer: stopping timer at 8197.87002166814

EVENT time: 8342.442091128269 type: 1 entity: 0
generateNextArrival(): called
generateNextArrival(): time is 8342.442091128269
generateNextArrival(): future time for event 1 at entity 0 will be 8360.093997009186
toLayer3: seqnum: 10 acknum: -1 checksum: 2269 payload: qqqqqqqqqqqqqqqqqqqq
toLayer3: packet being corrupted
toLayer3: scheduling arrival on other side
startTimer: starting timer at 8342.442091128269

EVENT time: 8351.139622180852 type: 2 entity: 1

EVENT time: 8360.093997009186 type: 1 entity: 0
generateNextArrival(): called
generateNextArrival(): time is 8360.093997009186
generateNextArrival(): future time for event 1 at entity 0 will be 8524.21033356731
toLayer3: seqnum: 11 acknum: -1 checksum: 2290 payload: rrrrrrrrrrrrrrrrrrrr
toLayer3: scheduling arrival on other side
startTimer: starting timer at 8360.093997009186
startTimer: Warning: Attempting to start a timer that is already running

EVENT time: 8368.97665334025 type: 2 entity: 1
toLayer3: seqnum: 0 acknum: 10 checksum: 10 payload:
toLayer3: packet being corrupted
toLayer3: scheduling arrival on other side

EVENT time: 8371.956999420148 type: 2 entity: 0

EVENT time: 8372.442091128269 type: 0 entity: 0
toLayer3: seqnum: 10 acknum: -1 checksum: 2269 payload: qqqqqqqqqqqqqqqqqqqq
toLayer3: scheduling arrival on other side
stopTimer: stopping timer at 8372.442091128269

stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 8372.442091128269

EVENT time: 8378.243812372202 type: 2 entity: 1
toLayer3: seqnum: 0 acknum: 12 checksum: 12 payload:
toLayer3: packet being corrupted
toLayer3: scheduling arrival on other side

EVENT time: 8380.824243903926 type: 2 entity: 0

EVENT time: 8402.442091128269 type: 0 entity: 0
toLayer3: seqnum: 10 acknum: -1 checksum: 2269 payload: qqqqqqqqqqqqqqqqqqqq
toLayer3: scheduling arrival on other side
stopTimer: stopping timer at 8402.442091128269
stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 8402.442091128269

EVENT time: 8405.838831751456 type: 2 entity: 1
toLayer3: seqnum: 0 acknum: 12 checksum: 12 payload:
toLayer3: scheduling arrival on other side

EVENT time: 8412.359080782493 type: 2 entity: 0
stopTimer: stopping timer at 8412.359080782493
.....
==============STATISTICS=====================
Number of original packets transmitted by A:1000
Number of retransmissions by A:975
Number of data packets delivered to layer 5 at B:1000
Number of ACK packets sent by B:1368
Number of corrupted packets:1019
Ratio of lost packets:-0.013161830690996112
Ratio of corrupted packets:0.30085621493947445
Average RTT:15.126216195774944
Average communication time:42.68302154603352
===
Case 3: data packet corrupted, sender retransmit data after RTO
-- * Network Simulator v1.0 * --
Enter number of messages to simulate (> 0): [10] 1000
Enter packet loss probability (0.0 for no loss): [0.0] 0.0
Enter packet corruption probability (0.0 for no corruption): [0.0] 0.3
Enter average time between messages from sender's layer 5 (> 0.0): [1000] 200
Enter window size (> 0): [8] 8
Enter retransmission timeout (>0.0) [15.0] 30
Enter trace level (>= 0): [0] 3

Enter random seed: [0] 1234

……

EVENT time: 58440.33326212347 type: 1 entity: 0

generateNextArrival(): called

generateNextArrival(): time is 58440.33326212347

generateNextArrival(): future time for event 1 at entity 0 will be 58546.18366039002

Sequence 13 first send time: 58440.33326212347

toLayer3: seqnum: 13 acknum: -1 checksum: 2252 payload: ppppppppppppppppppppp

toLayer3: packet being corrupted

toLayer3: scheduling arrival on other side

startTimer: starting timer at 58440.33326212347

EVENT time: 58445.877346110414 type: 2 entity: 1

 B received packet

seqnum: 13 acknum: -1 checksum: 2252 payload: ?ppppppppppppppppppppp

EVENT time: 58470.33326212347 type: 0 entity: 0

A timeout, retransmit

toLayer3: seqnum: 13 acknum: -1 checksum: 2252 payload: ppppppppppppppppppppp

toLayer3: scheduling arrival on other side

startTimer: starting timer at 58470.33326212347

EVENT time: 58474.00659199804 type: 2 entity: 1

 B received packet

seqnum: 13 acknum: -1 checksum: 2252 payload: ppppppppppppppppppppp

toLayer3: seqnum: 0 acknum: 14 checksum: 14 payload:

toLayer3: packet being corrupted

toLayer3: scheduling arrival on other side

……

==============STATISTICS====================

Number of original packets transmitted by A:1000

Number of retransmissions by A:975

Number of data packets delivered to layer 5 at B:1000

Number of ACK packets sent by B:1368

Number of corrupted packets:1019

Ratio of lost packets:-0.013161830690996112

Ratio of corrupted packets:0.30085621493947445

Average RTT:15.126216195774944

Average communication time:42.68302154603352

===

Case 4: ack lost, sender retransmitted after receiving duplicate ack

-- * Network Simulator v1.0 * --

Enter number of messages to simulate (> 0): [10] 1000
Enter packet loss probability (0.0 for no loss): [0.0] 0.3
Enter packet corruption probability (0.0 for no corruption): [0.0] 0.0
Enter average time between messages from sender's layer 5 (> 0.0): [1000] 200
Enter window size (> 0): [8] 8
Enter retransmission timeout (>0.0) [15.0] 30
Enter trace level (>= 0): [0] 3
Enter random seed: [0] 1234

.....

EVENT time: 11350.759056367686 type: 2 entity: 1
toLayer3: seqnum: 0 acknum: 13 checksum: 13 payload:
toLayer3: packet being lost

EVENT time: 11371.789605395672 type: 0 entity: 0
toLayer3: seqnum: 12 acknum: -1 checksum: 2111 payload: iiiiiiiiiiiiiiiiiii
toLayer3: scheduling arrival on other side
stopTimer: stopping timer at 11371.789605395672
stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 11371.789605395672

EVENT time: 11377.732230597858 type: 2 entity: 1
toLayer3: seqnum: 0 acknum: 13 checksum: 13 payload:
toLayer3: packet being lost

EVENT time: 11401.789605395672 type: 0 entity: 0
toLayer3: seqnum: 12 acknum: -1 checksum: 2111 payload: iiiiiiiiiiiiiiiiiii
toLayer3: scheduling arrival on other side
stopTimer: stopping timer at 11401.789605395672
stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 11401.789605395672

EVENT time: 11408.814142277291 type: 2 entity: 1
toLayer3: seqnum: 0 acknum: 13 checksum: 13 payload:
toLayer3: scheduling arrival on other side

EVENT time: 11410.338480788598 type: 2 entity: 0
stopTimer: stopping timer at 11410.338480788598

EVENT time: 11524.05774101993 type: 1 entity: 0
generateNextArrival(): called
generateNextArrival(): time is 11524.05774101993
generateNextArrival(): future time for event 1 at entity 0 will be 11846.516312143316
toLayer3: seqnum: 14 acknum: -1 checksum: 2153 payload: kkkkkkkkkkkkkkkkkkkk

toLayer3: packet being lost
startTimer: starting timer at 11524.05774101993

EVENT time: 11554.05774101993 type: 0 entity: 0
toLayer3: seqnum: 13 acknum: -1 checksum: 2132 payload: jjjjjjjjjjjjjjjjjjjj
toLayer3: packet being lost
stopTimer: stopping timer at 11554.05774101993
stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 11554.05774101993

EVENT time: 11584.05774101993 type: 0 entity: 0
toLayer3: seqnum: 13 acknum: -1 checksum: 2132 payload: jjjjjjjjjjjjjjjjjjjj
toLayer3: packet being lost
stopTimer: stopping timer at 11584.05774101993
stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 11584.05774101993

EVENT time: 11614.05774101993 type: 0 entity: 0
toLayer3: seqnum: 13 acknum: -1 checksum: 2132 payload: jjjjjjjjjjjjjjjjjjjj
toLayer3: scheduling arrival on other side
stopTimer: stopping timer at 11614.05774101993
stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 11614.05774101993
....
===============STATISTICS======================
Number of original packets transmitted by A:1000
Number of retransmissions by A:954
Number of data packets delivered to layer 5 at B:1000
Number of ACK packets sent by B:1354
Number of corrupted packets:0
Ratio of lost packets:0.2883917775090689
Ratio of corrupted packets:0.0
Average RTT:16.469906417455793
Average communication time:43.76293848558422
==

Case 5: data packet lost, a later delivered and cumulative ack moves the sender window by more than 1
-- * Network Simulator v1.0 * --
Enter number of messages to simulate (> 0): [10] 1000
Enter packet loss probability (0.0 for no loss): [0.0] 0.3
Enter packet corruption probability (0.0 for no corruption): [0.0] 0.0
Enter average time between messages from sender's layer 5 (> 0.0): [1000] 200

Enter retransmission timeout (>0.0) [15.0] 30
Enter trace level (>= 0): [0] 3
Enter random seed: [0] 1234
.....
EVENT time: 238.65474410229803 type: 1 entity: 0
generateNextArrival(): called
generateNextArrival(): time is 238.65474410229803
generateNextArrival(): future time for event 1 at entity 0 will be 529.581591235084
toLayer3: seqnum: 3 acknum: -1 checksum: 2002 payload: dddddddddddddddddddd
toLayer3: packet being lost
startTimer: starting timer at 238.65474410229803

EVENT time: 268.654744102298 type: 0 entity: 0
toLayer3: seqnum: 2 acknum: -1 checksum: 1981 payload: cccccccccccccccccccc
toLayer3: packet being lost
stopTimer: stopping timer at 268.654744102298
stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 268.654744102298

EVENT time: 298.654744102298 type: 0 entity: 0
toLayer3: seqnum: 2 acknum: -1 checksum: 1981 payload: cccccccccccccccccccc
toLayer3: scheduling arrival on other side
stopTimer: stopping timer at 298.654744102298
stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 298.654744102298

EVENT time: 303.3361003448591 type: 2 entity: 1
toLayer3: seqnum: 0 acknum: 3 checksum: 3 payload:
toLayer3: scheduling arrival on other side

EVENT time: 307.0055848872341 type: 2 entity: 0
stopTimer: stopping timer at 307.0055848872341

EVENT time: 529.581591235084 type: 1 entity: 0
generateNextArrival(): called
generateNextArrival(): time is 529.581591235084
generateNextArrival(): future time for event 1 at entity 0 will be 924.1126743369853
toLayer3: seqnum: 4 acknum: -1 checksum: 2023 payload: eeeeeeeeeeeeeeeeeeee
toLayer3: scheduling arrival on other side
startTimer: starting timer at 529.581591235084

EVENT time: 532.5597399822992 type: 2 entity: 1
toLayer3: seqnum: 0 acknum: 3 checksum: 3 payload:

toLayer3: scheduling arrival on other side

EVENT time: 540.8963591418195 type: 2 entity: 0
stopTimer: stopping timer at 540.8963591418195
toLayer3: seqnum: 3 acknum: -1 checksum: 2002 payload: ddddddddddddddddddddd
toLayer3: packet being lost
startTimer: starting timer at 540.8963591418195

EVENT time: 570.8963591418195 type: 0 entity: 0
toLayer3: seqnum: 3 acknum: -1 checksum: 2002 payload: ddddddddddddddddddddd
toLayer3: packet being lost
stopTimer: stopping timer at 570.8963591418195
stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 570.8963591418195

EVENT time: 600.8963591418195 type: 0 entity: 0
toLayer3: seqnum: 3 acknum: -1 checksum: 2002 payload: ddddddddddddddddddddd
toLayer3: scheduling arrival on other side
stopTimer: stopping timer at 600.8963591418195
stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 600.8963591418195

EVENT time: 607.7091280861843 type: 2 entity: 1
toLayer3: seqnum: 0 acknum: 5 checksum: 5 payload:
toLayer3: scheduling arrival on other side

EVENT time: 610.2434156315805 type: 2 entity: 0
stopTimer: stopping timer at 610.2434156315805
…….
==============STATISTICS=====================
Number of original packets transmitted by A:1000
Number of retransmissions by A:954
Number of data packets delivered to layer 5 at B:1000
Number of ACK packets sent by B:1354
Number of corrupted packets:0
Ratio of lost packets:0.2883917775090689
Ratio of corrupted packets:0.0
Average RTT:16.469906417455793
Average communication time:43.76293848558422
===

toLayer3: packet being lost
startTimer: starting timer at 11524.05774101993

EVENT time: 11554.05774101993 type: 0 entity: 0
toLayer3: seqnum: 13 acknum: -1 checksum: 2132 payload: jjjjjjjjjjjjjjjjjjjj
toLayer3: packet being lost
stopTimer: stopping timer at 11554.05774101993
stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 11554.05774101993

EVENT time: 11584.05774101993 type: 0 entity: 0
toLayer3: seqnum: 13 acknum: -1 checksum: 2132 payload: jjjjjjjjjjjjjjjjjjjj
toLayer3: packet being lost
stopTimer: stopping timer at 11584.05774101993
stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 11584.05774101993

EVENT time: 11614.05774101993 type: 0 entity: 0
toLayer3: seqnum: 13 acknum: -1 checksum: 2132 payload: jjjjjjjjjjjjjjjjjjjj
toLayer3: scheduling arrival on other side
stopTimer: stopping timer at 11614.05774101993
stopTimer: Warning: Unable to cancel your timer
startTimer: starting timer at 11614.05774101993
….
===============STATISTICS=====================
Number of original packets transmitted by A:1000
Number of retransmissions by A:954
Number of data packets delivered to layer 5 at B:1000
Number of ACK packets sent by B:1354
Number of corrupted packets:0
Ratio of lost packets:0.2883917775090689
Ratio of corrupted packets:0.0
Average RTT:16.469906417455793
Average communication time:43.76293848558422
==

Case 5: data packet lost, a later delivered and cumulative ack moves the sender window by more than 1

-- * Network Simulator v1.0 * --
Enter number of messages to simulate (> 0): [10] 1000
Enter packet loss probability (0.0 for no loss): [0.0] 0.3
Enter packet corruption probability (0.0 for no corruption): [0.0] 0.0
Enter average time between messages from sender's layer 5 (> 0.0): [1000] 200

Output Trace (GBN with SACK Option)

Case1: no loss and corruption, the GBN with SACK option works the same behavior as the SR with cumulative acks

generateNextArrival(): called

generateNextArrival(): time is 0.0

generateNextArrival(): future time for event 1 at entity 0 will be 43.4339426862392

EVENT time: 43.4339426862392   type: 1   entity: 0

generateNextArrival(): called

generateNextArrival(): time is 43.4339426862392

generateNextArrival(): future time for event 1 at entity 0 will be 67.36045411542099

toLayer3: seqnum: 0   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 1934   payload: aaaaaaaaaaaaaaaaaaaa

toLayer3: scheduling arrival on other side

startTimer: starting timer at 43.4339426862392

EVENT time: 45.66060365611743   type: 2   entity: 1

* B received packet*

seqnum: 0   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 1934   payload: aaaaaaaaaaaaaaaaaaaa

toLayer3: seqnum: 0   acknum: 1   sack: [0, 0, 0, 0, 0]   checksum: 1   payload:

toLayer3: scheduling arrival on other side

EVENT time: 47.5694753868822106   type: 2   entity: 0

* A received packet*

seqnum: 0   acknum: 1   sack: [0, 0, 0, 0, 0]   checksum: 1   payload:

stopTimer: stopping timer at 47.569475386822106

Sequence 0 received time: 47.569475386822106

tempTime = 4.135532700582907

It is not retransmit

totalRTT = 4.135532700582907

totalComTime = 4.135532700582907

EVENT time: 67.36045411542099   type: 1   entity: 0

generateNextArrival(): called

generateNextArrival(): time is 67.36045411542099

generateNextArrival(): future time for event 1 at entity 0 will be 83.31553086947233

toLayer3: seqnum: 1   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 1955   payload: bbbbbbbbbbbbbbbbbbbb

toLayer3: scheduling arrival on other side

startTimer: starting timer at 67.36045411542099

EVENT time: 68.78865932187871   type: 2   entity: 1

* B received packet*

seqnum: 1   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 1955   payload: bbbbbbbbbbbbbbbbbbbb

toLayer3: seqnum: 0   acknum: 2   sack: [0, 1, 0, 0, 0]   checksum: 3   payload:

toLayer3: scheduling arrival on other side

EVENT time: 70.75603503524887   type: 2   entity: 0

* A received packet*

seqnum: 0   acknum: 2   sack: [0, 1, 0, 0, 0]   checksum: 3   payload:

stopTimer: stopping timer at 70.75603503524887

Sequence 1 received time: 70.75603503524887

tempTime = 3.3955809198278786

It is not retransmit

totalRTT = 7.531113620410785

totalComTime = 7.531113620410785

Case2: Set loss rate as 0.4, and no corruption rate. The programs works well and the difference between GBN and SR is marked as ① below.

EVENT time: 17816.827906125065   type: 1   entity: 0

generateNextArrival(): called

generateNextArrival(): time is 17816.827906125065

generateNextArrival(): future time for event 1 at entity 0 will be 18090.81087679678

toLayer3:   seqnum: 6   acknum: -1   sack: [-1,   -1,   -1,   -1,   -1]   checksum: 2180   payload: mmmmmmmmmmmmmmmmmmmmm

toLayer3: packet being lost

startTimer: starting timer at 17816.827906125065

> Send the seq num 6 packet but is lost during transmission

EVENT time: 17846.827906125065   type: 0   entity: 0

A timeout, retransmit

toLayer3: seqnum: 4   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 2138   payload: kkkkkkkkkkkkkkkkkkkkk

toLayer3: scheduling arrival on other side

stopTimer: stopping timer at 17846.827906125065

stopTimer: Warning: Unable to cancel your timer

startTimer: starting timer at 17846.827906125065

EVENT time: 17847.95452742088   type: 2   entity: 1

* B received packet*

seqnum: 4   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 2138   payload: kkkkkkkkkkkkkkkkkkkkk

toLayer3: seqnum: 0   acknum: 5   sack: [0, 1, 2, 3, 4]   checksum: 15   payload:

toLayer3: scheduling arrival on other side

EVENT time: 17854.367931150242   type: 2   entity: 0

* A received packet*

seqnum: 0   acknum: 5   sack: [0, 1, 2, 3, 4]   checksum: 15   payload:

stopTimer: stopping timer at 17854.367931150242

Sequence 4 received time: 17854.367931150242

tempTime = 216.13434247871191

> The sequence number 4 packet is sent and received and A received ACK

totalComTime = 8610.553758354436

EVENT time: 18090.81087679678   type: 1   entity: 0

generateNextArrival(): called

generateNextArrival(): time is 18090.81087679678

generateNextArrival(): future time for event 1 at entity 0 will be 18454.860072634056

toLayer3: seqnum: 7   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 2201   payload: nnnnnnnnnnnnnnnnnnnn

toLayer3: packet being lost

startTimer: starting timer at 18090.81087679678

Send the seq num 7 packet but
is lost during transmission

EVENT time: 18120.81087679678   type: 0   entity: 0

A timeout, retransmit

toLayer3: seqnum: 5   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 2159   payload: lllllllllllllllllll

toLayer3: scheduling arrival on other side

stopTimer: stopping timer at 18120.81087679678

stopTimer: Warning: Unable to cancel your timer

startTimer: starting timer at 18120.81087679678

If the packet is lost during A to B, then there will be a timeout in A.
Ex, the seq num 5 packet timeout, retransmit

EVENT time: 18128.04196295053   type: 2   entity: 1

  B received packet

seqnum: 5   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 2159   payload: lllllllllllllllllll

toLayer3: seqnum: 0   acknum: 6   sack: [1, 2, 3, 4, 5]   checksum: 21   payload:

toLayer3: scheduling arrival on other side

B receives seq num 5 packet,
set ACK and SACK

EVENT time: 18137.208899197365   type: 2   entity: 0

  A received packet

seqnum: 0   acknum: 6   sack: [1, 2, 3, 4, 5]   checksum: 21   payload:

stopTimer: stopping timer at 18137.208899197365

Sequence 5 received time: 18137.208899197365

tempTime = 449.1080660420557

totalComTime = 9059.661824396491

A received ACK 6 for the
first time, slide window

EVENT time: 18454.860072634056   type: 1   entity: 0

generateNextArrival(): called

generateNextArrival(): time is 18454.860072634056

generateNextArrival(): future time for event 1 at entity 0 will be 18528.971221045565

toLayer3: seqnum: 8   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 2222   payload: oooooooooooooooooooo

toLayer3: scheduling arrival on other side

startTimer: starting timer at 18454.860072634056

Send the seq num 8 packet

EVENT time: 18462.297982726526   type: 2   entity: 1

  B received packet

seqnum: 8   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 2222   payload: oooooooooooooooooooo

B recieved out of order, not duplicate

B receives 8 but not received 6 and 7, send ACK 6

toLayer3: seqnum: 0   acknum: 6   sack: [2, 3, 4, 5, 8]   checksum: 28   payload:

toLayer3: scheduling arrival on other side

EVENT time: 18465.90044862209   type: 2   entity: 0

* A received packet*

seqnum: 0   acknum: 6   sack: [2, 3, 4, 5, 8]   checksum: 28   payload:

stopTimer: stopping timer at 18465.90044862209

A received duplicate ACK, retransmit

retransmitPacket: seqnum: 6   acknum: -1   sack: [-1,   -1,   -1,   -1,   -1]   checksum: 2180   payload: mmmmmmmmmmmmmmmmmmmmm

toLayer3:   seqnum: 6   acknum: -1   sack: [-1,   -1,   -1,   -1,   -1]   checksum: 2180   payload: mmmmmmmmmmmmmmmmmmmmm

toLayer3: scheduling arrival on other side

startTimer: starting timer at 18465.90044862209

retransmitPacket: seqnum: 7   acknum: -1   sack: [-1,   -1,   -1,   -1,   -1]   checksum: 2201   payload: nnnnnnnnnnnnnnnnnnnnn

toLayer3: seqnum: 7   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 2201   payload: nnnnnnnnnnnnnnnnnnnnn

toLayer3: packet being lost

startTimer: starting timer at 18465.90044862209

startTimer: Warning: Attempting to start a timer that is already running

EVENT time: 18473.749259926146   type: 2   entity: 1

* B received packet*

seqnum: 6   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 2180   payload: mmmmmmmmmmmmmmmmmmmmm

toLayer3: seqnum: 0   acknum: 7   sack: [3, 4, 5, 8, 6]   checksum: 33   payload:

toLayer3: packet being lost

① A received ACK 6 for the second time, retransmit all unacked packets, which are 7 and 8. It is different from SR in part 1

Case3. Set corruption rate as 0.4, and no loss rate.

EVENT time: 83.31553086947233   type: 1   entity: 0

generateNextArrival(): called

generateNextArrival(): time is 83.31553086947233

generateNextArrival(): future time for event 1 at entity 0 will be 238.65474410229803

Sequence 2 first send time: 83.31553086947233

toLayer3: seqnum: 2   acknum: -1   checksum: 1981   payload: cccccccccccccccccccc

toLayer3: packet being corrupted

toLayer3: scheduling arrival on other side

startTimer: starting timer at 83.31553086947233

startTimer: Warning: Attempting to start a timer that is already running

EVENT time: 84.74373607593004   type: 2   entity: 1

* B received packet*

seqnum: 2   acknum: -1   checksum: 1981   payload: ?cccccccccccccccccccc

if B receives corrupt packets, do nothing but wait A for timeout but retransmission

EVENT time: 97.36045411542099   type: 0   entity: 0

A timeout, retransmit

toLayer3: seqnum: 1   acknum: -1   checksum: 1960   payload: bbbbbbbbbbbbbbbbbbbb

toLayer3: scheduling arrival on other side

startTimer: starting timer at 97.36045411542099

……

EVENT time: 39596.435438093205   type: 1   entity: 0

generateNextArrival(): called

generateNextArrival(): time is 39596.435438093205

generateNextArrival(): future time for event 1 at entity 0 will be 39780.88930936613

toLayer3: seqnum: 5   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 2319   payload: ttttttttttttttttttttt

toLayer3: scheduling arrival on other side

startTimer: starting timer at 39596.435438093205

startTimer: Warning: Attempting to start a timer that is already running

> B receives out of order packet, B buffer it, send cumulative and selective ack

EVENT time: 39601.624103518785   type: 2   entity: 1

B received packet

seqnum: 5   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 2319   payload: ttttttttttttttttttttt

B recieved out of order, not duplicate

toLayer3: seqnum: 0   acknum: 4   sack: [0, 1, 2, 3, 5]   checksum: 15   payload:

toLayer3: packet being corrupted

toLayer3: scheduling arrival on other side

EVENT time: 39611.542802209544   type: 2   entity: 0

A received packet

seqnum: 0   acknum: 4   sack: [0, 1, 2, 3, 5]   checksum: 15   payload: ?

A received corrupt Ack

> A receives corrupt packet, A will wait for timeout and retransmit.

EVENT time: 39626.21570482498   type: 0   entity: 0

A timeout, retransmit

toLayer3: seqnum: 4   acknum: -1   sack: [-1, -1, -1, -1, -1]   checksum: 2298   payload: ssssssssssssssssssss

toLayer3: scheduling arrival on other side

stopTimer: stopping timer at 39626.21570482498

stopTimer: Warning: Unable to cancel your timer

startTimer: starting timer at 39626.21570482498

The program also works when there's corruption as well as loss. The outputs are similar as above.