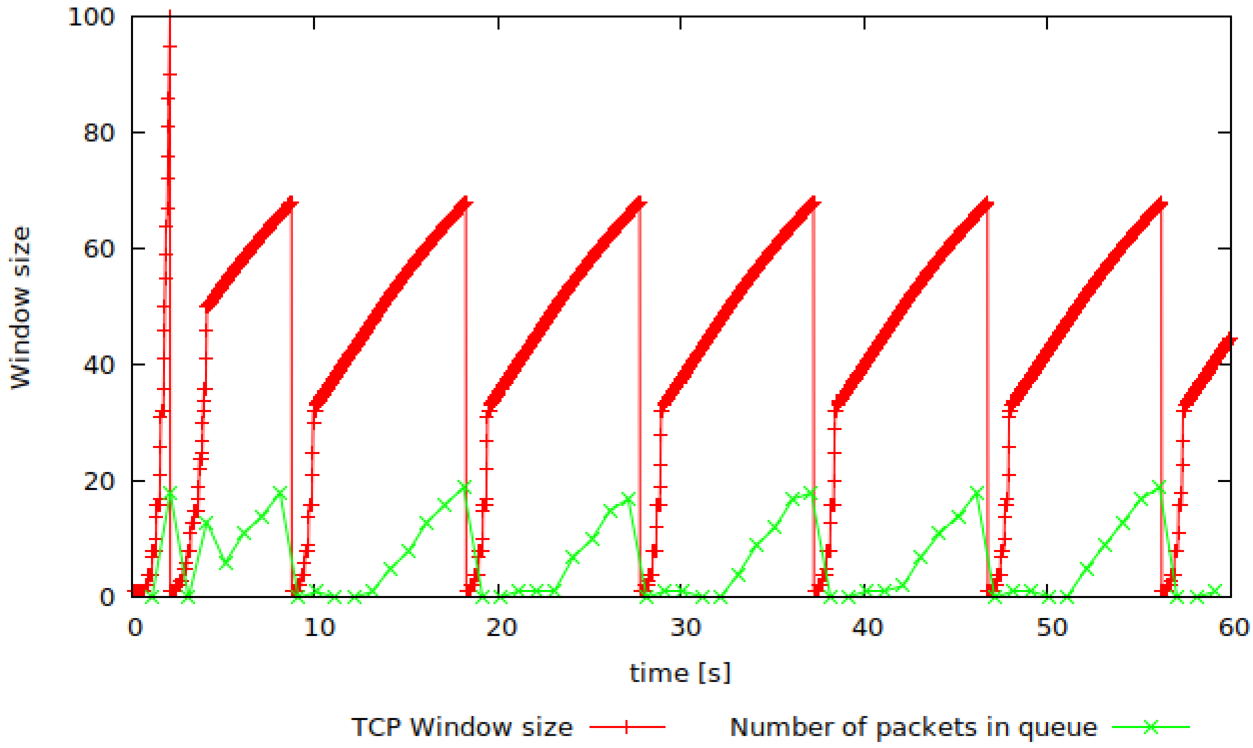
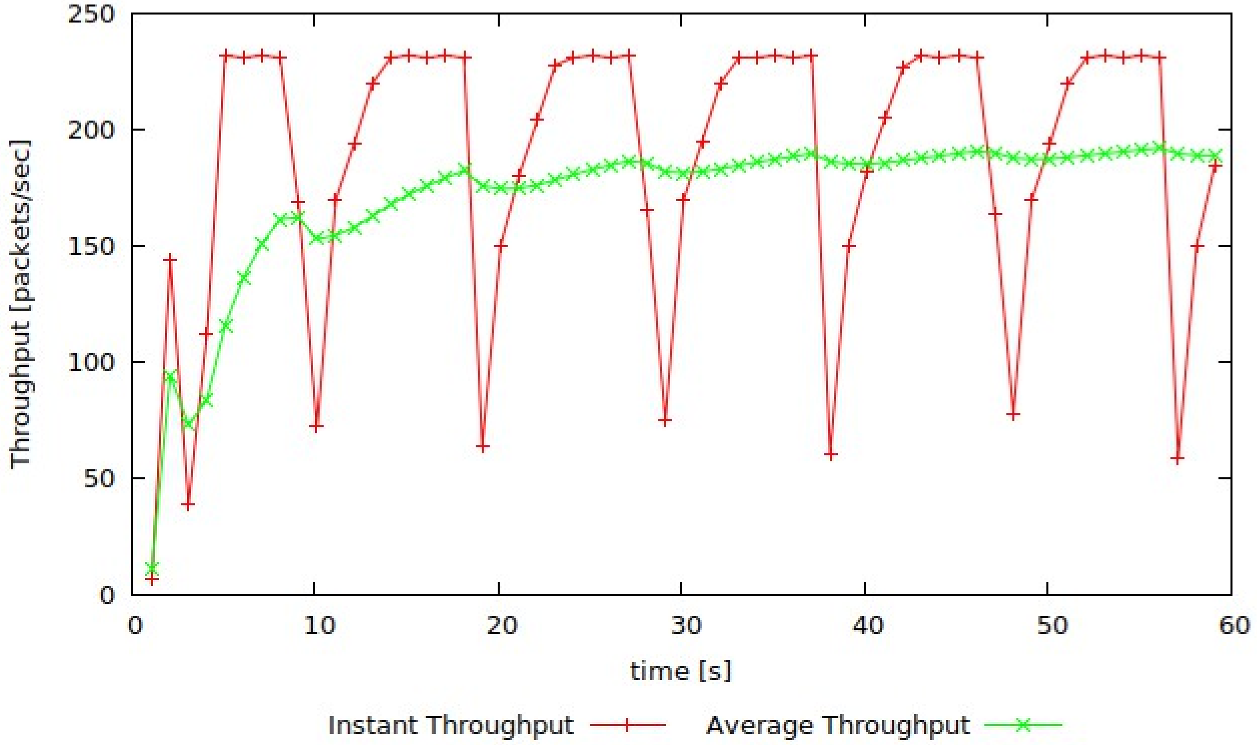
Exercise 1: Understanding TCP Congestion Control using ns-2

Question 1:
What is the maximum size of the congestion window that the TCP flow reaches in this case? What does the TCP flow do when the congestion window reaches this value? Why? What happens next? Include the graph in your submission report.

Over 100. It drops back to 1 from the start and because a time out has occurred. It starts from 1 and try to increase the cwnd using the slow start.



Question 2: From the simulation script we used, we know that the payload of the packet is 500 Bytes. Keep in mind that the size of the IP and TCP headers is 20 Bytes, each. Neglect any other headers. What is the average throughput of TCP in this case? (both in number of packets per second and bps)



The average packets/second is about 180packets/second.

The average throughput in bps is about 180*8bit/byte* (540) bps. That is 777,600bps.

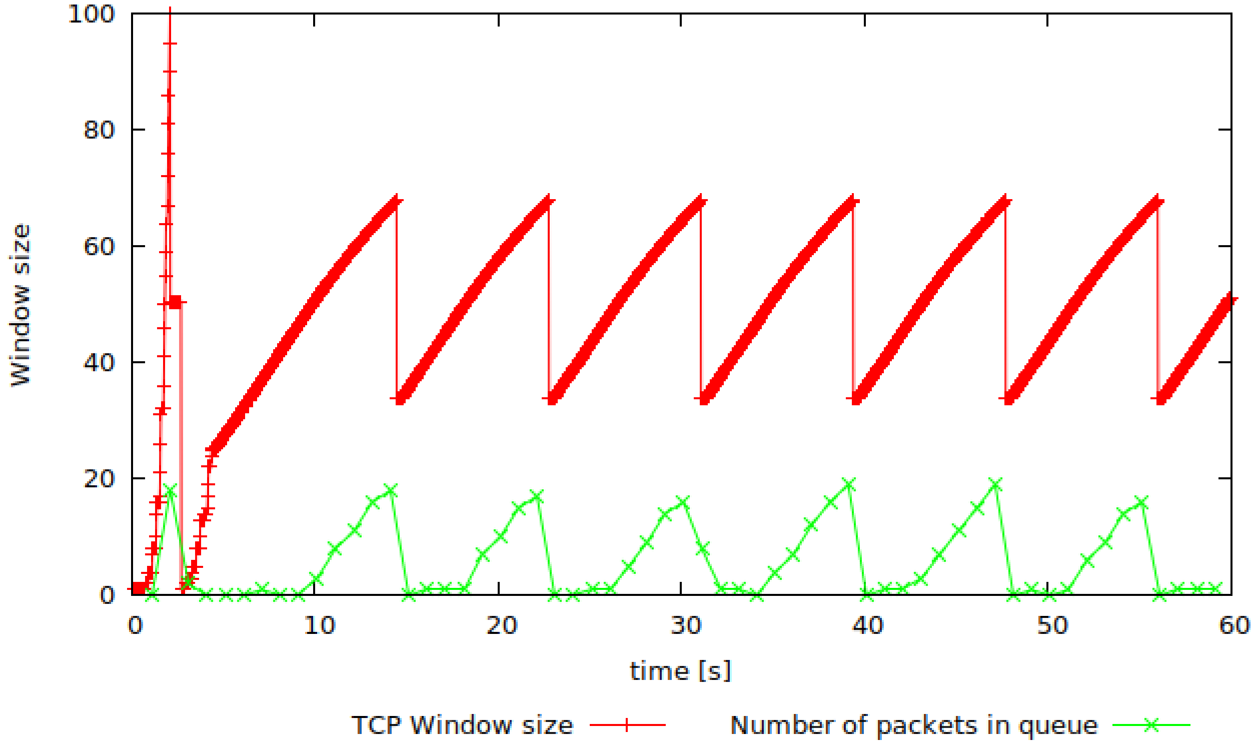
Question 3: Rerun the above script, each time with different values for the max congestion window size but the same RTT (i.e. 100ms). How does TCP respond to the variation of this parameter? Find the value of the maximum congestion window at which TCP stops oscillating (i.e., does not move up and down again) to reach a stable behaviour. What is the average throughput (in packets and bps) at this point? How does the actual average throughput compare to the link capacity (1Mbps)?

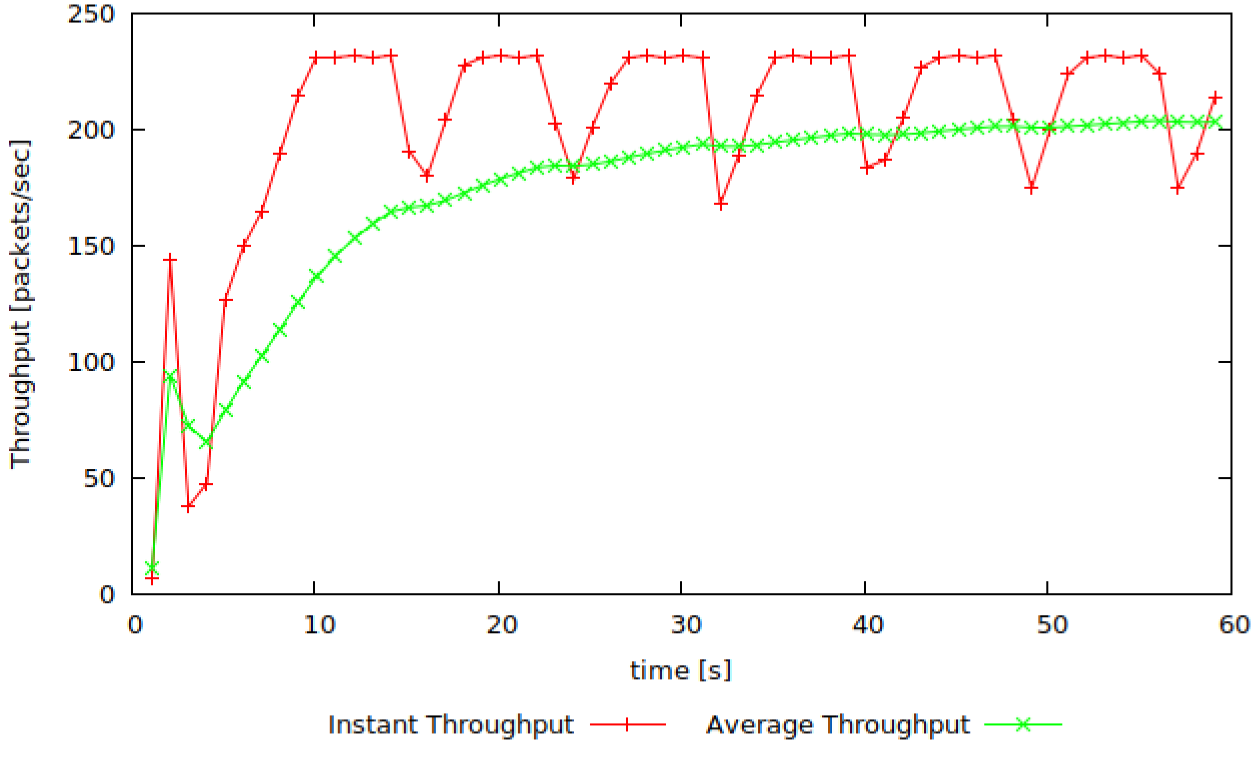
Question 4: Repeat the steps outlined in Question 1 and 2 (NOT Question 3) but for TCP Reno. Compare the graphs for the two implementations and explain the differences. (Hint: compare the number of times the congestion window goes back to zero in each case). How does the average throughput differ in both implementations?

The TCP Reno does not need to start from cwnd of 1 and do the slow start every time.

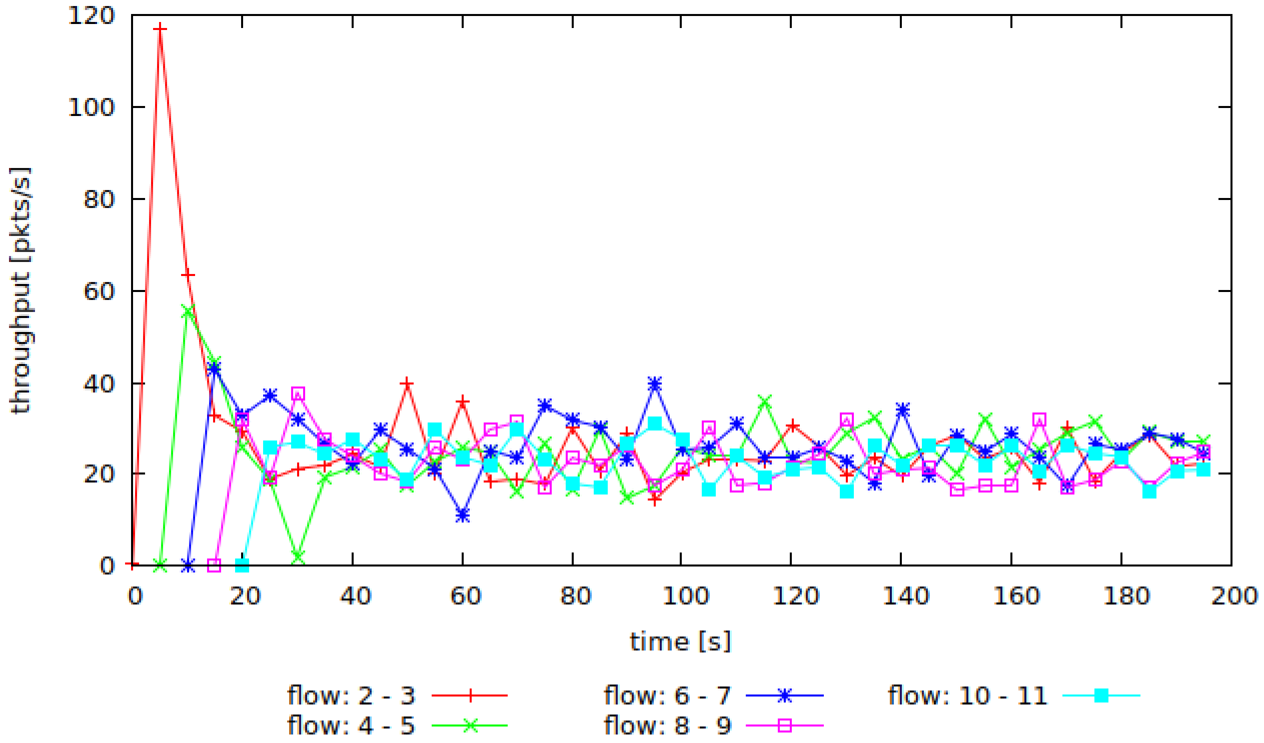
Throughput: (540)* 8 *200 = 864,000 bps

Hence TCP Reno performs better.


Window size
100
80
60
40
20
0
0
10
20
30
40
50
60
time [s]
TCP Window size
Number of packets in queue


Throughput [packets/sec]
250
200
150
100
50
0
0
10
20
30
40
50
60
time [s]
Instant Throughput
Average Throughput

Exercise 2



Question 1: Does each flow get an equal share of the capacity of the common link (i.e., is TCP fair) ? Explain which observations lead you to this conclusion.

Yes, the average of the throughput are equally alike when they are working simultaneously.

Question 2. What happens to the throughput of the pre-existing TCP flows when a new flow is created? Explain the mechanisms of TCP which contribute to this behaviour. Argue about whether you consider this behaviour to be fair or unfair.

When the new flow is created, the window of the pre-existing flow is decreased. This is considered to be fair and can make the best use of the link.

EXERCISE 3:

Question 1: How do you expect the TCP flow and the UDP flow to behave if the capacity of the link is 5 Mbps ?

I expect UDP will have greater throughput.

Question 2: Why does one flow achieve higher throughput than the other? Try to explain what mechanisms force the two flows to stabilise to the observed throughput.

UDP has no congestion control hence it might have higher throughput.

Question 3: List the advantages and the disadvantages of using UDP instead of TCP for a file transfer, when our connection has to compete with other flows for the same link. What would happen if everybody started using UDP instead of TCP for that same reason?

The network might collapse.