

# מטרה 3 - רשתות תקשורת

## 208113381,319096251

### Tcp Congestion:-

Transmission Control Protocol (TCP) uses a network congestion-avoidance algorithm that includes various aspects of an additive increase/multiplicative decrease (AIMD) scheme, along with other schemes including slow start<sup>[1]</sup> and congestion window (CWND), to achieve congestion avoidance. The TCP congestion-avoidance algorithm is the primary basis for congestion control in the Internet.<sup>[2][3][4]</sup> Per the end-to-end principle, congestion control is largely a function of internet hosts, not the network itself. There are several variations and versions of the algorithm implemented in protocol stacks of operating systems of computers that connect to the Internet.

### Congestion window

---

In TCP, the congestion window (CWND) is one of the factors that determines the number of bytes that can be sent out at any time. The congestion window is maintained by the sender and is a means of stopping a *link* between the sender and the receiver from becoming overloaded with too much traffic. This should not be confused with the sliding window maintained by the sender which exists to prevent the receiver from becoming overloaded. The congestion window is calculated by estimating how much congestion there is on the link.

When a connection is set up, the congestion window, a value maintained independently at each host, is set to a small multiple of the maximum segment size (MSS) allowed on that connection. Further variance in the congestion window is dictated by an additive increase/multiplicative decrease (AIMD) approach. This means that if all segments are received and the acknowledgments reach the sender on time, some constant is added to the window size. It will follow different algorithms.

A system administrator may adjust the maximum window size limit, or adjust the constant added during additive increase, as part of TCP tuning.

The flow of data over a TCP connection is also controlled by the use of the receive window advertised by the receiver. A sender can send data less than its own congestion window and the receive window.

**הסביר על כל אלגוריתם שמשתמש בהם בקוד.**

**CUBIC** is a [network congestion avoidance](#) algorithm for [TCP](#) which can achieve high bandwidth connections over networks more quickly and reliably in the face of high latency than earlier algorithms. It helps optimize [long fat networks](#).

CUBIC increases its window to be real-time dependent, not RTT dependent like BIC. The calculation for cwnd (congestion window) is simpler than BIC, too.

Define the following variables:

- $\beta$ : Multiplicative decrease factor
- $w_{\max}$ : Window size just before the last reduction
- $T$ : Time elapsed since the last window reduction
- $C$ : A scaling constant
- $cwnd$ : The congestion window at the current time

**Reno** is the extension of [TCP Tahoe](#), and **NewReno** is the extension of TCP Reno. In Reno, when packet loss occurs, the sender reduces the cwnd by 50% along with the ssthresh value. This would allow the network to come out of the congestion state easily. But Reno suffered from a very critical backlog which hurts its performance.

When multiple packets are dropped in the same congestion window (say 1-10) then every time it knows about a packet loss it reduces the cwnd by 50%. So, for 2 packet loss, it will reduce the cwnd by 4 times (50% twice). But, one reduction of 50% per congestion window was enough for recovering all those lost packets. Say  $cwnd=1024$  and 10 packets are dropped in this window, Reno will reduce cwnd by 50% 10 times, finally  $cwnd=1024/2^{10} = 1$ , it is astonishing. It would take 10 RTTs by the sender to again grow its cwnd up to 1024 using [slow start](#) leave alone the AIMD algorithm.

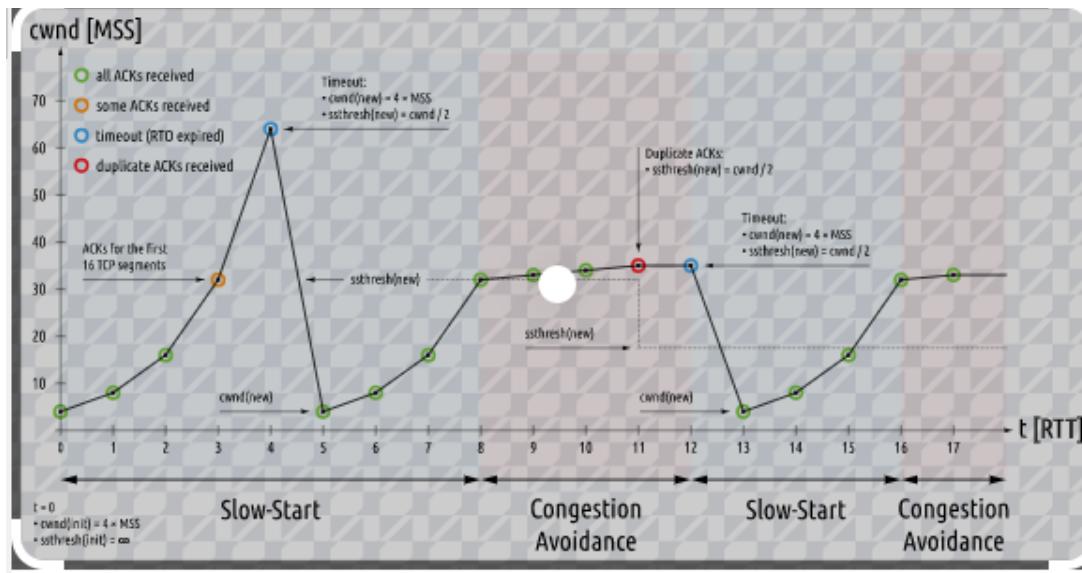
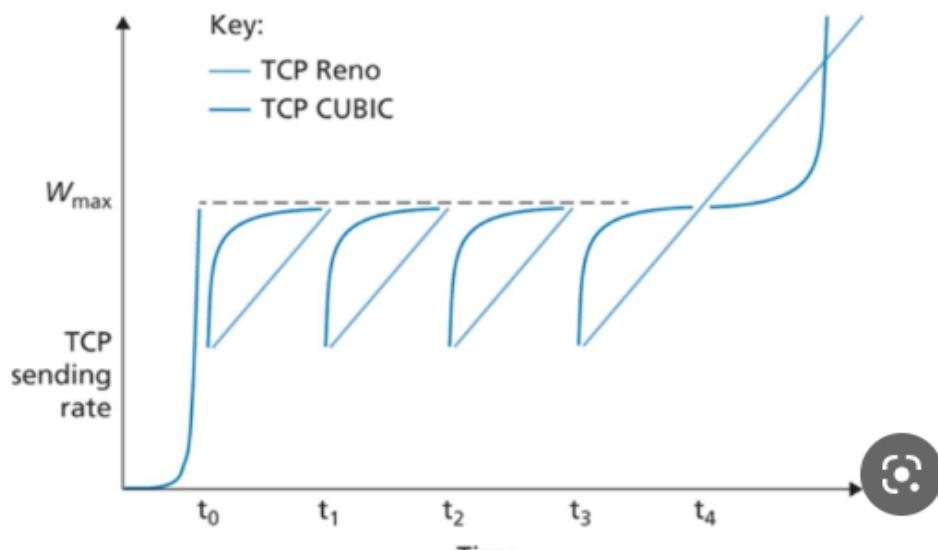


Figure 3.54 TCP congestion avoidance sending rates: TCP Reno and TCP CUBIC



In all iterations and works in this code work with file 1mb , buffer size 524288.

The first work with 0% loss data , 8 times send the file.

```
codebind@codebind:~/CLionProjects/Hw3_TCC$ ./sender
client connected to server!
RUN 1
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno
client (0) Current CC, type: reno
Send the second part of the file
1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 2
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno
client (0) Current CC, type: reno
Send the second part of the file
1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 3
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno
client (0) Current CC, type: reno
Send the second part of the file
1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 4
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno
client (0) Current CC, type: reno
Send the second part of the file
1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 5
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno
client (0) Current CC, type: reno
Send the second part of the file
1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
*****  
Sender send the exit message for the receiver!!
RUN 6
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno
client (0) Current CC, type: reno
Send the second part of the file
1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 7
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 8
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
0
*****
```

Sender send the exit message for the receiver!!  
codebind@codebind:~/CLionProjects/Hw3\_TCC\$

### The receivers of them:-

NOTE (Where write \*\*in\*) this is mean the authentication sent  
sorry in the final i'm noted it!.

```
codebind@codebind:~/CLionProjects/Hw3_TCC$ make all
make: Nothing to be done for 'all'.
codebind@codebind:~/CLionProjects/Hw3_TCC$ sudo tc qdisc del dev lo root netem
[sudo] password for codebind:
codebind@codebind:~/CLionProjects/Hw3_TCC$ ./receiver
Waiting for incoming connections
the server is ready!

Receive 1
client number connection accepted
2      65482
2      130964
2      524288
**in*
Time in microseconds: 0.000048 microseconds
size2==524288
2      524288
Switching to Reno algorithm
Time in microseconds: 0.000284 microseconds
size2==1048576
All the file is recv done
*****
codebind@codebind:~/CLionProjects/Hw3_TCC$ ./receiver
Receive 2
2      524288
**in*
Time in microseconds: 0.000362 microseconds
size2==524288
2      524288
Switching to Reno algorithm
Time in microseconds: 0.000383 microseconds
size2==1048576
All the file is recv done
*****
```

```
codebind@codebind:~/CLionProjects/Hw3_TCC$ ./receiver
Receive 3
2      524288
**in*
Time in microseconds: 0.000308 microseconds
size2==524288
2      524288
Switching to Reno algorithm
Time in microseconds: 0.000292 microseconds
size2==1048576
All the file is recv done
*****
codebind@codebind:~/CLionProjects/Hw3_TCC$ ./receiver
Receive 4
2      327415
2      458381
2      523864
2      524288
**in*
Time in microseconds: 0.000025 microseconds
size2==524288
2      524288
Switching to Reno algorithm
Time in microseconds: 0.000421 microseconds
size2==1048576
All the file is recv done
*****
```

```
codebind@codebind:~/CLionProjects/Hw3_TCC$ ./receiver
Receive 5
2      524288
**in*
Time in microseconds: 0.000379 microseconds
size2==524288
2      524288
Switching to Reno algorithm
Time in microseconds: 0.000519 microseconds
size2==1048576
All the file is recv done
*****
codebind@codebind:~/CLionProjects/Hw3_TCC$ ./receiver
Receive 6
2      524288
**in*
Time in microseconds: 0.000545 microseconds
size2==524288
2      524288
Switching to Reno algorithm
Time in microseconds: 0.000624 microseconds
size2==1048576
All the file is recv done
*****
```

```

Receive 7
2      524288
**in*
Time in microseconds: 0.000443 microseconds
size2==524288

2      524288

Switching to Reno algorithm
Time in microseconds: 0.000621 microseconds
size2==1048576

All the file is recv done
*****
```

```

Receive 8
2      523864
2      524288
**in*
Time in microseconds: 0.000004 microseconds
size2==524288

2      524288

Switching to Reno algorithm
Time in microseconds: 0.000318 microseconds
size2==1048576

All the file is recv done
*****
```

The Average of them for each part:-  
part 1 with cc algorithm cubic  
part 2 with cc algorithm reno

```

Times Summary:
Part1:-
0.000048
0.000362
0.000308
0.000025
0.000379
0.000545
0.000443
0.000004
Part2:-
0.000284
0.000383
0.000292
0.000421
0.000519
0.000624
0.000621
0.000318

*****
Averages:
Part1: 0.000264
Averages:
Part2: 0.000433

END
codebind@codebind:~/CLionProjects/Hw3_TCC$
```

Work with wireshark:-

Two pictures to the public get packets. (One for the beginning ,other for the end).

No.	Time	Source	Destination	Protocol	Length	Info	No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7601972 Ack=148 Win=65536 Len=6548	225	16.297784998	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7601972 Ack=148 Win=65536 Len=6548
2	0.000000000	127.0.0.1	227.0.0.1	TCP	60536	44598 ... 5336 [ACK] Seq=7601972 Ack=148 Win=65536 Len=6548	226	16.297783940	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7601972 Ack=148 Win=65536 Len=6548
3	0.000000100	127.0.0.1	227.0.0.1	TCP	664498	- 5336 [ACK] Seq=1 Ack=1 Win=65536 Len=9 Tsvl=1467	227	16.297807374	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
4	0.000000100	127.0.0.1	227.0.0.1	TCP	32867	44598 ... 5336 [ACK] Seq=1 Ack=1 Win=65536 Len=9 Tsvl=1467	228	16.297807374	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
5	0.000585078	127.0.0.1	227.0.0.1	TCP	66538	- 44598 [ACK] Seq=1 Ack=32742 Win=48648 Len=9 Tsvl=1	229	16.297983108	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
6	0.000585078	127.0.0.1	227.0.0.1	TCP	32867	44598 ... 5336 [PSH, ACK] Seq=32742 Ack=1 Win=65536 Len=327	230	16.297933834	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [PSH, ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
7	0.000585078	127.0.0.1	227.0.0.1	TCP	66538	- 44598 [ACK] Seq=32742 Ack=1 Win=65536 Len=327	231	16.298024454	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
8	0.000585176	127.0.0.1	227.0.0.1	TCP	32867	44598 ... 5336 [ACK] Seq=65483 Ack=1 Win=65536 Len=327	232	16.298084454	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
9	0.000585176	127.0.0.1	227.0.0.1	TCP	66538	- 44598 [ACK] Seq=65483 Ack=1 Win=65536 Len=327	233	16.298481988	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
10	0.000585176	127.0.0.1	227.0.0.1	TCP	32867	44598 ... 5336 [ACK] Seq=1 Ack=19860 Win=65536 Len=9 Tsvl=1	234	16.298481988	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
11	0.000585176	127.0.0.1	227.0.0.1	TCP	66538	- 44598 [ACK] Seq=1 Ack=19860 Win=65536 Len=9 Tsvl=1	235	16.298483753	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
12	0.000585176	127.0.0.1	227.0.0.1	TCP	32867	44598 ... 5336 [ACK] Seq=1 Ack=19860 Win=65536 Len=9 Tsvl=1	236	16.298483753	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
13	0.000585176	127.0.0.1	227.0.0.1	TCP	66538	- 44598 [ACK] Seq=1 Ack=19860 Win=65536 Len=9 Tsvl=1	237	16.298542893	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
14	0.000585176	127.0.0.1	227.0.0.1	TCP	32867	44598 ... 5336 [ACK] Seq=1 Ack=19860 Win=65536 Len=9 Tsvl=1	238	16.298542893	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
15	0.000585176	127.0.0.1	227.0.0.1	TCP	66538	- 44598 [ACK] Seq=1 Ack=19860 Win=65536 Len=9 Tsvl=1	239	16.298542893	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
16	0.000585176	127.0.0.1	227.0.0.1	TCP	32867	44598 ... 5336 [ACK] Seq=1 Ack=19860 Win=65536 Len=9 Tsvl=1	240	16.298591618	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
17	0.000585176	127.0.0.1	227.0.0.1	TCP	66538	- 44598 [ACK] Seq=1 Ack=19860 Win=65536 Len=9 Tsvl=1	241	16.298812497	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
18	0.000585176	127.0.0.1	227.0.0.1	TCP	32867	44598 ... 5336 [ACK] Seq=261629 Ack=1 Win=65536 Len=327	242	16.298812497	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
19	0.000585176	127.0.0.1	227.0.0.1	TCP	66538	- 44598 [ACK] Seq=261629 Ack=1 Win=65536 Len=327	243	16.298812497	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
20	0.000585176	127.0.0.1	227.0.0.1	TCP	32867	44598 ... 5336 [ACK] Seq=1 Ack=229188 Win=65536 Len=327	244	16.298952598	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
21	0.000585176	127.0.0.1	227.0.0.1	TCP	66538	- 44598 [ACK] Seq=1 Ack=229188 Win=65536 Len=327	245	16.298976944	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
22	0.000585176	127.0.0.1	227.0.0.1	TCP	32867	44598 ... 5336 [ACK] Seq=1 Ack=229188 Win=65536 Len=327	246	16.299051344	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
23	0.000585176	127.0.0.1	227.0.0.1	TCP	66538	- 44598 [ACK] Seq=1 Ack=229188 Win=65536 Len=327	247	16.299051344	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
24	0.000585176	127.0.0.1	227.0.0.1	TCP	32867	44598 ... 5336 [ACK] Seq=1 Ack=386152 Win=65536 Len=327	248	16.299259629	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
25	0.000585176	127.0.0.1	227.0.0.1	TCP	66538	- 44598 [ACK] Seq=1 Ack=386152 Win=65536 Len=327	249	16.299259629	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
26	0.000585176	127.0.0.1	227.0.0.1	TCP	32867	44598 ... 5336 [ACK] Seq=1 Ack=392893 Win=65536 Len=32741 Ts	250	16.299254666	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548
27	0.000585176	127.0.0.1	227.0.0.1	TCP	66538	- 44598 [ACK] Seq=1 Ack=392893 Win=65536 Len=32741 Ts	251	16.299254666	127.0.0.1	227.0.0.1	TCP	60549	44598 ... 5336 [ACK] Seq=7607455 Ack=148 Win=65536 Len=6548

Know i put pictures for ack and push that we get after send each file, and the end finally.

1.)

51 0.052032207	127.0.0.1	127.0.0.1	TCP	66 5336 → 44598 [ACK] Seq=22 Ack=1048577 Win=2684928 Len=0 TS
52 7.497232796	127.0.0.1	127.0.0.1	TCP	67 44598 → 5336 [PSH, ACK] Seq=1048577 Ack=22 Win=65536 Len=0 TS
53 7.497251925	127.0.0.1	127.0.0.1	TCP	66 5336 → 44598 [ACK] Seq=22 Ack=1048578 Win=2684928 Len=0 TS
54 7.497443113	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=1048578 Ack=22 Win=65536 Len=65483
55 7.497454318	127.0.0.1	127.0.0.1	TCP	66 5336 → 44598 [ACK] Seq=22 Ack=1114061 Win=2815872 Len=0 TS
56 7.497475530	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=1114061 Ack=22 Win=65536 Len=65483
57 7.497482772	127.0.0.1	127.0.0.1	TCP	66 5336 → 44598 [ACK] Seq=22 Ack=1170514 Win=28146916 Len=0 TS

2.)

80 7.498512473	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=1031247 Ack=43 Win=65536 Len=65483
81 7.498538757	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=2031247 Ack=43 Win=65536 Len=65483
82 7.498551012	127.0.0.1	127.0.0.1	TCP	490 44598 → 5336 [PSH, ACK] Seq=2096730 Ack=43 Win=65536 Len=4
83 7.498655410	127.0.0.1	127.0.0.1	TCP	66 5336 → 44598 [ACK] Seq=43 Ack=2097154 Win=3112448 Len=0 TS
84 9.428691060	127.0.0.1	127.0.0.1	TCP	67 44598 → 5336 [PSH, ACK] Seq=2097154 Ack=43 Win=65536 Len=0
85 9.428914751	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=2097155 Ack=43 Win=65536 Len=65483
86 9.428954311	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=2162638 Ack=43 Win=65536 Len=65483

3.)

118 11.539710215	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=3604113 Ack=64 Win=65536 Len=65483
119 11.539740279	127.0.0.1	127.0.0.1	TCP	490 44598 → 5336 [PSH, ACK] Seq=3669596 Ack=64 Win=65536 Len=4
120 11.539744289	127.0.0.1	127.0.0.1	TCP	66 5336 → 44598 [ACK] Seq=64 Ack=3670020 Win=3144960 Len=0 TS
121 11.539768720	127.0.0.1	127.0.0.1	TCP	87 5336 → 44598 [PSH, ACK] Seq=64 Ack=3670020 Win=3144960 Len=0
122 11.540069214	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=3670020 Ack=85 Win=65536 Len=65483
123 11.540118669	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=3735503 Ack=85 Win=65536 Len=65483

4.)

140 10.236300307	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=4002000 Ack=80 Win=65536 Len=65483
147 13.236301324	127.0.0.1	127.0.0.1	TCP	66 5336 → 44598 [ACK] Seq=85 Ack=4718173 Win=3112448 Len=0 TS
148 13.236330812	127.0.0.1	127.0.0.1	TCP	490 44598 → 5336 [PSH, ACK] Seq=4718173 Ack=85 Win=65536 Len=4
149 13.236466290	127.0.0.1	127.0.0.1	TCP	87 5336 → 44598 [PSH, ACK] Seq=85 Ack=4718597 Win=3144960 Len=0
150 13.236791804	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=4718597 Ack=106 Win=65536 Len=65483
151 13.236845096	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=4784080 Ack=106 Win=65536 Len=65483
152 13.236880282	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=4849563 Ack=106 Win=65536 Len=65483

There more but cut here,

the final is:-

8.)

174 14.303990971	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=5701267 Ack=106 Win=65536 Len=65483
175 14.394019780	127.0.0.1	127.0.0.1	TCP	490 44598 → 5336 [PSH, ACK] Seq=5766750 Ack=106 Win=65536 Len=0
176 14.394023989	127.0.0.1	127.0.0.1	TCP	66 5336 → 44598 [ACK] Seq=106 Ack=5767174 Win=3144960 Len=0
177 14.394042197	127.0.0.1	127.0.0.1	TCP	87 5336 → 44598 [PSH, ACK] Seq=106 Ack=5767174 Win=3144960 Len=0
178 14.394275817	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=5767174 Ack=127 Win=65536 Len=65483
179 14.394323621	127.0.0.1	127.0.0.1	TCP	65549 44598 → 5336 [ACK] Seq=5832657 Ack=127 Win=65536 Len=65483

The second work , with 10% loss of data.

**The first work with 10% loss data , 8 times send the file.**

```
codebind@codebind:~/CLionProjects/Hw3_TCC$ ./sender
client connected to server!
RUN 1
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

Sender send the exit message for the receiver!!

```
Sender send the exit message for the receiver!!
RUN 2
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 3
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 4
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 5
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 6
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
RUN 7
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 8
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
0
*****
```

Sender send the exit message for the receiver!!  
codebind@codebind:~/CLionProjects/Hw3\_TCC\$

The End of sends.

NOTE:-

The sentences “Sender send the exit message for the receiver” is that the to options , 1 for continue = send the file one more 0=else ,Stop.

know ,pictures for receiving the sent files.

NOTE (Where write \*\*in\*) this is mean the authentication sent  
sorry in the final i'm noted it!.

```
codebind@codebind:~/CtionProjects/Hw3_TCC$ sudo tc qdisc add dev lo root netem loss 10%
[sudo] password for codebind:
codebind@codebind:~/CtionProjects/Hw3_TCC$ make all
gcc -o receiver receiver.c
codebind@codebind:~/CtionProjects/Hw3_TCC$ ./receiver
Waiting for incoming connections
the server is ready!
Receive 1
client number connection accepted
2      65482
2      130964
2      196446
2      425633
2      523883
2      524288
**in*
Time in microseconds: 0.6399932 microseconds
size2==524288
2      392898
2      458381
2      524288
Switching to Reno algorithm
Time in microseconds: 0.000021 microseconds
size2==1048576
All the file is recv done
*****
```

```
Receive 2
2      524288
**in*
Time in microseconds: 0.0000566 microseconds

size2==524288

2      65483
2      523864
2      524288

Switching to Reno algorithm
Time in microseconds: 0.000004 microseconds

size2==1048576

All the file is recv done

*****
```

```
Receive 3
2      524288
**in*
Time in microseconds: 0.0000803 microseconds

size2==524288

2      261932
2      524288

Switching to Reno algorithm
Time in microseconds: 0.0000719 microseconds

size2==1048576

All the file is recv done

*****
```

```
Receive 4
2      65483
2      261932
2      458381
2      524288
**in*
Time in microseconds: 0.000016 microseconds

size2==524288

2      130966
2      327415
2      524288

Switching to Reno algorithm
Time in microseconds: 0.000038 microseconds

size2==1048576
```

```
Receive 5
2      261932
2      327415
2      392898
2      524288
**in*
Time in microseconds: 0.237928 microseconds

size2==524288

2      130966
2      196449
2      327415
2      524288

Switching to Reno algorithm
Time in microseconds: 0.000033 microseconds

size2==1048576

All the file is recv done

*****
```

```
Receive 6
2      65483
2      196449
2      261932
2      524288
**in*
Time in microseconds: 0.042859 microseconds

size2==524288

2      130966
2      327415
2      524288

Switching to Reno algorithm
Time in microseconds: 0.000083 microseconds

size2==1048576

All the file is recv done

*****
```

```
Receive 7
2      524288
**in*
Time in microseconds: 0.000098 microseconds

size2==524288

2      196449
2      392898
2      524288

Switching to Reno algorithm
Time in microseconds: 0.000018 microseconds

size2==1048576

All the file is recv done

*****
```

```
Receive 8
2      65483
2      130966
2      524288
**in*
Time in microseconds: 0.070156 microseconds

size2==524288

2      196449
2      261932
2      523864
2      524288

Switching to Reno algorithm
Time in microseconds: 0.226558 microseconds

size2==1048576

All the file is recv done

*****
```

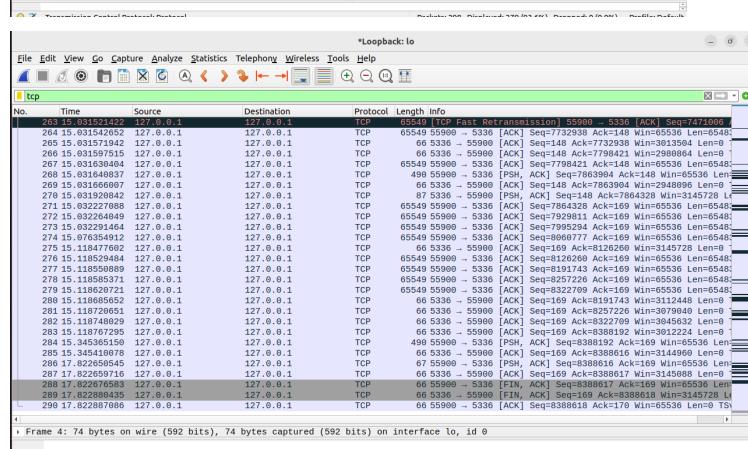
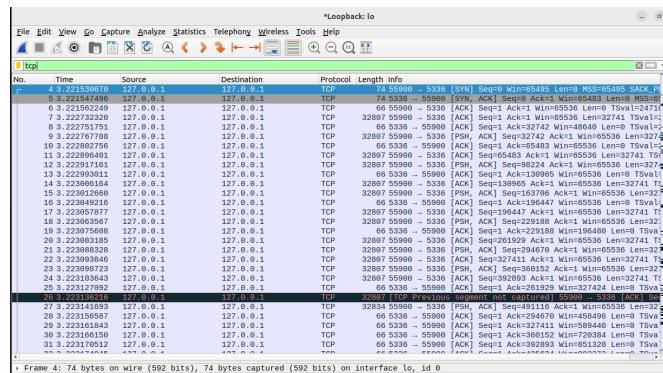
The Times Summary and Average for each part (part1 cubic, part2 reno):-

```

Times Summary:
Part1:- 
0.639932
0.000566
0.000803
0.000016
0.237928
0.042859
0.000098
0.070156
Part2:- 
0.000021
0.000004
0.000719
0.000038
0.000033
0.000083
0.000018
0.226558
0.070156
*****
Averages:
Part1: 0.124045
Averages:
Part2: 0.028434
END
codebind@codebind:~/CLionProjects/Hw3_TCC$
```

Know working with wireshark:-

Two pictures public (first, end the file wireshark).



Here we see the loss data

24	3.223163643	127.0.0.1	127.0.0.1	TCP	32807 55900 → 5336 [ACK] Seq=392893 Ack=1 Win=65536 Len=32741 TS
25	3.223127092	127.0.0.1	127.0.0.1	TCP	66 5336 → 55900 [ACK] Seq=1 Ack=261929 Win=327424 Len=0 TS
26	3.223136216	127.0.0.1	127.0.0.1	TCP	32807 [TCP Previous segment not captured] 55900 → 5336 [ACK] Seq=1 Ack=261929 Win=327424 Len=0 TS
27	3.223141693	127.0.0.1	127.0.0.1	TCP	32834 55900 → 5336 [PSH, ACK] Seq=491116 Ack=1 Win=65536 Len=32741 TS
28	3.223141693	127.0.0.1	127.0.0.1	TCP	66 5336 → 55900 [ACK] Seq=1 Ack=261929 Win=327424 Len=0 TS

and know , retransmission

34	3.223186525	127.0.0.1	127.0.0.1	TCP	78 [TCP Window Update] 5336 → 55900 [ACK] Seq=1 Ack=425634 Win=65536 Len=0 TS
35	3.241521142	127.0.0.1	127.0.0.1	TCP	32807 [TCP Retransmission] 55900 → 5336 [PSH, ACK] Seq=425634 Ack=1 Win=65536 Len=0 TS
36	3.881450729	127.0.0.1	127.0.0.1	TCP	32807 [TCP Retransmission] 55900 → 5336 [PSH, ACK] Seq=425634 Ack=1 Win=65536 Len=0 TS
37	3.881489261	127.0.0.1	127.0.0.1	TCP	78 5336 → 55900 [ACK] Seq=1 Ack=523884 Win=1375232 Len=0 TS
29	3.881489261	127.0.0.1	127.0.0.1	TCP	174 55900 → 5336 [ACK] Seq=523884 Ack=1 Win=65536 Len=0 TS

Here we receive the firs file (where push, ack size the file 1mb)

and there more lost and fast retransmission (Because we change the cc algorithm).

55	3.938855646	127.0.0.1	127.0.0.1	TCP	66 5336 → 55900 [ACK] Seq=22 Ack=1048577 Win=2684928 Len=0 TS
64	5.810145009	127.0.0.1	127.0.0.1	TCP	67 55900 → 5336 [PSH, ACK] Seq=1048577 Ack=22 Win=65536 Len=0 TS
65	5.810160920	127.0.0.1	127.0.0.1	TCP	66 5336 → 55900 [ACK] Seq=22 Ack=1048578 Win=2684928 Len=0 TS
66	5.8101862188	127.0.0.1	127.0.0.1	TCP	65549 [TCP Previous segment not captured] 55900 → 5336 [ACK] Seq=1048578
67	5.8101884464	127.0.0.1	127.0.0.1	TCP	65549 55900 → 5336 [ACK] Seq=1179544 Ack=22 Win=65536 Len=65483
68	5.81018515561	127.0.0.1	127.0.0.1	TCP	78 [TCP Window Update] 5336 → 55900 [ACK] Seq=22 Ack=1048578
69	5.81018461363	127.0.0.1	127.0.0.1	TCP	65549 55900 → 5336 [ACK] Seq=1245027 Ack=22 Win=65536 Len=65483
70	5.810171055	127.0.0.1	127.0.0.1	TCP	78 [TCP Window Update] 5336 → 55900 [ACK] Seq=22 Ack=1048578
71	5.81019492542	127.0.0.1	127.0.0.1	TCP	65549 55900 → 5336 [ACK] Seq=1310510 Ack=22 Win=65536 Len=65483
72	5.8101500669	127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 65#1] 5336 → 55900 [ACK] Seq=22 Ack=1048578
73	5.8101517656	127.0.0.1	127.0.0.1	TCP	65549 55900 → 5336 [ACK] Seq=1375993 Ack=22 Win=65536 Len=65483
74	5.8101527440	127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 65#2] 5336 → 55900 [ACK] Seq=22 Ack=1048578
75	5.8101542924	127.0.0.1	127.0.0.1	TCP	65549 55900 → 5336 [ACK] Seq=1441476 Ack=22 Win=65536 Len=65483
76	5.810155636	127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 65#3] 5336 → 55900 [ACK] Seq=22 Ack=1048578
77	5.8101566984	127.0.0.1	127.0.0.1	TCP	65549 55900 → 5336 [ACK] Seq=1566959 Ack=22 Win=65536 Len=65483
78	5.8101574272	127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 65#4] 5336 → 55900 [ACK] Seq=22 Ack=1048578
79	5.8101592263	127.0.0.1	127.0.0.1	TCP	65549 [TCP Fast Retransmission] 55900 → 5336 [ACK] Seq=1048578
80	5.8101606231	127.0.0.1	127.0.0.1	TCP	66 5336 → 55900 [ACK] Seq=22 Ack=1572442 Win=2881280 Len=0 TS
81	5.8101616802	127.0.0.1	127.0.0.1	TCP	400 55900 → 5336 [ACK] Seq=1572442 Ack=22 Win=65536 Len=0 TS

In the final:-

283	15.1118767295	127.0.0.1	127.0.0.1	TCP	66 5336 → 55900 [ACK] Seq=169 Ack=8388192 Win=3012224 Len=0 TS
284	15.345365150	127.0.0.1	127.0.0.1	TCP	498 55900 → 5336 [PSH, ACK] Seq=8388192 Ack=169 Win=65536 Len=0 TS
285	15.345410078	127.0.0.1	127.0.0.1	TCP	66 5336 → 55900 [ACK] Seq=169 Ack=8388616 Win=3144960 Len=0 TS
286	17.822655045	127.0.0.1	127.0.0.1	TCP	67 55900 → 5336 [PSH, ACK] Seq=8388616 Ack=169 Win=65536 Len=0 TS
287	17.822659716	127.0.0.1	127.0.0.1	TCP	66 5336 → 55900 [ACK] Seq=169 Ack=8388617 Win=3145088 Len=0 TS
288	17.822676583	127.0.0.1	127.0.0.1	TCP	66 55900 → 5336 [FIN, ACK] Seq=8388617 Ack=169 Win=65536 Len=0 TS
289	17.822880435	127.0.0.1	127.0.0.1	TCP	66 5336 → 55900 [FIN, ACK] Seq=169 Ack=8388618 Win=3145728 Len=0 TS
290	17.822887086	127.0.0.1	127.0.0.1	TCP	66 55900 → 5336 [ACK] Seq=8388618 Ack=170 Win=65536 Len=0 TS

End.

**The first work with 15% loss data , 5 times send the file.**

**The sends:-**

```
codebind@codebind:~/CLionProjects/Hw3_TCC$ ./sender
client connected to server!
RUN 1
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 2
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 3
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 4
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
Sender send the exit message for the receiver!!
RUN 5
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
0
*****
```

```
Sender send the exit message for the receiver!!
codebind@codebind:~/CLionProjects/Hw3_TCC$ █
```

The receivers:-

NOTE (Where write \*\*in\*) this is mean the authentication sent  
sorry in the final i'm noted it!.

```
codebind@codebind:~/CloudProjects/Hw3_TCC$ sudo tc qdisc add dev lo root netem loss 15%
[sudo] password for codebind:
codebind@codebind:~/CloudProjects/Hw3_TCC$ make all
make: Nothing to be done for 'all'.
codebind@codebind:~/CloudProjects/Hw3_TCC$ ./receiver
Waiting for incoming connections
the server is ready!
Receive 1
client number connection accepted
2      65482
2      98223
2      130964
2      458374
2      523883
2      524288
**in*
Time in microseconds: 0.211847 microseconds
size2==524288
2      524288
Switching to Reno algorithm
Time in microseconds: 0.0002319 microseconds
size2==1048576
All the file is recv done
*****
```

```
Receive 2
2      65483
2      524288
**in*
Time in microseconds: 0.000811 microseconds

size2==524288

2      196449
2      327415
2      524288

Switching to Reno algorithm
Time in microseconds: 0.000200 microseconds

size2==1048576

All the file is recv done
*****
```

```
Receive 3
2      65483
2      196449
2      524288
**in*
Time in microseconds: 0.000061 microseconds

size2==524288

2      196449
2      261932
2      524288

Switching to Reno algorithm
Time in microseconds: 0.047957 microseconds

size2==1048576

All the file is recv done
*****
```

```
Receive 4
2      65483
2      130966
2      196449
2      524288
**in*
Time in microseconds: 0.000339 microseconds

size2==524288

2      130966
2      196449
2      524288

Switching to Reno algorithm
Time in microseconds: 0.000451 microseconds

size2==1048576

All the file is recv done
*****
```

```

Receive 5
2      261932
2      524288
**in*
Time in microseconds: 0.000057 microseconds

size2==524288

2      130966
2      327415
2      458381
2      524288

Switching to Reno algorithm
Time in microseconds: 0.014783 microseconds

size2==1048576

All the file is recv done

*****

```

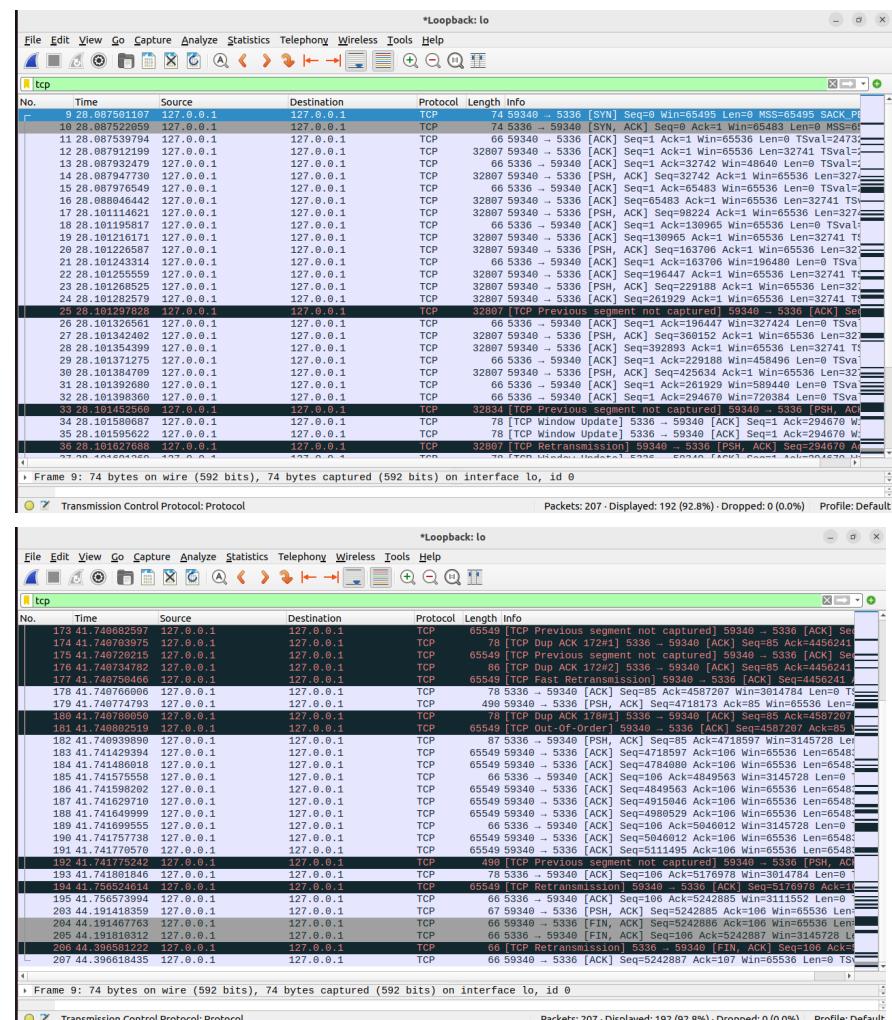
Times Summary:  
Part1:-  
0.211847  
0.000811  
0.000061  
0.000339  
0.000057  
Part2:-  
0.002319  
0.000200  
0.047957  
0.000451  
0.014783

\*\*\*\*\*  
Averages:  
Part1: 0.042623  
Averages:  
Part2: 0.013142

END  
codebind@codebind:~/CLionProjects/Hw3\_TCC\$

## Know Working with wireshark:-

Two pictures public(firs ,end).



Here we see, the lost od data, and then retransmission data

No.	Time	Source	Destination	Protocol	Length Info
32	28.101398360	127.0.0.1	127.0.0.1	TCP	66 5336 → 59340 [ACK] Seq=1 Ack=294670 Win=720384 Len=0 TSv4
33	28.101452560	127.0.0.1	127.0.0.1	TCP	32834 [TCP Previous segment not captured] 59340 → 5336 [PSH, ACK]
34	28.101589687	127.0.0.1	127.0.0.1	TCP	78 [TCP Window Update] 5336 → 59340 [ACK] Seq=1 Ack=294670 Win=65536 Len=405
35	28.101595622	127.0.0.1	127.0.0.1	TCP	78 [TCP Window Update] 5336 → 59340 [ACK] Seq=1 Ack=294670 Win=65536 Len=405
36	28.101627688	127.0.0.1	127.0.0.1	TCP	32887 [TCP Retransmission] 59340 → 5336 [PSH, ACK] Seq=294670 Ack=1
37	28.101691260	127.0.0.1	127.0.0.1	TCP	78 [TCP Window Update] 5336 → 59340 [ACK] Seq=1 Ack=294670 Win=65536 Len=405
38	28.101702867	127.0.0.1	127.0.0.1	TCP	86 [TCP Window Update] 5336 → 59340 [ACK] Seq=1 Ack=294670 Win=65536 Len=405
39	28.101717407	127.0.0.1	127.0.0.1	TCP	78 5336 → 59340 [ACK] Seq=1 Ack=458375 Win=1506176 Len=0 TSv4
40	28.309024729	127.0.0.1	127.0.0.1	TCP	32887 [TCP Retransmission] 59340 → 5336 [ACK] Seq=458375 Ack=1
41	28.309160085	127.0.0.1	127.0.0.1	TCP	66 5336 → 59340 [ACK] Seq=1 Ack=523884 Win=1637120 Len=0 TSv4
42	28.520947273	127.0.0.1	127.0.0.1	TCP	471 59340 → 5336 [PSH, ACK] Seq=523884 Ack=1 Win=65536 Len=405
43	28.521089597	127.0.0.1	127.0.0.1	TCP	66 5336 → 59340 [ACK] Seq=1 Ack=524289 Win=1768064 Len=0 TSv4
44	28.521268098	127.0.0.1	127.0.0.1	TCP	87 5336 → 59340 [PSH, ACK] Seq=1 Ack=524289 Win=1768064 Len=222
45	28.521282132	127.0.0.1	127.0.0.1	TCP	66 59340 → 5336 [ACK] Seq=524289 Ack=22 Win=65536 Len=0 TSv4
46	28.522073968	127.0.0.1	127.0.0.1	TCP	65549 59340 → 5336 [ACK] Seq=524289 Ack=22 Win=65536 Len=65483
47	28.522177668	127.0.0.1	127.0.0.1	TCP	65549 59340 → 5336 [ACK] Seq=589772 Ack=22 Win=65536 Len=65483
48	28.522234975	127.0.0.1	127.0.0.1	TCP	66 5336 → 59340 [ACK] Seq=22 Ack=655255 Win=2030080 Len=0 TSv4
49	28.522474171	127.0.0.1	127.0.0.1	TCP	65549 [TCP Previous segment not captured] 59340 → 5336 [ACK] Seq=22 Ack=655255
50	28.522660297	127.0.0.1	127.0.0.1	TCP	78 [TCP Window Update] 5336 → 59340 [ACK] Seq=22 Ack=655255
51	28.522704628	127.0.0.1	127.0.0.1	TCP	65549 59340 → 5336 [ACK] Seq=786221 Ack=22 Win=65536 Len=65483
52	28.522742477	127.0.0.1	127.0.0.1	TCP	78 [TCP Window Update] 5336 → 59340 [ACK] Seq=22 Ack=655255
53	28.522893414	127.0.0.1	127.0.0.1	TCP	65549 [TCP Previous segment not captured] 59340 → 5336 [ACK] Seq=22 Ack=655255
54	28.522962892	127.0.0.1	127.0.0.1	TCP	86 [TCP Window Update] 5336 → 59340 [ACK] Seq=22 Ack=655255
55	28.522964336	127.0.0.1	127.0.0.1	TCP	65549 [TCP Retransmission] 59340 → 5336 [ACK] Seq=655255 Ack=222
56	28.523091815	127.0.0.1	127.0.0.1	TCP	78 5336 → 59340 [ACK] Seq=22 Ack=851704 Win=2553856 Len=0 TSv4
57	28.523039521	127.0.0.1	127.0.0.1	TCP	65549 [TCP Retransmission] 59340 → 5336 [ACK] Seq=851704 Ack=222
58	28.523068840	127.0.0.1	127.0.0.1	TCP	66 5336 → 59340 [ACK] Seq=22 Ack=982670 Win=2684928 Len=0 TSv4
59	28.523098814	127.0.0.1	127.0.0.1	TCP	490 [TCP Previous segment not captured] 59340 → 5336 [PSH, ACK] Seq=22 Ack=982670
60	28.523100327	127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 59#1] 5326 → 5326 [ACK] Seq=22 Ack=982670

Then more loss data see, but with fast retransmission data . because we change the algorithm.

No.	Time	Source	Destination	Protocol	Length Info
70	32.678793978	127.0.0.1	127.0.0.1	TCP	65549 59340 → 5336 [ACK] Seq=1310510 Ack=22 Win=65536 Len=65483
71	32.678807322	127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 65#1] 5336 → 59340 [ACK] Seq=22 Ack=1114661
72	32.678837407	127.0.0.1	127.0.0.1	TCP	65549 59340 → 5336 [ACK] Seq=1375993 Ack=22 Win=65536 Len=65483
73	32.679054524	127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 65#2] 5336 → 59340 [ACK] Seq=22 Ack=1114661
74	32.679076502	127.0.0.1	127.0.0.1	TCP	65549 59340 → 5336 [ACK] Seq=1441476 Ack=22 Win=65536 Len=65483
75	32.679207197	127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 65#3] 5336 → 59340 [ACK] Seq=22 Ack=1114661
76	32.679227039	127.0.0.1	127.0.0.1	TCP	65549 [TCP Fast Retransmission] 59340 → 5336 [ACK] Seq=1114661
77	32.679254537	127.0.0.1	127.0.0.1	TCP	66 5336 → 59340 [ACK] Seq=22 Ack=1506959 Win=2948096 Len=0 TSv4
78	32.679273409	127.0.0.1	127.0.0.1	TCP	65549 59340 → 5336 [ACK] Seq=1566959 Ack=22 Win=65536 Len=65483
79	32.679276475	127.0.0.1	127.0.0.1	TCP	490 59340 → 5336 [PSH, ACK] Seq=1572442 Ack=22 Win=65536 Len=405
80	32.679422822	127.0.0.1	127.0.0.1	TCP	66 5336 → 59340 [ACK] Seq=22 Ack=1572866 Win=3112448 Len=0 TSv4
81	32.679466071	127.0.0.1	127.0.0.1	TCP	87 5336 → 59340 [PSH, ACK] Seq=22 Ack=1572866 Win=3112448 Len=0 TSv4
82	32.679791828	127.0.0.1	127.0.0.1	TCP	65549 59340 → 5336 [ACK] Seq=1572866 Ack=43 Win=65536 Len=65483
83	32.679849591	127.0.0.1	127.0.0.1	TCP	65549 59340 → 5336 [ACK] Seq=1638349 Ack=43 Win=65536 Len=65483
84	32.679885586	127.0.0.1	127.0.0.1	TCP	65549 59340 → 5336 [ACK] Seq=1703832 Ack=43 Win=65536 Len=65483
85	32.679917433	127.0.0.1	127.0.0.1	TCP	66 5336 → 59340 [ACK] Seq=43 Ack=1703832 Win=3112448 Len=0 TSv4
86	32.679991858	127.0.0.1	127.0.0.1	TCP	65549 59340 → 5336 [ACK] Seq=1769315 Ack=43 Win=65536 Len=65483
87	32.680003595	127.0.0.1	127.0.0.1	TCP	66 5336 → 59340 [ACK] Seq=43 Ack=1834798 Win=3112448 Len=0 TSv4
88	32.680024130	127.0.0.1	127.0.0.1	TCP	65549 59340 → 5336 [ACK] Seq=1834798 Ack=43 Win=65536 Len=65483
89	32.680048910	127.0.0.1	127.0.0.1	TCP	65549 [TCP Previous segment not captured] 59340 → 5336 [ACK] Seq=43 Ack=1834798
90	32.680062199	127.0.0.1	127.0.0.1	TCP	78 5336 → 59340 [ACK] Seq=43 Ack=1900281 Win=3112448 Len=0 TSv4
91	32.680073452	127.0.0.1	127.0.0.1	TCP	65549 59340 → 5336 [ACK] Seq=2031247 Ack=43 Win=65536 Len=65483
92	32.680084765	127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 99#1] 5336 → 59340 [ACK] Seq=43 Ack=1900281
93	32.680092862	127.0.0.1	127.0.0.1	TCP	490 59340 → 5336 [PSH, ACK] Seq=2096730 Ack=43 Win=65536 Len=405
94	32.680096125	127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 99#2] 5336 → 59340 [ACK] Seq=43 Ack=1900281
95	32.680111451	127.0.0.1	127.0.0.1	TCP	65549 [TCP Fast Retransmission] 59340 → 5336 [ACK] Seq=1900281
96	32.680130672	127.0.0.1	127.0.0.1	TCP	66 5336 → 59340 [ACK] Seq=43 Ack=2097154 Win=3040016 Len=0 TSv4
97	35.296149006	127.0.0.1	127.0.0.1	TCP	67 59340 → 5336 [PSH, ACK] Seq=2097154 Ack=43 Win=65536 Len=405
98	35.296149006	127.0.0.1	127.0.0.1	TCP	65540 59340 → 5336 [ACK] Seq=2097155 Ack=43 Win=65536 Len=405

In each [PSH,ACK] ,we get the file (1mb) that we send from the sender.

until we get the last, and FIN.

194	41.756524614	127.0.0.1	127.0.0.1	TCP	65549 [TCP Retransmission] 59340 → 5336 [ACK] Seq=5176978 Ack=1
195	41.756573994	127.0.0.1	127.0.0.1	TCP	66 5336 → 59340 [ACK] Seq=106 Ack=5242885 Win=3111552 Len=0
203	44.191418359	127.0.0.1	127.0.0.1	TCP	67 59340 → 5336 [PSH, ACK] Seq=5242885 Ack=106 Win=65536 Len=405
204	44.191467763	127.0.0.1	127.0.0.1	TCP	66 59340 → 5336 [FIN, ACK] Seq=5242886 Ack=106 Win=65536 Len=405
205	44.191467763	127.0.0.1	127.0.0.1	TCP	66 5336 → 59340 [ACK] Seq=2097155 Ack=43 Win=65536 Len=405

END 15% loss.

Know work with 20% loss data.

The send:-

```
codebind@codebind:~/CLionProjects/Hw3_TCC$ ./sender
client connected to server!
RUN 1
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****  
  
Sender send the exit message for the receiver!!
RUN 2
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****  
  
Sender send the exit message for the receiver!!
RUN 3
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****  
  
Sender send the exit message for the receiver!!
RUN 4
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
1
*****
```

```
1
*****
Sender send the exit message for the receiver!!
RUN 5
Send the first part of the file
524288
*
Check!
Change the cc algorithm to reno

client (0) Current CC, type: reno

Send the second part of the file

1048576
sent all the 1MB file: 1048576
User Decision:-
click 1 to continue ,0(Zero) to exit!
0
*****  
  
Sender send the exit message for the receiver!!
codebind@codebind:~/CLionProjects/Hw3_TCC$
```

The receives:-

```
codebind@codebind:~/ClionProjects/Hw3_TCC$ sudo tc qdisc change dev lo root netem loss 20%
[sudo] password for codebind:
codebind@codebind:~/ClionProjects/Hw3_TCC$ make all
make: Nothing to be done for 'all'.
codebind@codebind:~/ClionProjects/Hw3_TCC$ ./receiver
Waiting for incoming connections
the server is ready!

Receive 1

client number connection accepted
2          32741
2          65482
2          98223
2          130964
2          196446
2          261928
2          294669
2          327410
2          524288
**in*
Time in microseconds: 0.000219 microseconds

size2==524288

2          130966
2          524288

Switching to Reno algorithm
Time in microseconds: 0.002485 microseconds

size2==1048576

All the file is recv done
*****
```

```
Receive 3

2          523864
2          524288
**in*
Time in microseconds: 0.232419 microseconds

size2==524288

2          392898
2          524288

Switching to Reno algorithm
Time in microseconds: 0.264495 microseconds

size2==1048576

All the file is recv done
*****
```

```
Receive 4

2          196449
2          524288
**in*
Time in microseconds: 0.000389 microseconds

size2==524288

2          65483
2          130966
2          196449
2          392898
2          524288

Switching to Reno algorithm
Time in microseconds: 0.251993 microseconds

size2==1048576

All the file is recv done
*****
```

```
Receive 5
2      130966
2      523864
2      524288
**in*
Time in microseconds: 0.029954 microseconds

size2==524288

2      196449
2      524288

Switching to Reno algorithm
Time in microseconds: 0.211886 microseconds

size2==1048576

All the file is recv done

*****
```

Times Summary,Averages:-

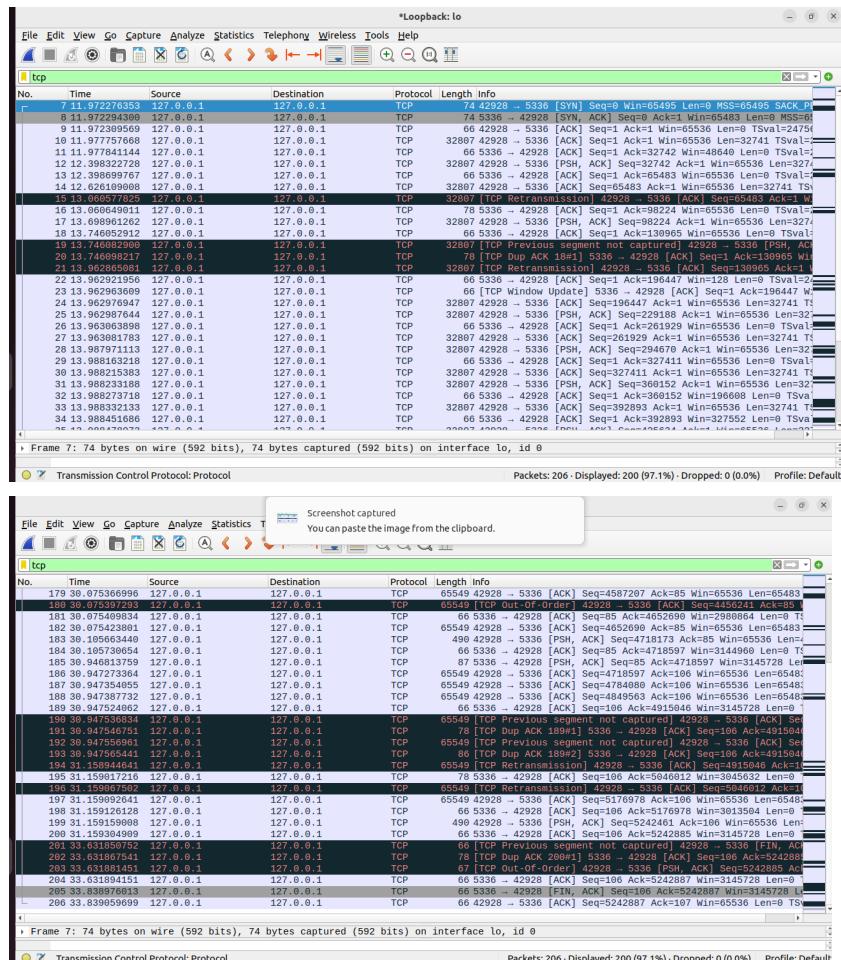
```
Times Summary:
Part1:-
0.000219
0.262124
0.232419
0.000389
0.029954
Part2:-
0.002485
0.275117
0.264495
0.251993
0.211886

*****
Averages:
Part1: 0.105021
Averages:
Part2: 0.201195

END
codebind@codebind:~/CLionProjects/Hw3_TCC$
```

Work with wireshark:-end).

Two pictures public (first,



Here see the retransmission

14 12.626109008 127.0.0.1	127.0.0.1	TCP	32807 42928 - 5336 [ACK]	Seq=65483 Ack=1 Win=65536 Len=32741 TSval=24754
15 13.060577825 127.0.0.1	127.0.0.1	TCP	32807 [TCP Retransmission]	42928 - 5336 [PSH, ACK]
16 13.060649011 127.0.0.1	127.0.0.1	TCP	78 5336 - 42928 [ACK]	Seq=1 Ack=98224 Win=65536 Len=0 TSval=24754
17 13.698961262 127.0.0.1	127.0.0.1	TCP	32807 42928 - 5336 [PSH, ACK]	Seq=89224 Ack=1 Win=65536 Len=32741 TSval=24754
18 13.746052912 127.0.0.1	127.0.0.1	TCP	66 5336 - 42928 [ACK]	Seq=1 Ack=130965 Win=65536 Len=0 TSval=24754
19 13.746082900 127.0.0.1	127.0.0.1	TCP	32807 [TCP Previous segment not captured]	42928 - 5336 [PSH, ACK]
20 13.746098217 127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 18#1] 5336 - 42928 [ACK]	Seq=1 Ack=130965 Win=65536 Len=0 TSval=24754
21 13.962865501 127.0.0.1	127.0.0.1	TCP	32807 [TCP Retransmission]	42928 - 5336 [ACK] Seq=130965 Ack=1
22 13.988011006 127.0.0.1	127.0.0.1	TCP	66 5336 - 42928 [ACK]	Seq=130965 Ack=1 Win=196447 Len=0 TSval=24754
23 13.989363609 127.0.0.1	127.0.0.1	TCP	32807 [TCP Window Update]	5336 - 42928 [ACK] Seq=130965 Ack=1 Win=196447 Len=0 TSval=24754
24 13.982376947 127.0.0.1	127.0.0.1	TCP	32807 42928 - 5336 [ACK] Seq=196447 Ack=1 Win=65536 Len=32741 T1	TSval=24754
25 13.982387644 127.0.0.1	127.0.0.1	TCP	32807 42928 - 5336 [PSH, ACK]	Seq=229188 Ack=1 Win=65536 Len=32741 T1
26 13.983081873 127.0.0.1	127.0.0.1	TCP	66 5336 - 42928 [ACK]	Seq=261929 Ack=1 Win=65536 Len=32741 T1
27 13.988111303 127.0.0.1	127.0.0.1	TCP	32807 42928 - 5336 [ACK]	Seq=261929 Ack=1 Win=65536 Len=32741 T1
28 13.988163508 127.0.0.1	127.0.0.1	TCP	66 5336 - 42928 [ACK]	Seq=274111 Ack=1 Win=65536 Len=32741 T1
29 13.988215533 127.0.0.1	127.0.0.1	TCP	32807 42928 - 5336 [ACK]	Seq=327411 Ack=1 Win=65536 Len=32741 T1
30 13.988223188 127.0.0.1	127.0.0.1	TCP	32807 42928 - 5336 [PSH, ACK]	Seq=360152 Ack=1 Win=65536 Len=32741 T1
31 13.988273718 127.0.0.1	127.0.0.1	TCP	66 5336 - 42928 [ACK]	Seq=1 Ack=360152 Win=196608 Len=0 TSval=24754
32 13.988332133 127.0.0.1	127.0.0.1	TCP	32807 42928 - 5336 [ACK]	Seq=392893 Ack=1 Win=65536 Len=32741 T1
33 13.988451680 127.0.0.1	127.0.0.1	TCP	66 5336 - 42928 [ACK]	Seq=1 Ack=392893 Win=327552 Len=0 TSval=24754
34 13.988451680 127.0.0.1	127.0.0.1	TCP	32807 42928 - 5336 [ACK]	Seq=1 Ack=392893 Win=327552 Len=0 TSval=24754

more loss data and retransmission ,there more and more (because it's 20%).

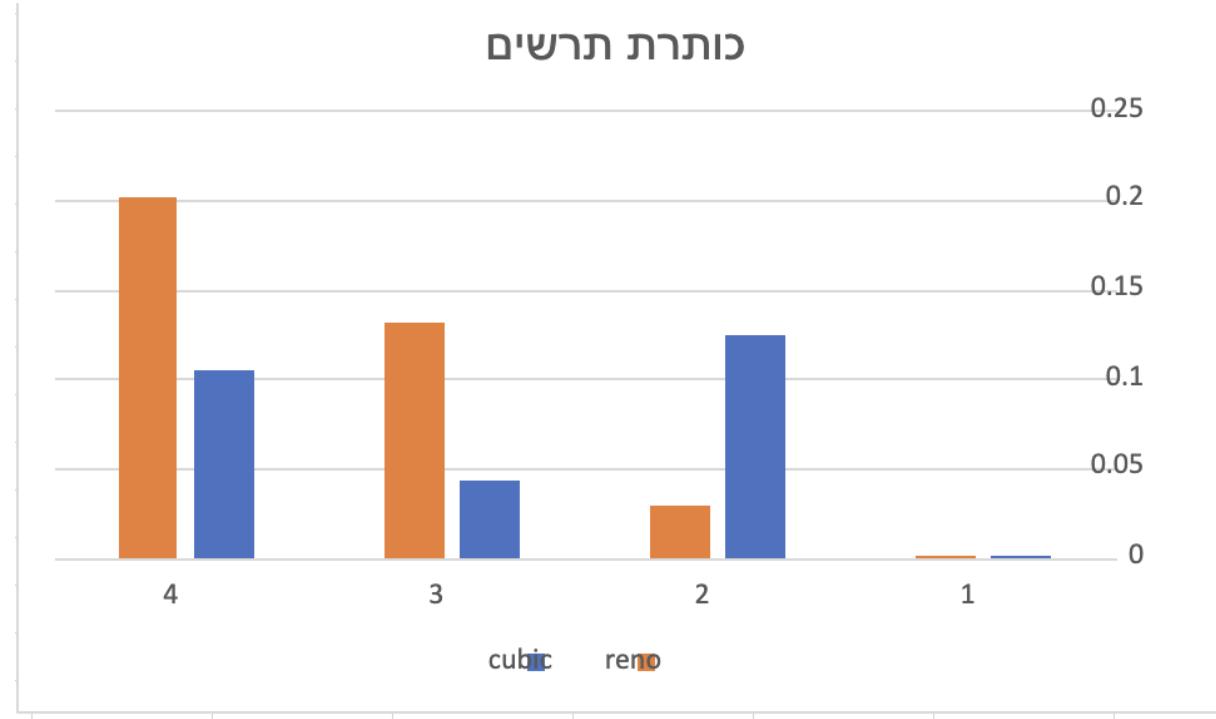
37 13.900533740 127.0.0.1	127.0.0.1	TCP	66 5336 - 42928 [ACK]	Seq=1 Ack=458375 Win=196447 Len=0 TSval=24754
38 13.988555909 127.0.0.1	127.0.0.1	TCP	32807 [TCP Previous segment not captured]	42928 - 5336 [PSH, ACK]
39 13.988578844 127.0.0.1	127.0.0.1	TCP	78 [TCP Window Update]	5336 - 42928 [ACK] Seq=1 Ack=458375 Win=196447 Len=0 TSval=24754
40 13.988611589 127.0.0.1	127.0.0.1	TCP	32807 [TCP Retransmission]	42928 - 5336 [ACK] Seq=458375 Ack=1
41 13.988639519 127.0.0.1	127.0.0.1	TCP	66 5336 - 42928 [ACK]	Seq=1 Ack=458375 Win=196447 Len=0 TSval=24754
42 13.988655658 127.0.0.1	127.0.0.1	TCP	498 42928 - 5336 [PSH, ACK]	Seq=523857 Ack=1 Win=65536 Len=43
43 13.988666867 127.0.0.1	127.0.0.1	TCP	66 5336 - 42928 [ACK]	Seq=524289 Win=982400 Len=0 TSval=24754
44 16.09103373 127.0.0.1	127.0.0.1	TCP	87 5336 - 42928 [PSH, ACK]	Seq=1 Ack=524289 Win=982400 Len=2
45 16.099328330 127.0.0.1	127.0.0.1	TCP	66 42928 - 5336 [ACK]	Seq=524289 Ack=22 Win=65536 Len=0 TSval=24754
46 16.102793630 127.0.0.1	127.0.0.1	TCP	65549 42928 - 5336 [ACK]	Seq=524289 Ack=22 Win=65536 Len=65483
47 16.102944997 127.0.0.1	127.0.0.1	TCP	65549 42928 - 5336 [ACK]	Seq=589772 Ack=22 Win=65536 Len=65483
48 16.102998630 127.0.0.1	127.0.0.1	TCP	66 5336 - 42928 [ACK]	Seq=22 Ack=655255 Win=1244288 Len=0 TSval=24754
49 16.103292137 127.0.0.1	127.0.0.1	TCP	65549 [TCP Previous segment not captured]	42928 - 5336 [ACK] Seq=1 Ack=196447 Win=128 Len=0 TSval=24754
50 16.103320533 127.0.0.1	127.0.0.1	TCP	78 [TCP Window Update]	5336 - 42928 [ACK] Seq=22 Ack=655255 Win=196447 Len=0 TSval=24754
51 16.103489761 127.0.0.1	127.0.0.1	TCP	65549 42928 - 5336 [ACK]	Seq=786221 Ack=22 Win=65536 Len=65483
52 16.103537006 127.0.0.1	127.0.0.1	TCP	78 [TCP Window Update]	5336 - 42928 [ACK] Seq=22 Ack=655255 Win=196447 Len=0 TSval=24754
53 16.103583742 127.0.0.1	127.0.0.1	TCP	65549 42928 - 5336 [ACK]	Seq=851704 Ack=22 Win=65536 Len=65483
54 16.103652464 127.0.0.1	127.0.0.1	TCP	65549 42928 - 5336 [ACK]	Seq=917187 Ack=22 Win=65536 Len=65483
55 16.104012185 127.0.0.1	127.0.0.1	TCP	78 [TCP Window Update]	5336 - 42928 [ACK] Seq=22 Ack=655255 Win=1244288 Len=0 TSval=24754
56 16.104063997 127.0.0.1	127.0.0.1	TCP	65549 [TCP Retransmission]	42928 - 5336 [ACK] Seq=655255 Ack=22
57 16.104136648 127.0.0.1	127.0.0.1	TCP	66 5336 - 42928 [ACK]	Seq=22 Ack=982670 Win=1899136 Len=0 TSval=24754
58 16.104157054 127.0.0.1	127.0.0.1	TCP	490 [TCP Previous segment not captured]	42928 - 5336 [PSH, ACK]
59 16.104169705 127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 57#1] 5336 - 42928 [ACK]	Seq=22 Ack=982670 Win=1899136 Len=0 TSval=24754
60 16.1042021674 127.0.0.1	127.0.0.1	TCP	65549 [TCP Retransmission]	42928 - 5336 [PSH, ACK] Seq=982670 Ack=22
61 16.104236451 127.0.0.1	127.0.0.1	TCP	66 5336 - 42928 [ACK]	Seq=22 Ack=1048577 Win=2030080 Len=0 TSval=24754
62 20.252609237 127.0.0.1	127.0.0.1	TCP	67 42928 - 5336 [PSH, ACK]	Seq=1048577 Ack=22 Win=65536 Len=0 TSval=24754

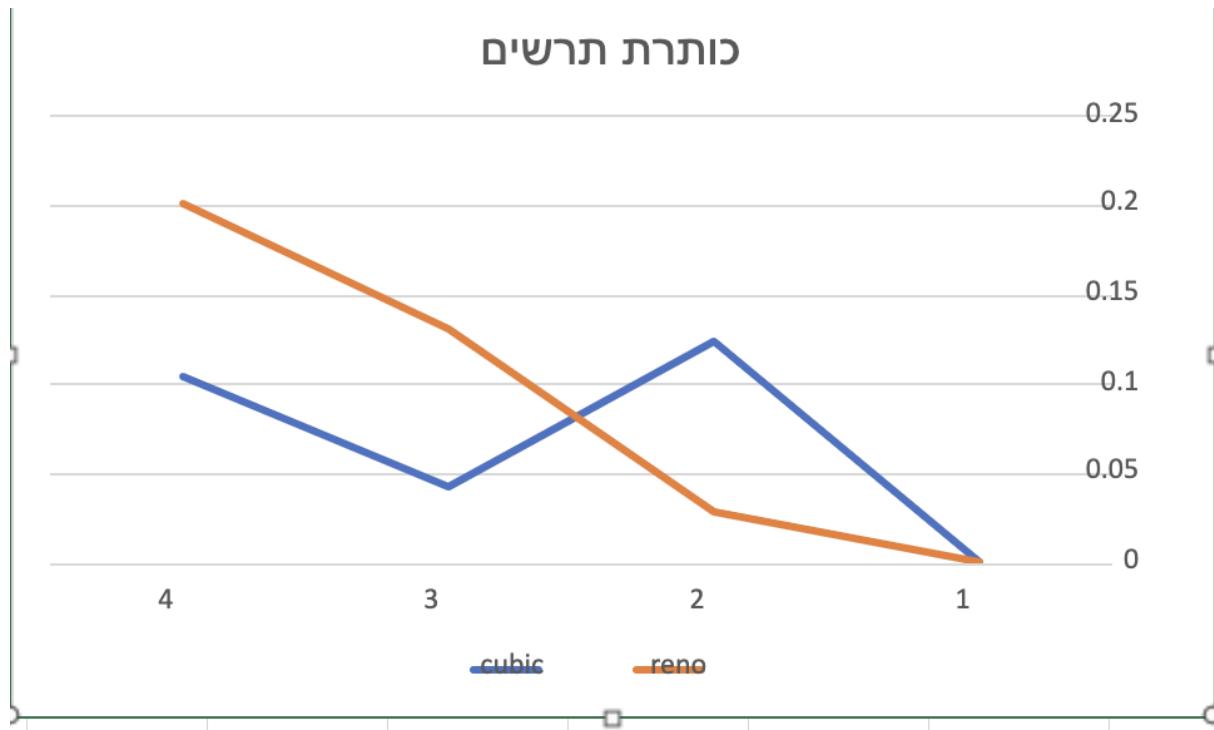
159 27.354881583	127.0.0.1	127.0.0.1	TCP	66 5336 → 42928 [ACK] Seq=3866469 Ack=85 Win=65536 Len=65483
156 27.354758975	127.0.0.1	127.0.0.1	TCP	65549 42928 → 5336 [ACK] Seq=3931952 Ack=85 Win=65536 Len=65483
157 27.354814007	127.0.0.1	127.0.0.1	TCP	65549 42928 → 5336 [ACK] Seq=3931952 Ack=85 Win=65536 Len=65483
158 27.354857319	127.0.0.1	127.0.0.1	TCP	66 5336 → 42928 [ACK] Seq=85 Ack=3997435 Win=3079040 Len=0 TS
159 27.354881583	127.0.0.1	127.0.0.1	TCP	65549 [TCP Previous segment not captured] 42928 → 5336 [ACK] Seq=85 Ack=85 Win=65536 Len=65483
160 27.354883643	127.0.0.1	127.0.0.1	TCP	490 42928 → 5336 [PSH, ACK] Seq=4193884 Ack=85 Win=65536 Len=4
161 27.354894695	127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 158#1] 5336 → 42928 [ACK] Seq=85 Ack=3997435 Win=65536 Len=0 TS
162 27.354897465	127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 158#2] 5336 → 42928 [ACK] Seq=85 Ack=3997435 Win=65536 Len=0 TS
163 27.354938707	127.0.0.1	127.0.0.1	TCP	65549 [TCP Fast Retransmission] 42928 → 5336 [ACK] Seq=3997435 Ack=85 Win=3044736 Len=0 TS
164 27.354980105	127.0.0.1	127.0.0.1	TCP	78 5336 → 42928 [ACK] Seq=85 Ack=4062918 Win=3044736 Len=0 TS
165 27.666659688	127.0.0.1	127.0.0.1	TCP	65549 [TCP Retransmission] 42928 → 5336 [ACK] Seq=4062918 Ack=85 Win=3044736 Len=0 TS
166 27.666877708	127.0.0.1	127.0.0.1	TCP	66 5336 → 42928 [ACK] Seq=85 Ack=4194308 Win=3078144 Len=0 TS
167 30.074795400	127.0.0.1	127.0.0.1	TCP	67 42928 → 5336 [PSH, ACK] Seq=4194308 Ack=85 Win=65536 Len=0 TS
168 30.074826631	127.0.0.1	127.0.0.1	TCP	66 5336 → 42928 [ACK] Seq=85 Ack=4194309 Win=3145088 Len=0 TS
169 30.075084273	127.0.0.1	127.0.0.1	TCP	65549 42928 → 5336 [ACK] Seq=4194309 Ack=85 Win=65536 Len=65483
170 30.075104517	127.0.0.1	127.0.0.1	TCP	66 5336 → 42928 [ACK] Seq=85 Ack=4259792 Win=3112448 Len=0 TS
171 30.075162256	127.0.0.1	127.0.0.1	TCP	65549 42928 → 5336 [ACK] Seq=4259792 Ack=85 Win=65536 Len=65483
172 30.075187675	127.0.0.1	127.0.0.1	TCP	66 5336 → 42928 [ACK] Seq=85 Ack=4325275 Win=3079680 Len=0 TS
173 30.075282372	127.0.0.1	127.0.0.1	TCP	65549 42928 → 5336 [ACK] Seq=4325275 Ack=85 Win=65536 Len=65483
174 30.075292973	127.0.0.1	127.0.0.1	TCP	66 5336 → 42928 [ACK] Seq=85 Ack=4390758 Win=3112448 Len=0 TS
175 30.075311868	127.0.0.1	127.0.0.1	TCP	65549 42928 → 5336 [ACK] Seq=4398758 Ack=85 Win=65536 Len=65483
176 30.075322063	127.0.0.1	127.0.0.1	TCP	66 5336 → 42928 [ACK] Seq=85 Ack=4456241 Win=3079040 Len=0 TS
177 30.075339638	127.0.0.1	127.0.0.1	TCP	65549 [TCP Previous segment not captured] 42928 → 5336 [ACK] Seq=85 Ack=85 Win=3079040 Len=0 TS
178 30.075353315	127.0.0.1	127.0.0.1	TCP	78 [TCP Dup ACK 176#1] 5336 → 42928 [ACK] Seq=85 Ack=4456241 Win=3079040 Len=0 TS
179 30.075366996	127.0.0.1	127.0.0.1	TCP	65549 42928 → 5336 [ACK] Seq=4587207 Ack=85 Win=65536 Len=65483
180 30.075397293	127.0.0.1	127.0.0.1	TCP	65549 [TCP Out-of-Order] 42928 → 5336 [ACK] Seq=4456241 Ack=85 Win=65536 Len=65483
181 30.075409834	127.0.0.1	127.0.0.1	TCP	66 5336 → 42928 [ACK] Seq=85 Ack=4652690 Win=2980864 Len=0 TS

until get the final:-

203 33.631881451	127.0.0.1	127.0.0.1	TCP	67 [TCP Out-of-Order] 42928 → 5336 [PSH, ACK] Seq=5242885 Ack=85 Win=65536 Len=65483
204 33.631894151	127.0.0.1	127.0.0.1	TCP	66 5336 → 42928 [ACK] Seq=106 Ack=5242887 Win=3145728 Len=0 TS
205 33.838976013	127.0.0.1	127.0.0.1	TCP	66 5336 → 42928 [FIN, ACK] Seq=106 Ack=5242887 Win=3145728 Len=0 TS
206 33.839059699	127.0.0.1	127.0.0.1	TCP	66 42928 → 5336 [ACK] Seq=5242887 Ack=107 Win=65536 Len=0 TS

## Conclusion:-





The cubic algorithm it's faster,I thought that the reason is default cc algorithm.