

Analysis of Test 1: The hw3 tcpdump results as well as client/server tcpdump results.

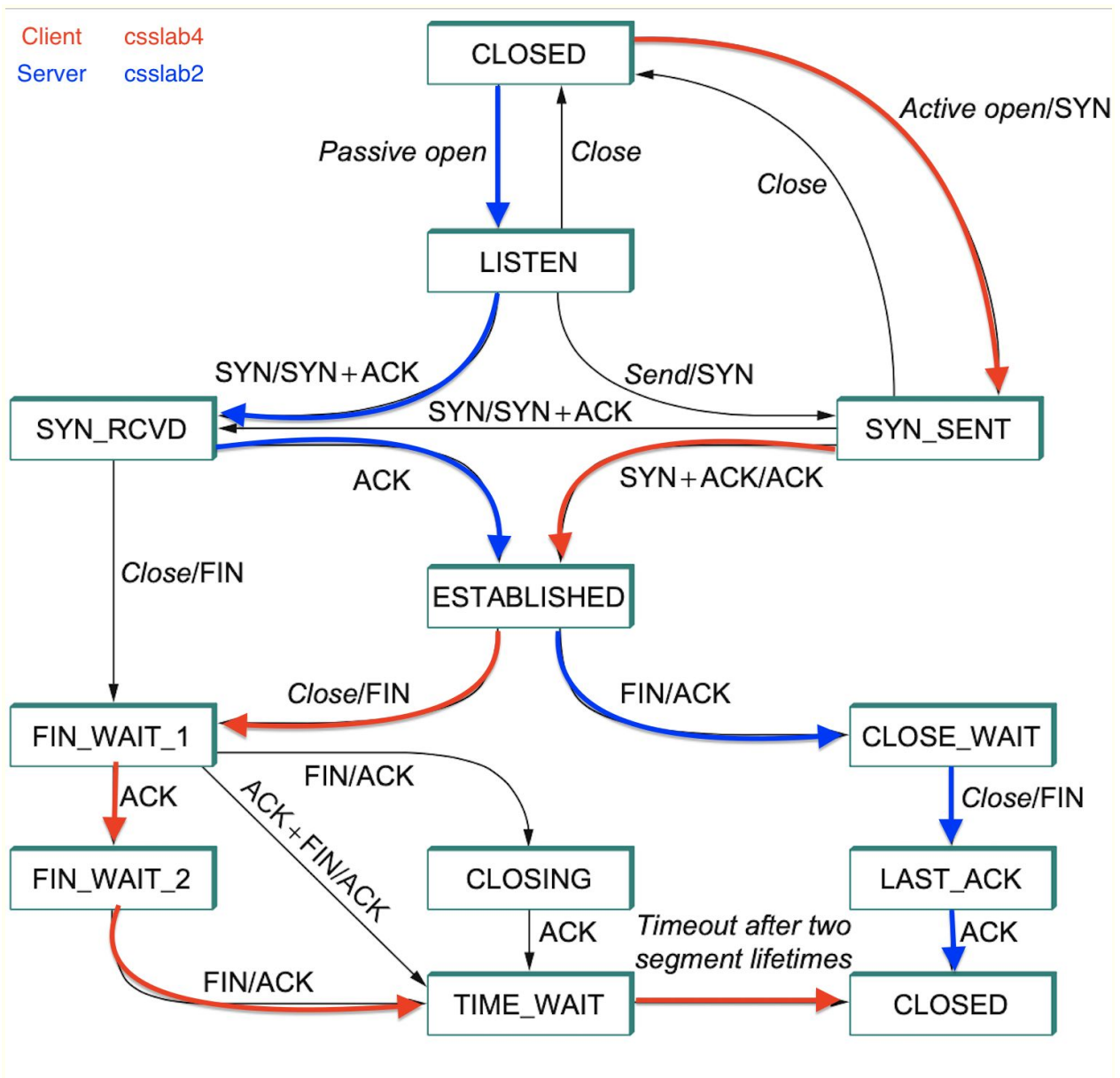
Results from hw3 tcpdump: Client Server

```
[youngjon@csslab2 hw3]$ tcpdump -vtt host csslab4 and port 58101 and tcp
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
1589521518.721468 IP (tos 0x0, ttl 64, id 15777, offset 0, flags [DF], proto TCP (6), length 60)
    csslab4.umb.edu.58852 > csslab2.umb.edu.58101: Flags [S], cksum 0xf487 (correct), seq 724949433, win 26880, options [mss 8960,sackOK,TS val 3307791384 ecr 0,nop,wscale 7], length 0
1589521518.721537 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 60)
    csslab2.umb.edu.58101 > csslab4.umb.edu.58852: Flags [S.], cksum 0x758f (incorrect -> 0x6961), seq 3116596988, ack 724949434, win 26844, options [mss 8960,sackOK,TS val 2627712473 ecr 3307791384,nop,wscale 7], length 0
1589521518.721794 IP (tos 0x0, ttl 64, id 15778, offset 0, flags [DF], proto TCP (6), length 52)
    csslab4.umb.edu.58852 > csslab2.umb.edu.58101: Flags [.], cksum 0xd83 (correct), ack 1, win 210, options [nop,nop,TS val 3307791385 ecr 2627712473], length 0
1589521518.721812 IP (tos 0x0, ttl 64, id 15779, offset 0, flags [DF], proto TCP (6), length 62)
    csslab4.umb.edu.58852 > csslab2.umb.edu.58101: Flags [P.], cksum 0xd71 (correct), seq 1:11, ack 1, win 210, options [nop,nop,TS val 3307791385 ecr 2627712473], length 10
1589521518.721817 IP (tos 0x0, ttl 64, id 52084, offset 0, flags [DF], proto TCP (6), length 52)
    csslab2.umb.edu.58101 > csslab4.umb.edu.58852: Flags [.], cksum 0x7587 (incorrect -> 0xd79), ack 11, win 210, options [nop,nop,TS val 2627712473 ecr 3307791385], length 0
1589521518.721877 IP (tos 0x0, ttl 64, id 52085, offset 0, flags [DF], proto TCP (6), length 62)
    csslab2.umb.edu.58101 > csslab4.umb.edu.58852: Flags [P.], cksum 0x7591 (incorrect -> 0xd67), seq 1:11, ack 11, win 210, options [nop,nop,TS val 2627712473 ecr 3307791385], length 10
1589521518.721889 IP (tos 0x0, ttl 64, id 52086, offset 0, flags [DF], proto TCP (6), length 52)
    csslab2.umb.edu.58101 > csslab4.umb.edu.58852: Flags [F.], cksum 0x7587 (incorrect -> 0x1d6e), seq 11, ack 11, win 210, options [nop,nop,TS val 2627712473 ecr 3307791385], length 0
1589521518.722008 IP (tos 0x0, ttl 64, id 15780, offset 0, flags [DF], proto TCP (6), length 52)
    csslab4.umb.edu.58852 > csslab2.umb.edu.58101: Flags [.], cksum 0xd6f (correct), ack 11, win 210, options [nop,nop,TS val 3307791385 ecr 2627712473], length 0
1589521518.722016 IP (tos 0x0, ttl 64, id 15781, offset 0, flags [DF], proto TCP (6), length 1582)
    csslab4.umb.edu.58852 > csslab2.umb.edu.58101: Flags [P.], cksum 0xc377 (correct), seq 11:1461, ack 12, win 210, options [nop,nop,TS val 3307791385 ecr 2627712473], length 1450
1589521518.722020 IP (tos 0x0, ttl 64, id 15782, offset 0, flags [DF], proto TCP (6), length 52)
    csslab4.umb.edu.58852 > csslab2.umb.edu.58101: Flags [F.], cksum 0x17c3 (correct), seq 1461, ack 12, win 210, options [nop,nop,TS val 3307791385 ecr 2627712473], length 0
1589521518.722031 IP (tos 0x0, ttl 64, id 52087, offset 0, flags [DF], proto TCP (6), length 52)
    csslab2.umb.edu.58101 > csslab4.umb.edu.58852: Flags [.], cksum 0x7587 (incorrect -> 0x17ac), ack 1462, win 233, options [nop,nop,TS val 2627712473 ecr 3307791385], length 0
^C
11 packets captured
19 packets received by filter
0 packets dropped by kernel
[younjon@csslab2 hw3]$
```

Results from personal tcpdump: Client Server

```
[youngjon@csslab2 prog3]$ tcpdump -vtt host csslab4 and port 58101 and tcp
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
1589526677.580112 IP (tos 0x0, ttl 64, id 23321, offset 0, flags [DF], proto TCP (6), length 60)
    csslab4.umb.edu.58944 > csslab2.umb.edu.58101: Flags [S], cksum 0x18ea (correct), seq 1742520458, win 26880, options [mss 8960,sackOK,TS val 3312950163 ecr 0,nop,wscale 7], length 0
1589526677.580248 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 60)
    csslab2.umb.edu.58101 > csslab4.umb.edu.58944: Flags [S.], cksum 0x758f (incorrect -> 0x0f18), seq 1476765596, ack 1742520459, win 26844, options [mss 8960,sackOK,TS val 2632871252 ecr 3312950163,nop,wscale 7], length 0
1589526677.580516 IP (tos 0x0, ttl 64, id 23322, offset 0, flags [DF], proto TCP (6), length 52)
    csslab4.umb.edu.58944 > csslab2.umb.edu.58101: Flags [.], cksum 0xc33a (correct), ack 1, win 210, options [nop,nop,TS val 3312950163 ecr 2632871252], length 0
1589526677.580560 IP (tos 0x0, ttl 64, id 23323, offset 0, flags [DF], proto TCP (6), length 62)
    csslab4.umb.edu.58944 > csslab2.umb.edu.58101: Flags [P.], cksum 0x5111 (correct), seq 1:11, ack 1, win 210, options [nop,nop,TS val 3312950164 ecr 2632871252], length 10
1589526677.580568 IP (tos 0x0, ttl 64, id 51913, offset 0, flags [DF], proto TCP (6), length 52)
    csslab2.umb.edu.58101 > csslab4.umb.edu.58944: Flags [.], cksum 0x7587 (incorrect -> 0xc32f), ack 11, win 210, options [nop,nop,TS val 2632871252 ecr 3312950164], length 0
1589526677.580614 IP (tos 0x0, ttl 64, id 51914, offset 0, flags [DF], proto TCP (6), length 62)
    csslab2.umb.edu.58101 > csslab4.umb.edu.58944: Flags [P.], cksum 0x7591 (incorrect -> 0x5107), seq 1:11, ack 11, win 210, options [nop,nop,TS val 2632871252 ecr 3312950164], length 10
1589526677.580626 IP (tos 0x0, ttl 64, id 51915, offset 0, flags [DF], proto TCP (6), length 52)
    csslab2.umb.edu.58101 > csslab4.umb.edu.58944: Flags [F.], cksum 0x7587 (incorrect -> 0xc324), seq 11, ack 11, win 210, options [nop,nop,TS val 2632871252 ecr 3312950164], length 0
1589526677.580901 IP (tos 0x0, ttl 64, id 23324, offset 0, flags [DF], proto TCP (6), length 52)
    csslab4.umb.edu.58944 > csslab2.umb.edu.58101: Flags [.], cksum 0xc325 (correct), ack 11, win 210, options [nop,nop,TS val 3312950164 ecr 2632871252], length 0
1589526677.580928 IP (tos 0x0, ttl 64, id 23325, offset 0, flags [DF], proto TCP (6), length 1502)
    csslab4.umb.edu.58944 > csslab2.umb.edu.58101: Flags [P.], cksum 0x179b (correct), seq 11:1461, ack 12, win 210, options [nop,nop,TS val 3312950164 ecr 2632871252], length 1450
1589526677.580936 IP (tos 0x0, ttl 64, id 23326, offset 0, flags [DF], proto TCP (6), length 52)
    csslab4.umb.edu.58944 > csslab2.umb.edu.58101: Flags [F.], cksum 0xbd79 (correct), seq 1461, ack 12, win 210, options [nop,nop,TS val 3312950164 ecr 2632871252], length 0
1589526677.581083 IP (tos 0x0, ttl 64, id 51916, offset 0, flags [DF], proto TCP (6), length 52)
    csslab2.umb.edu.58101 > csslab4.umb.edu.58944: Flags [.], cksum 0x7587 (incorrect -> 0xbd62), ack 1462, win 233, options [nop,nop,TS val 2632871252 ecr 3312950164], length 0
^C
11 packets captured
19 packets received by filter
0 packets dropped by kernel
[younjon@csslab2 prog3]$
```

Analysis 1: State transition diagram and timing chart that traces the hw3 program.



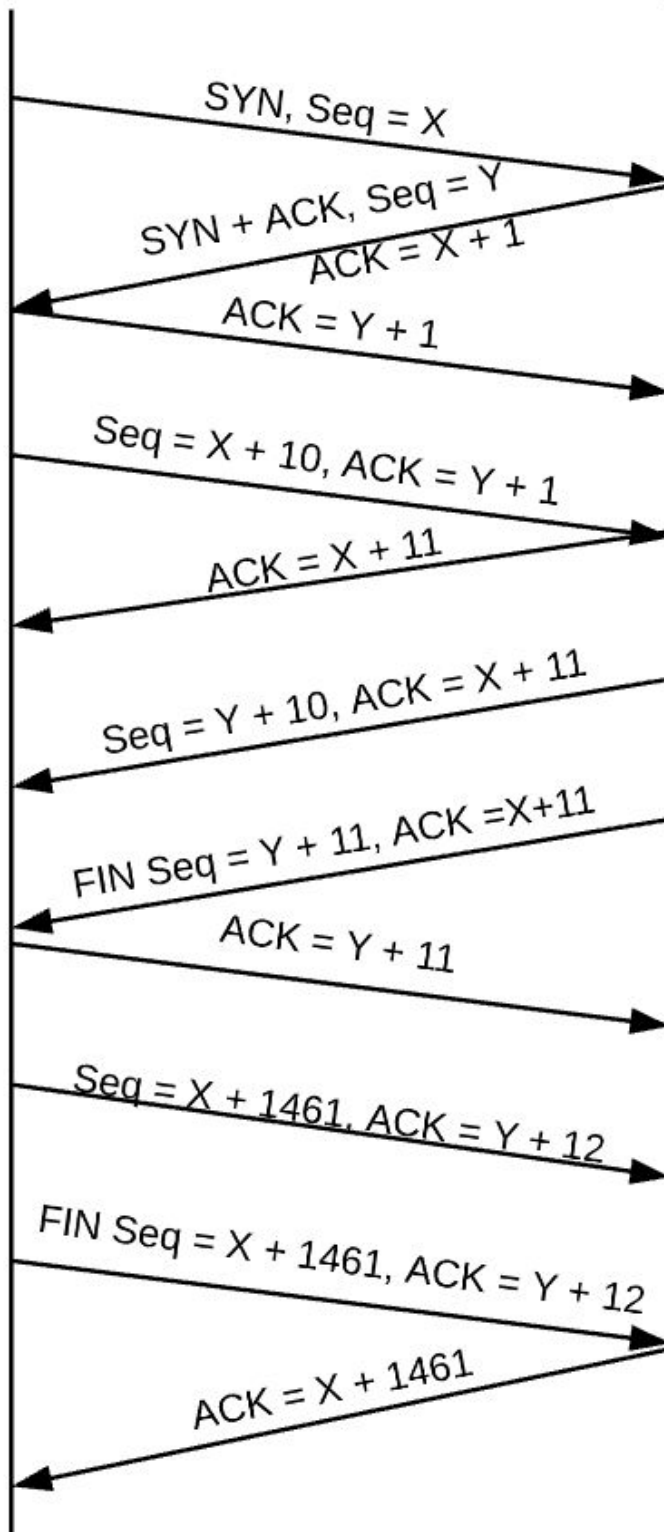
■ **FIGURE 5.7** TCP state-transition diagram.

Taken from "Computer Networks A Systems Approach" Peterson, Davie.

Timing Chart

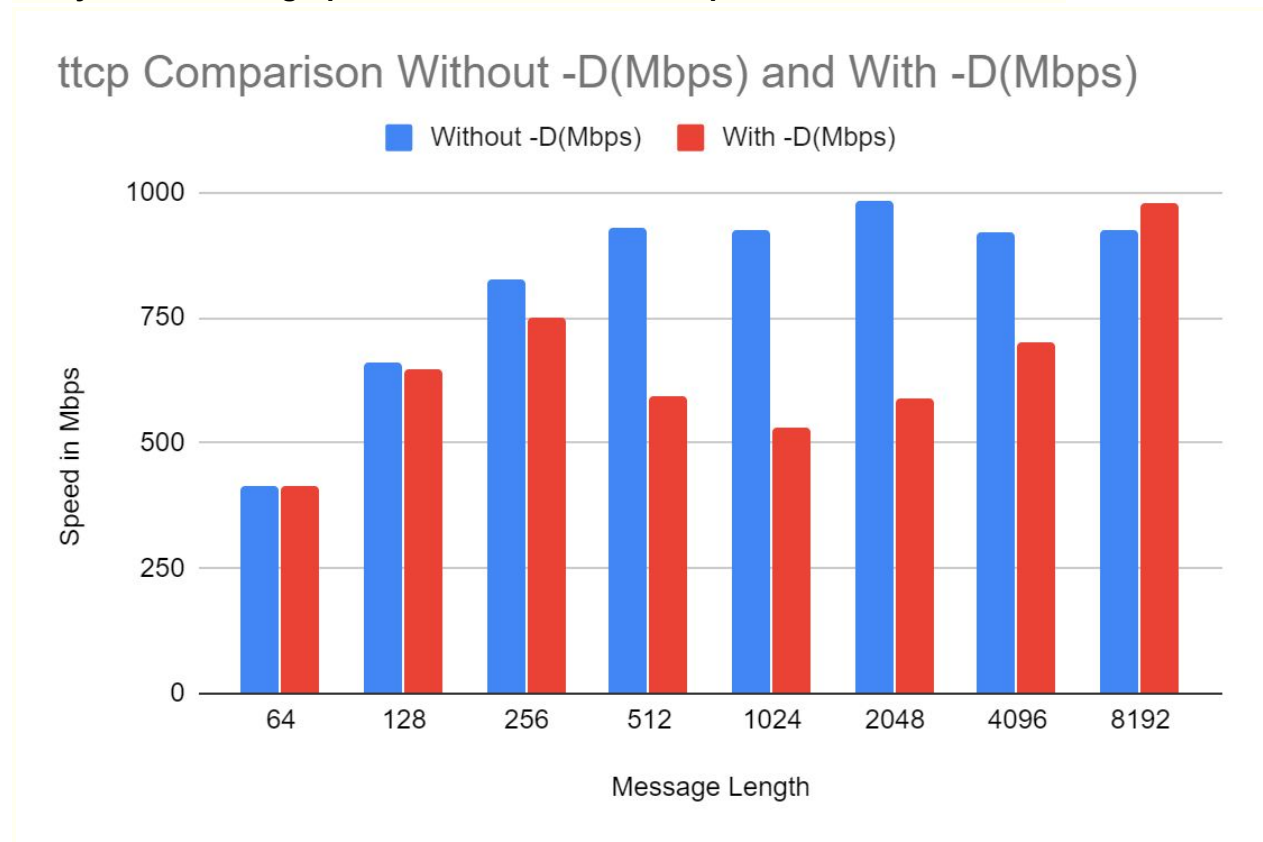
Client

Server



Test 2: Run ttcp on any two of csslab(x)'s machines, in the following test cases without and with -D option.

Analysis 2: Make a graph or a table in terms of Mbps for the test 2. Discuss:



ttcp Results Table for Length 64 *2i

Length (-l)	Number of Messages (-n)	Without -D(Mbps)	With -D(Mbps)
64	1048576	413.332	414.267
128	524288	662.163	645.424
256	262144	824.319	751.242
512	131072	931.209	594.829
1024	65536	923.317	529.403
2048	32768	984.716	588.779
4096	16384	919.373	702.035
8192	8192	924.237	979.227

The effect of buffer length without using -D option

1. Without the -D we can measure a steady, consistent increase in transmission speed. At message lengths of 512 or greater, there is not a significant increase in transfer speed.

Our fastest speed comes at 2048 message length giving a speed of 984.716Mbps, but that does not vary from the slowest speed in the frame from 512 to 8192 of 919.373Mbps at 4096 message length.

- 2. The effect of -D option** There is a noticeable speed delay from the third frame set of 256 through 4096 message length. With the lone exception of our final frame of 8192 which gave the only noticeable speed increase of 979.227Mbps over its counterpart. These results are unexpected since -D removes the TCP buffer writes and implies that messages would be sent faster.

Test 3: Run `ttcp` in the following particular test case without -D while you are running `tcpdump` on another xterm. Check from your dump file if TCP maximum segment size (MSS) is 1460 bytes or not.

Analysis 3: Make a graph describing how the advertised window changed on the server side when you ran test 3. Just focus on the first 10 acknowledgments from the server. Make clear:

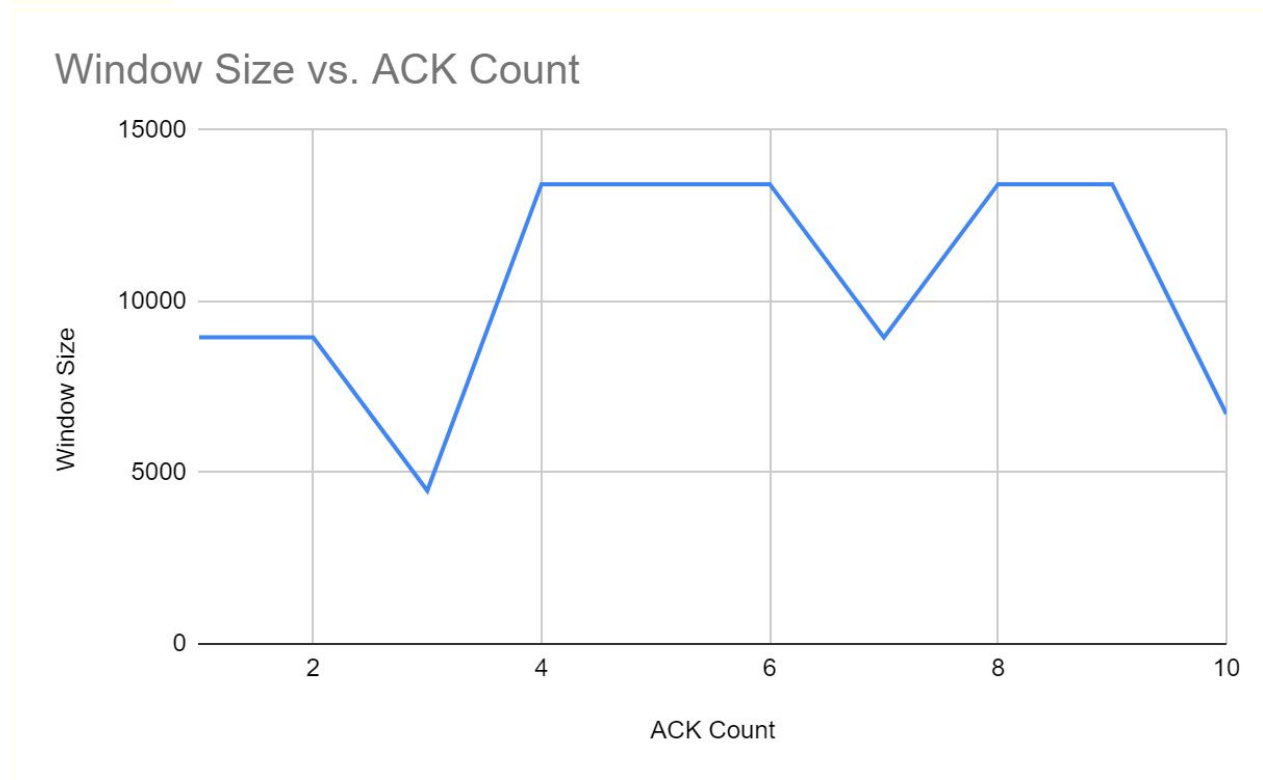


Table Representing ACK Count and Advertised Window Size

ACK Count	Window Size
1	8948
2	8948
3	8948
4	4474
5	13422

6	13422
7	13422
8	8948
9	13422
10	13422

1. Does the growth of the advertised window follow additive increment, slow start, or perhaps a different algorithm?

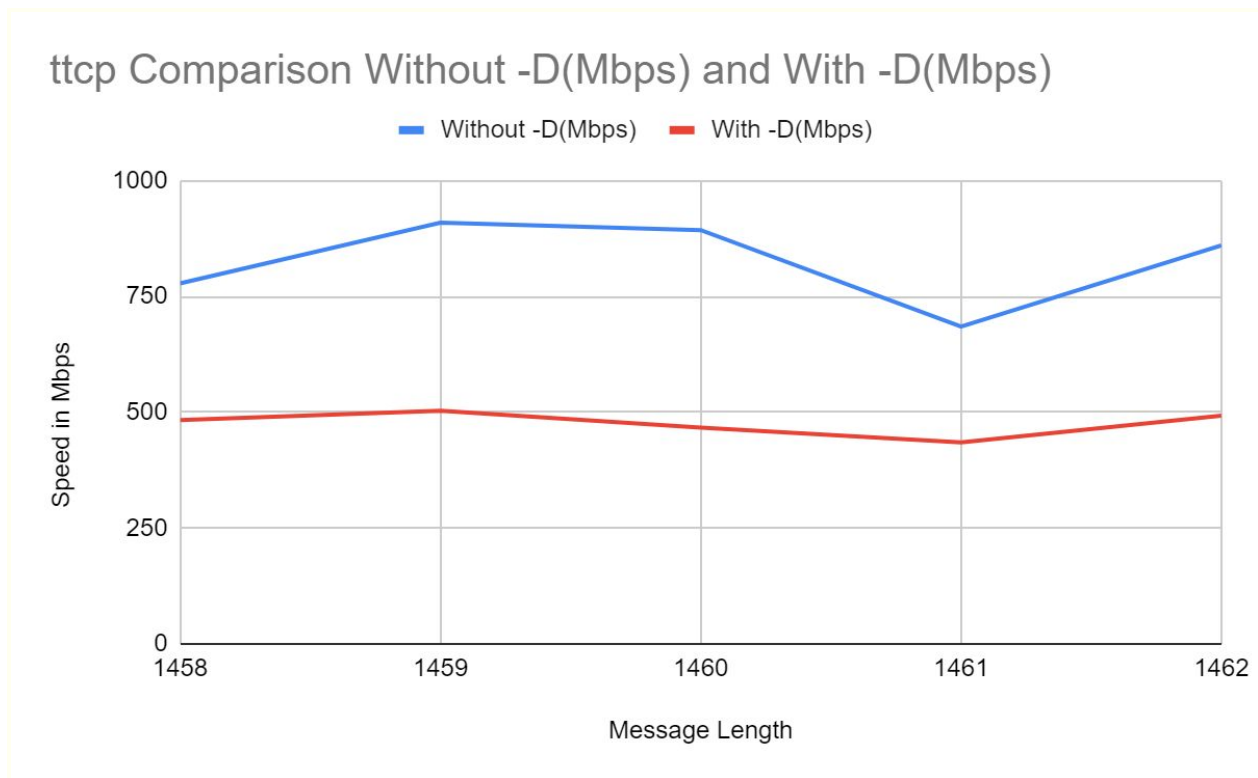
The window size seems to follow Fast Recovery protocols, since after ACK 3 we drop $\frac{1}{2}$ of 8948 to 4474, then increase again by 8948 to reach a peak of 13422. The next decline reduces the window size to the previously known good window size of 8948, dropping by 4474, before returning to the max window size again.

2. How large is MSS in TCP? Is it 1448 or 1460 bytes?

The MSS in this test was 8960 bytes.

Test 4: Run `ttcp` in the following 5 test cases where `-l` option is from 1458 to 1462 **with and without -D** option.

Analysis 4: Make a graph or a table in terms of Mbps for the test 4. Discuss:



ttcp ResultsTable For Length 1458 Through 1462

Length (-l)	Number of Messages (-n)	Without -D(Mbps)	With -D(Mbps)
1458	46028	778.459	483.196
1459	45996	909.289	503.239
1460	45965	893.286	466.892
1461	45934	685.076	435.078
1462	45902	860.197	492.663

1. The effect of buffer length without using -D option

This series of tests showed a consistently higher speed than its counterpart test using the -D option. Transfer speeds appear inconsistent, with minor changes to message length implying that this test series offers faster but unpredictable speed.

2. The effect of -D option

This series of tests showed that the -D option maintained a consistent speed, but still lower than without using the option. This is not inconsistent with the results from the 2nd test series showing supporting results from window sizes 1024 to 2048, having a similar transfer speed as well as difference in speed between both options.

Test 5: Run ttcp in the following particular test case **with and without -D** option. Run netstat right before and after each execution of ttcp to count the tcp packets sent, received, and retransmitted. Thereafter, run " strace -ttT ttcp " in the following particular test case **without and with -D** option.

Analysis 5: Discuss the effect of **-D option** based on the following results you received from the test 5:

1. the number of TCP packets sent, received, and retransmitted

Client Values Without -D

Received Before	92273902	Sent Before	109743201	Retransmitted Before	66448
Received After	92280687	Sent After	109753302	Retransmitted After	66448
Received Difference	6785	Sent Difference	10101	Retransmitted Difference	0

Total Segments 16886

Client Values With -D

Received Before	92280832	Sent Before	109753452	Retransmitted Before	66448
Received After	92289305	Sent After	109762924	Retransmitted After	66449
Received Difference	8473	Sent Difference	9472	Retransmitted Difference	1

Total Segments 17945

Server Values Without -D

Received Before	161475967	Sent Before	153517897	Retransmitted Before	69810
Received After	161485131	Sent After	153524583	Retransmitted After	69810
Received Difference	9164	Sent Difference	6686	Retransmitted Difference	0

Total Segments 15850

Server Values With -D

Received Before	161341460	Sent Before	153382330	Retransmitted Before	69810
Received After	161348856	Sent After	153390783	Retransmitted After	69810
Received Difference	7396	Sent Difference	8453	Retransmitted Difference	0

Total Segments 15849

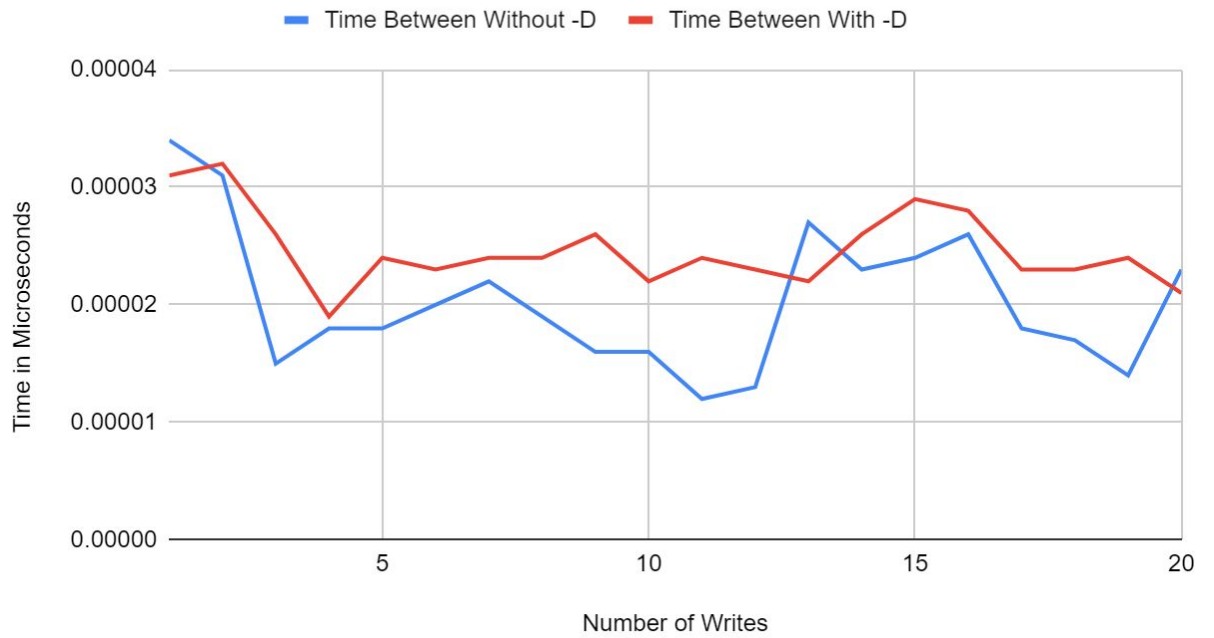
Client Differences: Increase in reception of segments with -D, decrease in sent segments. Only retransmission occurs with -D one time.

Server Differences: Decrease in reception of segments with -D, increase in sent segments. No retransmission.

Through this test we can observe that -D increases Client receptions and Server sent segments, while decreasing Client sent and Server reception segments. This suggests that it reduces server side work, while pushing more work on the clients side.

2. the time required for OS to execute each write system call (focus on the first 20 writes.)

Time Comparison of Writes Without -D and Time Between With -D



Write Time Comparison Table

Number of Writes	Time Between Without -D	Time Between With -D
1	0.000034	0.000031
2	0.000031	0.000032
3	0.000015	0.000026
4	0.000018	0.000019
5	0.000018	0.000024
6	0.000020	0.000023
7	0.000022	0.000024
8	0.000019	0.000024
9	0.000016	0.000026
10	0.000016	0.000022
11	0.000012	0.000024
12	0.000013	0.000023
13	0.000027	0.000022
14	0.000023	0.000026
15	0.000024	0.000029

16	0.000026	0.000028
17	0.000018	0.000023
18	0.000017	0.000023
19	0.000014	0.000024
20	0.000023	0.000021

The results of this test are very close, but we can see that without -D is slightly faster on average. With an average write speed of 0.0000203 microseconds without -D, and 0.0000247 microseconds with -D, we see roughly an 18% improvement without -D.