CN Assignment - 3

Samyak Jain, 2019098

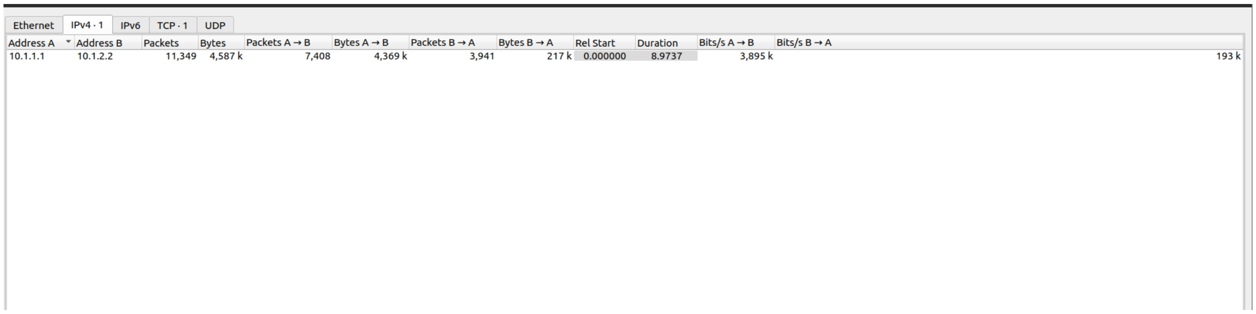
<u>Question - 1</u>

(A) Link 1 (N0-N1) bandwidth = 10Mbps
Link 2 (N1-N2) bandwidth = 5Mbps
The bottleneck bandwidth is 5 Mbps. Ideally, without taking into account any link delays, this should be the maximum expected theoretical value of the throughput i.e - 5 Mbps**.**
Also if we try to calculate the maximum bandwidth, we see that the total number of packets received at node N2 is 10^5. The packet size used here is 1460 bytes that is 1460*8 = 11680 bits. The total data received thus at node N2 is :
D_{total} = (Number of packets) * (Packet Size) = 10^5 * 11680 bits = 1168 Mb
The delays for both the links are 10ms and 15ms respectively. So total time taken for the transmission = T_{total} = (D_{total} / 5) + 0.01 + 0.015 seconds. (We divide D_{total} by 5 because the bottleneck bandwidth is 5 Mbps. On putting D_{total} = 1168 Mb we get
T_{total} = 233.625 seconds.
Throughput = D_{tota} / T_{total} = 1168 Mb / 233.625 = 4.99946 Mbps
We can thus see that this value of throughput calculated taking into account the delays is also very close to 5 Mbps.

(B) For the bandwidth-delay product,
BDP (in terms of no. of packets) = (Data link capacity * (RTT))/ (packet size)
Here link capacity is the bottleneck capacity given by N1-N2 link i.e - 5 Mbps which is
5 * 10^6 bits per second.
RTT = 2 * (N0-N1 link delay + N1-N2 link delay) = 2 * (10ms + 15ms)
= 50ms = 0.050 seconds
Packet size = 1460 bytes = 1460 * 8 bits = 11680 bits

BDP (in terms of no. of packets) = (5 * 10^6 * 0.050) / (11680) = 21.404
We can take approximation of rounding down the decimal answer of BDP to 21 because no. of packets can't be fractional.

(C) Average throughput of TCP transfer = 3,895 kbps = 3.895 Mbps

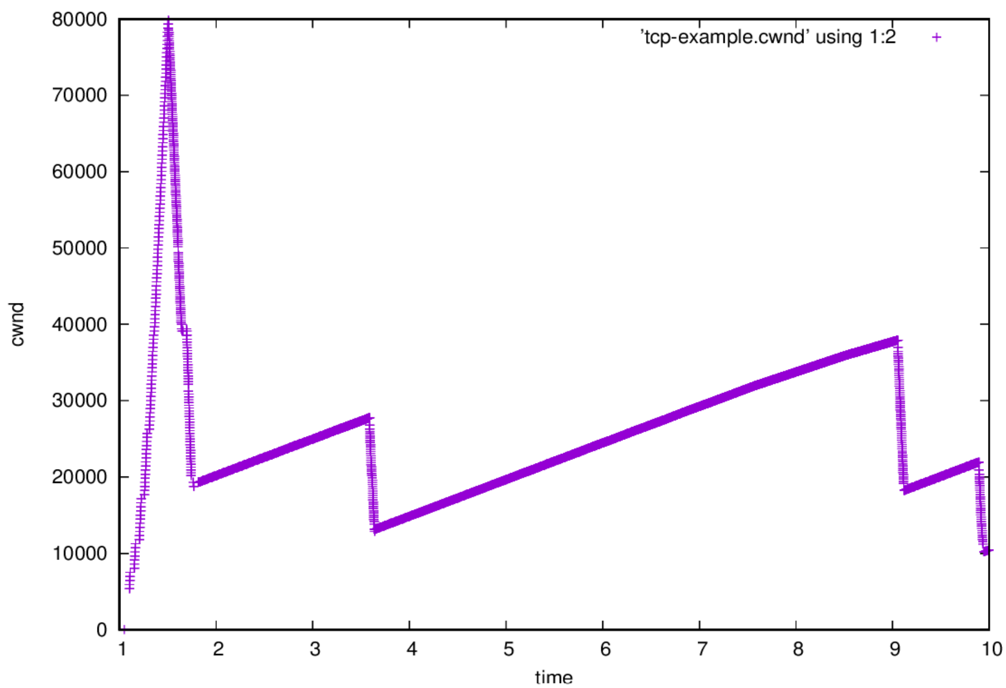


Ethernet | IPv4 · 1 | IPv6 | TCP · 1 | UDP

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
10.1.1.1	10.1.2.2	11,349	4,587 k	7,408	4,369 k	3,941	217 k	0.000000	8.9737	3,895 k	193 k

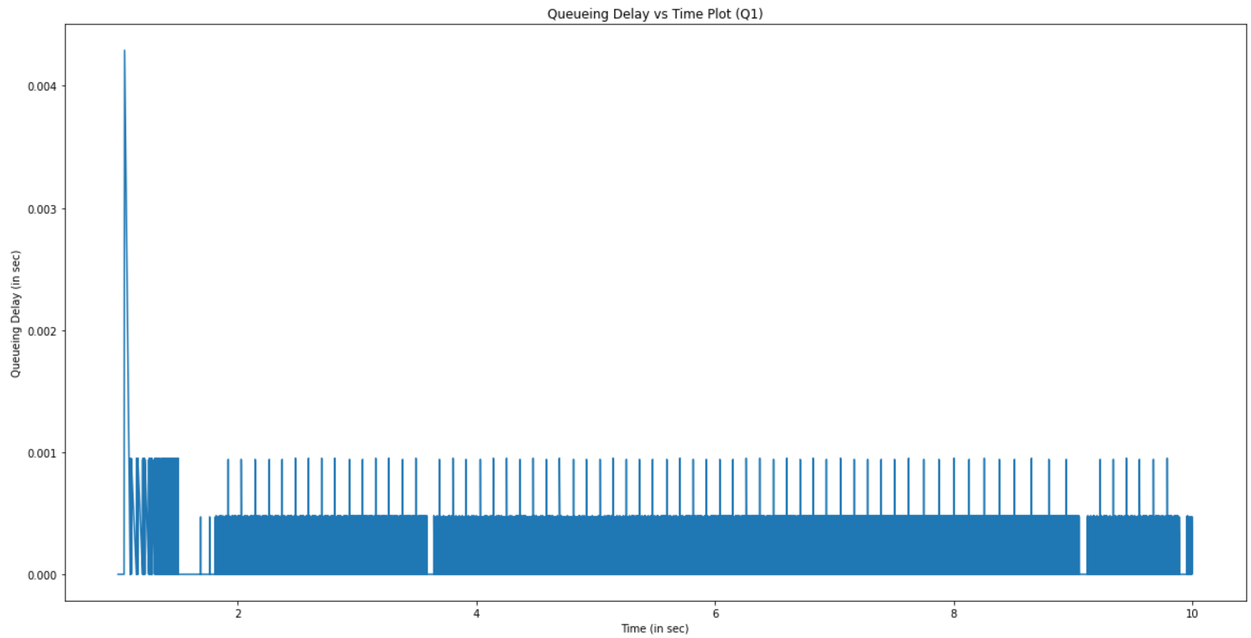
(D) No it is not equal. We can observe that the observed average throughput is lesser than the maximum theoretical expected value of throughput. It can be explained by multiple factors which control the throughput. We can see that there are delays associated with the given network links which contributes to a lesser average throughput. Also, in TCP transmissions the packets not properly transmitted or lost have to be retransmitted which can lead to lowering of the throughput. There are also some queuing delays which are dependent on the queue size factor and it has an effect on the throughput.

(E) Plot of Congestion Window with time

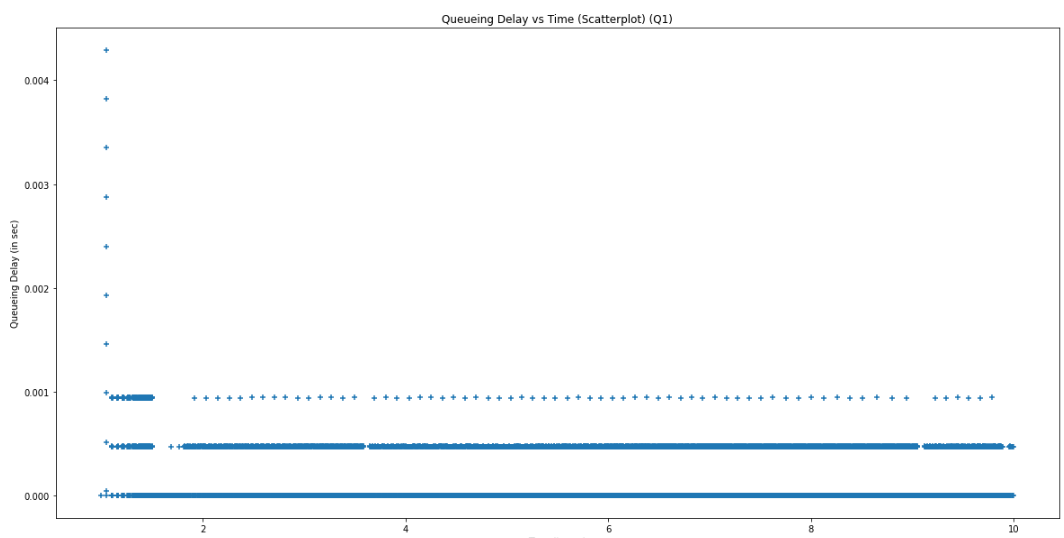


(F) Queueing delay with time

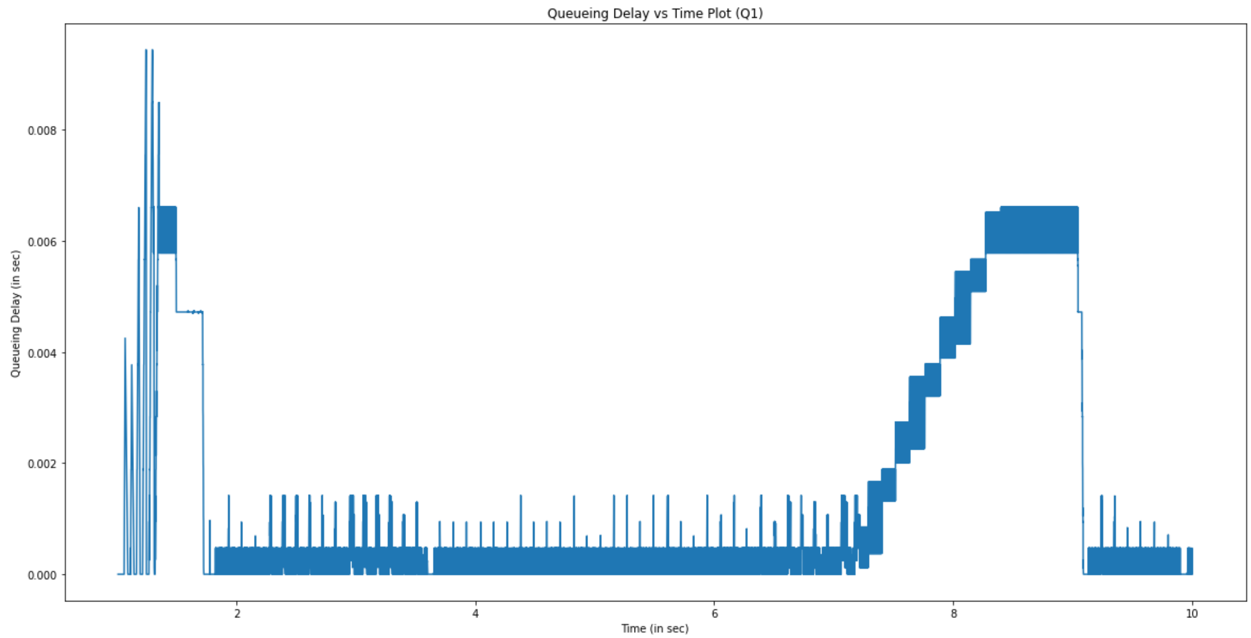
For N0 - Line Plot:



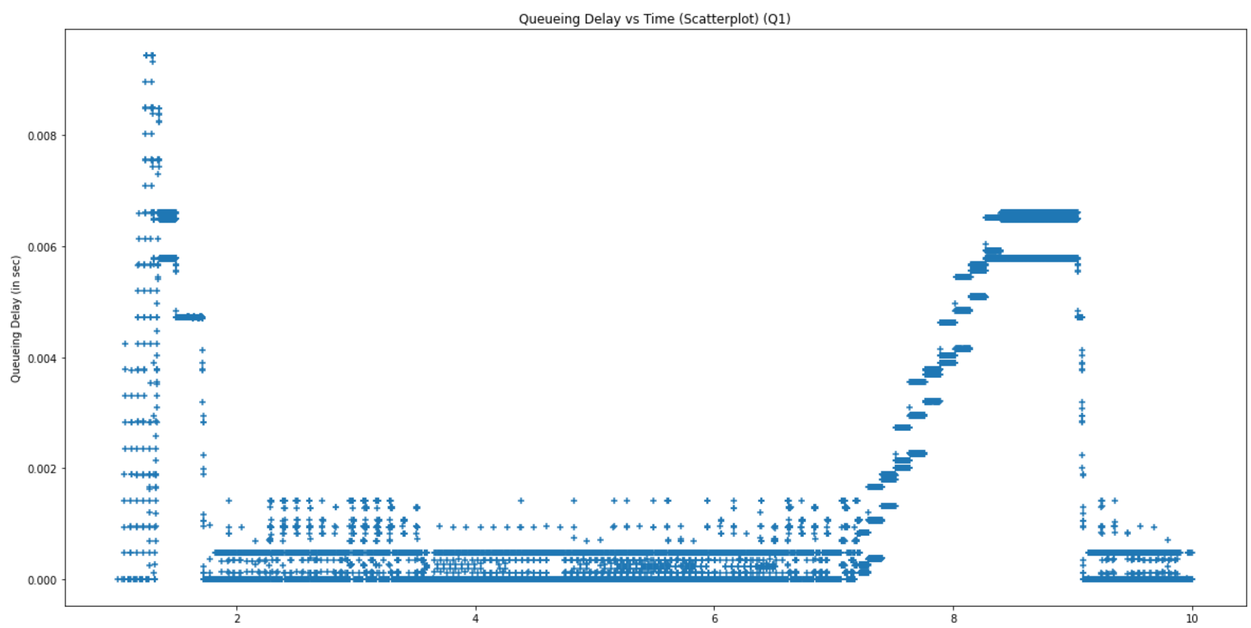
For N0 - Scatter plot :



For N1- Lineplot:



For N1 - scatter plot:



(G) The plots in 1(e) and 1(f) are related. We can observe from the graph that whenever the queueing delay reaches a peak in the graph, the congestion window size as seen from the CWND plot is halved. It can be observed from seeing the CWND and delay at node one around 1.5 seconds & at around 9-sec mark. This corresponds to the TCP phase of a fast recovery when the window size is halved. During the phase of linear increase in window size, the queueing delay somewhat is increasing slowly and when it reaches a peak, the window size is halved.

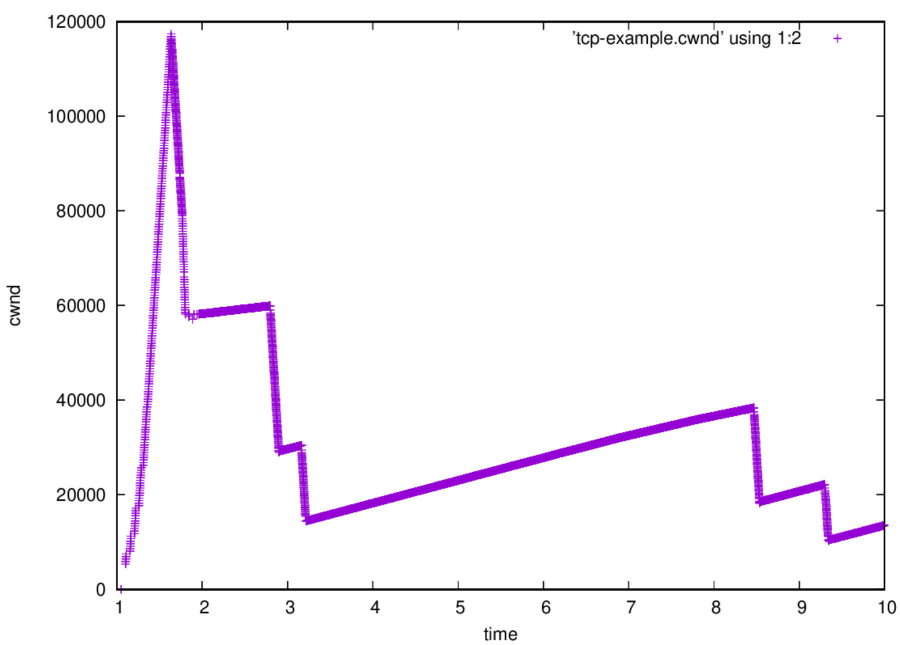
<u>Question - 2</u>

(A) Average throughput of TCP transfer = 4,024 kbps = 4.024 Mbps

Ethernet | IPv4 · 1 | IPv6 | TCP · 1 | UDP

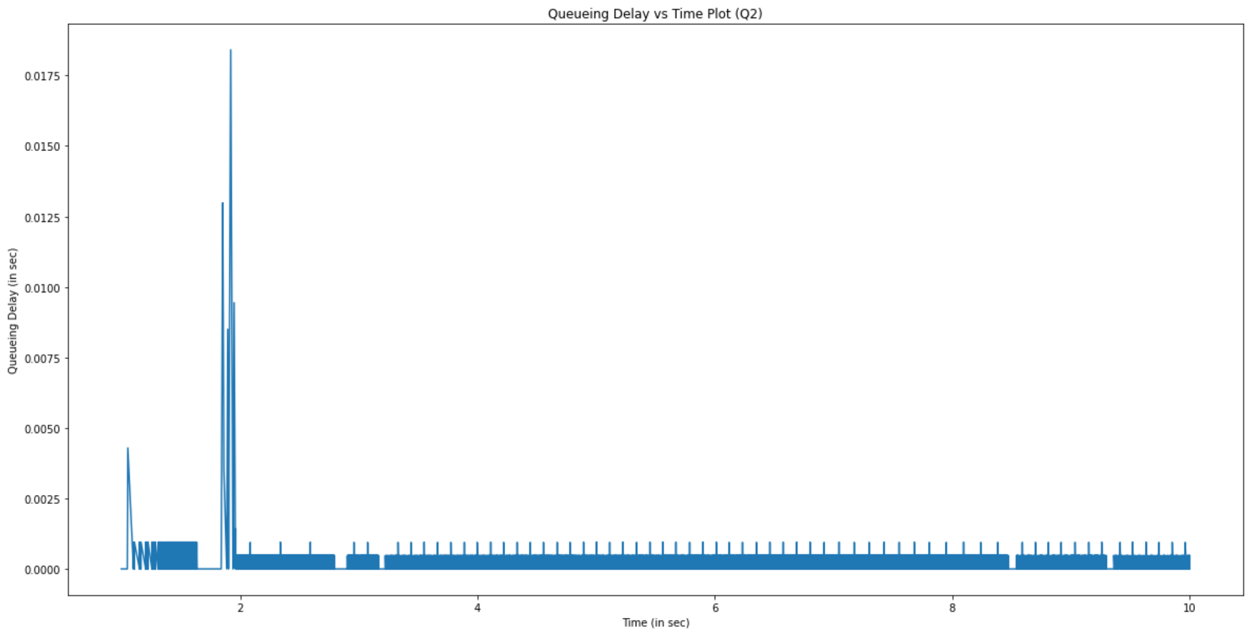
Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
10.1.1.1	10.1.2.2	11,800	4,745 k	7,656	4,514 k	4,144	230 k	0.000000	8.9746	4,024 k	205 k

(B) Plot of CWND with time

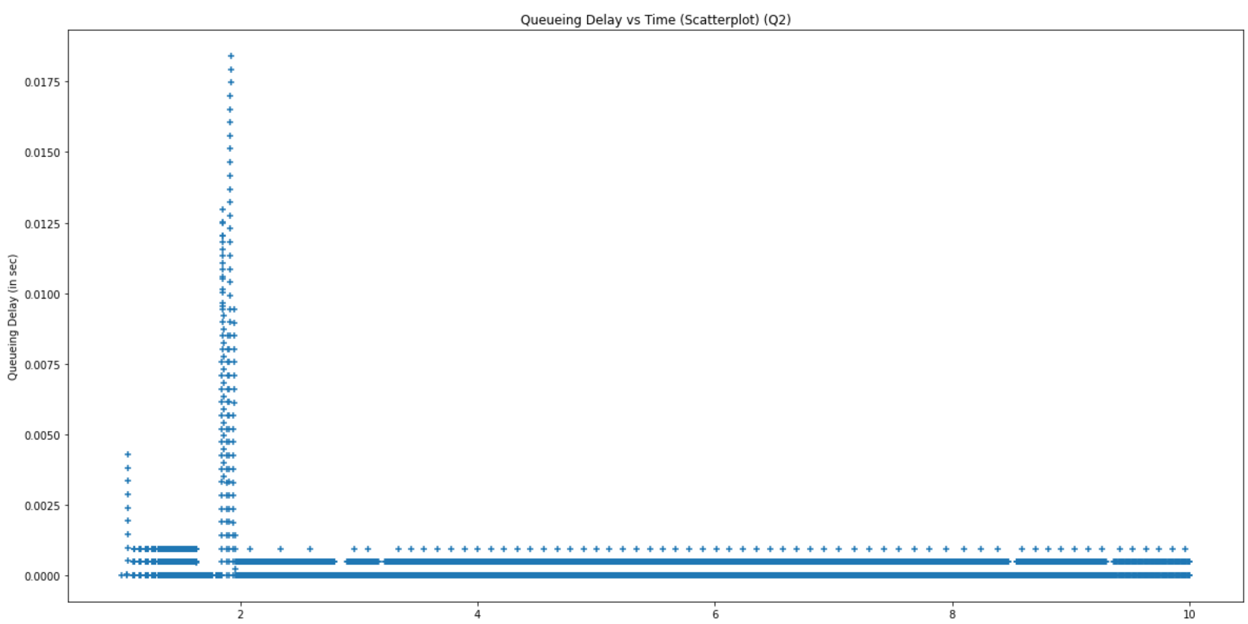


(C) Queueing delay with time

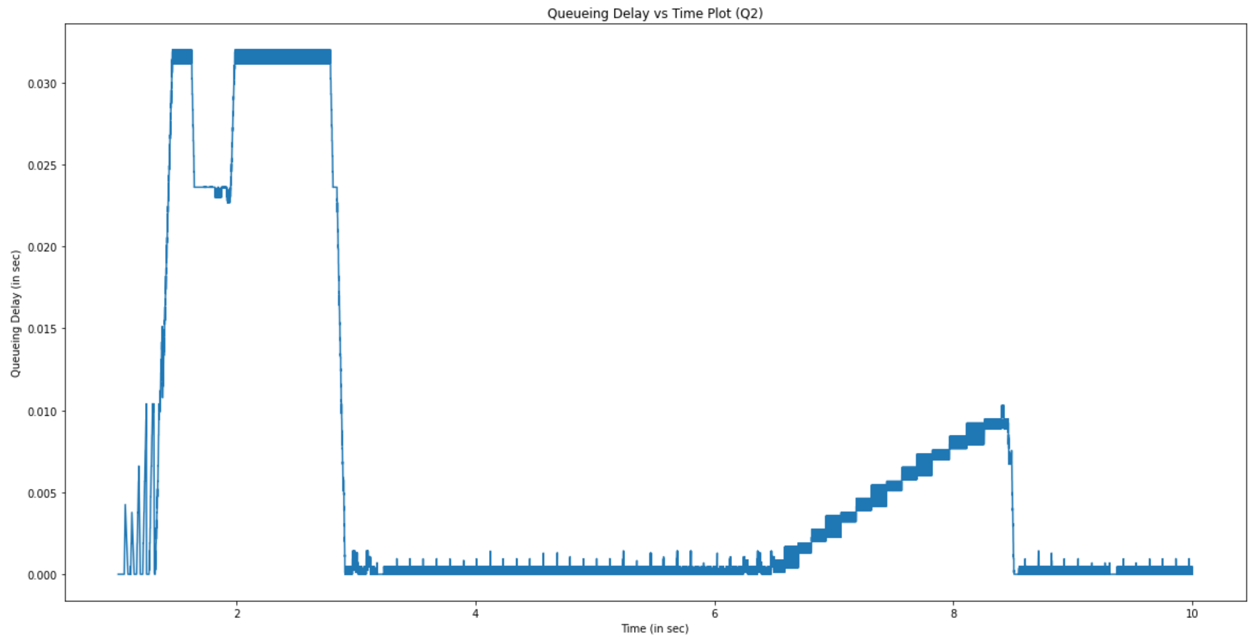
For N0 Line plot:



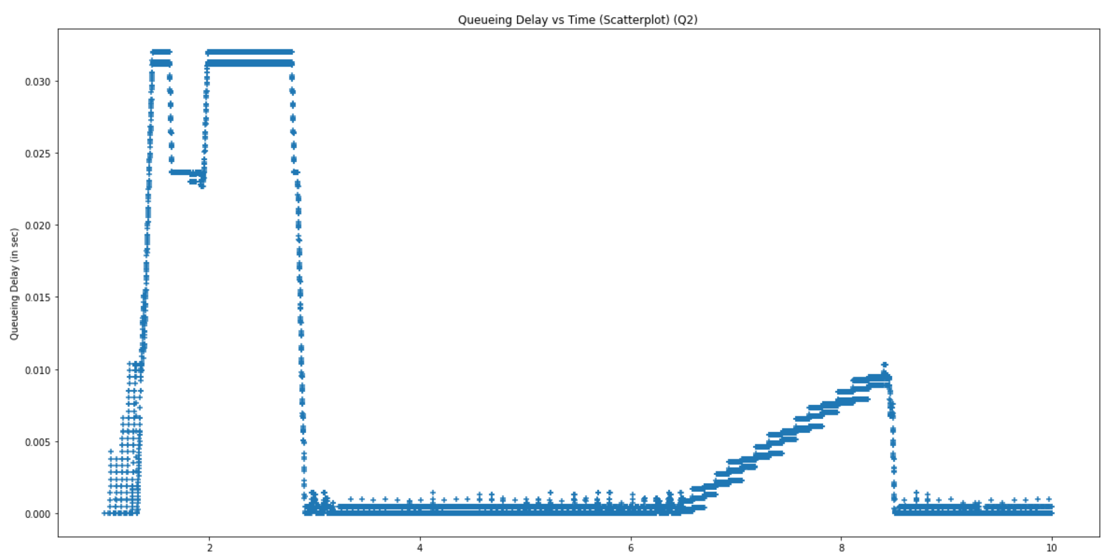
For N0 Scatter plot:



For N1 Lineplot:



For N1 scatter plot:



(D) On comparing the CWND plots for Q1 and Q2, I made the observation that the peak value of CWND for Q1 (80000) is lower than the peak value for CWND for Q2(120000). This is probably because increasing the queue size allows more packets to enter the buffer and wait in the queue. The sender can thus send more packets from its end without any packet drops or losses. This corresponds to a higher CWND value. Further, the values of CWND are reduced every time whenever we encounter a lost packet(duplicate ack) or a timeout. The slow start thresholds are initialized to a value equal to half of the value of CWND. In Q2 due to a higher peak value of CWND in the slow-start phase, the further values of CWND (reduced at every phase change) and slow start thresholds are higher as compared to Q1.

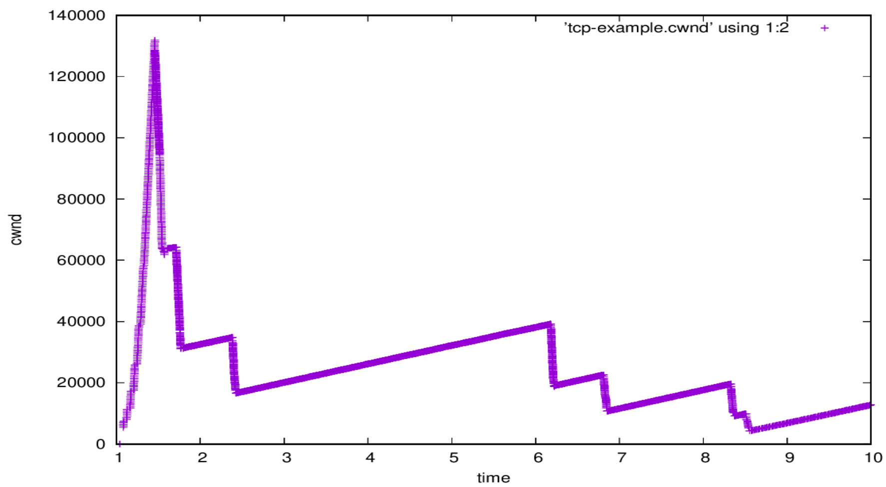
Question - 3

(A) Average throughput of TCP transfer = 4,717 kbps = 4.717 Mbps

Ethernet | IPv4 · 1 | IPv6 | TCP · 1 | UDP

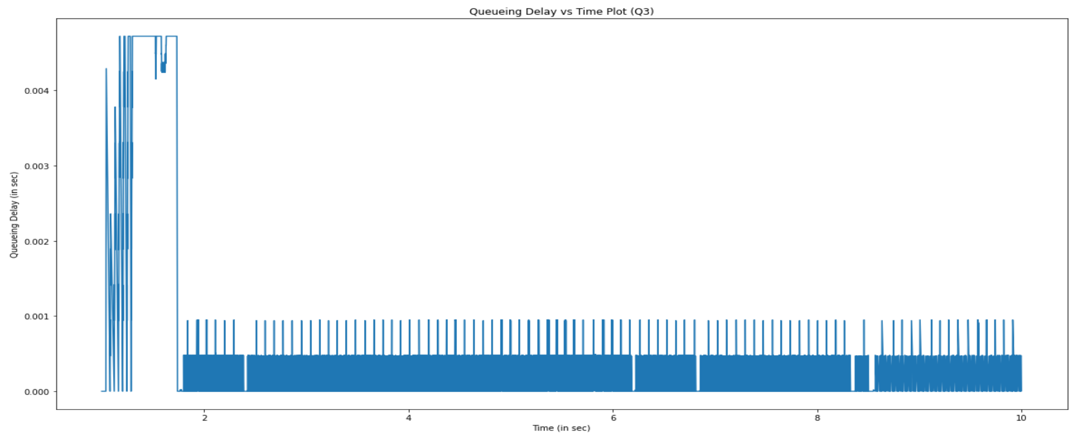
Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
10.1.1.1	10.1.2.2	13,868	5,562 k	8,981	5,292 k	4,887	270 k	0.000000	8.9746	4,717 k	240 k

(B) Plot of CWND with time

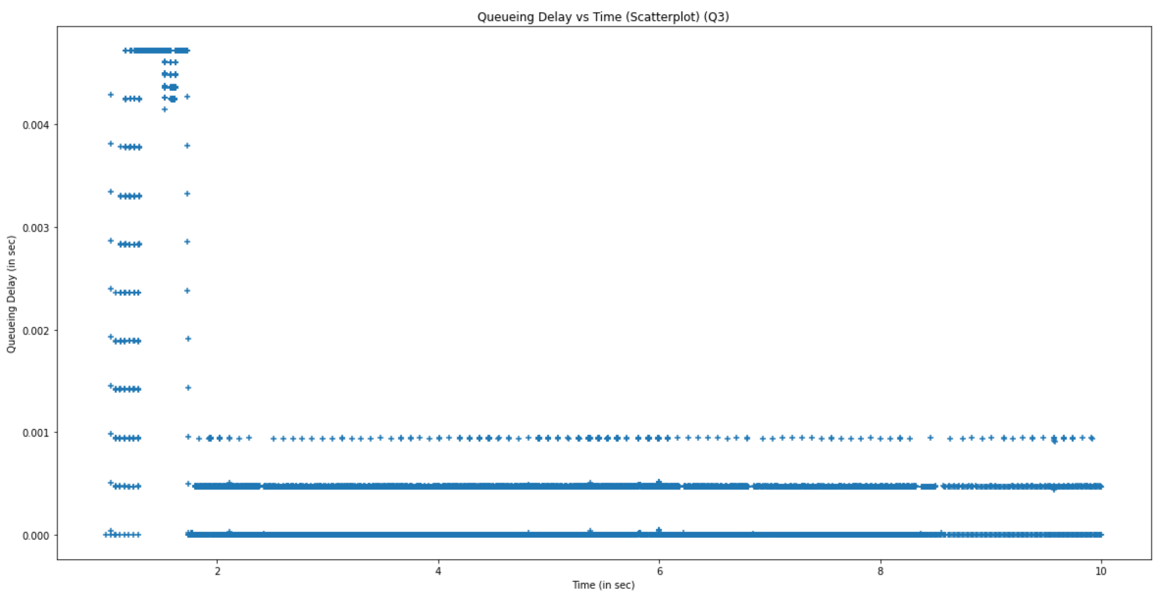


(c) Queueing delay plots

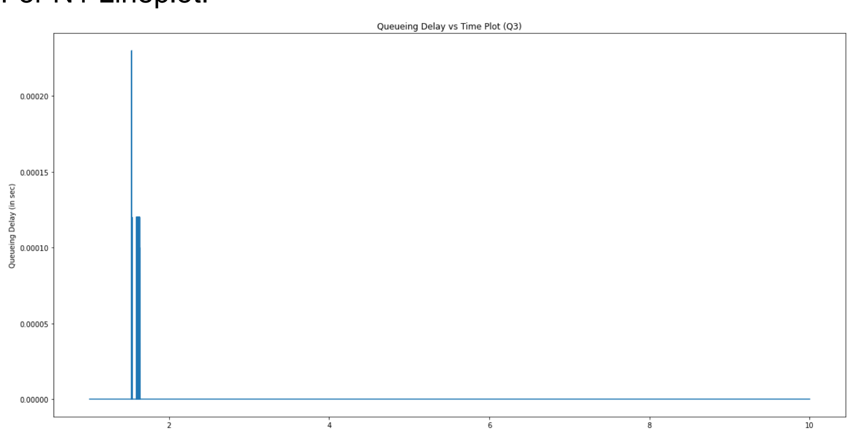
For N0 Line Plot:



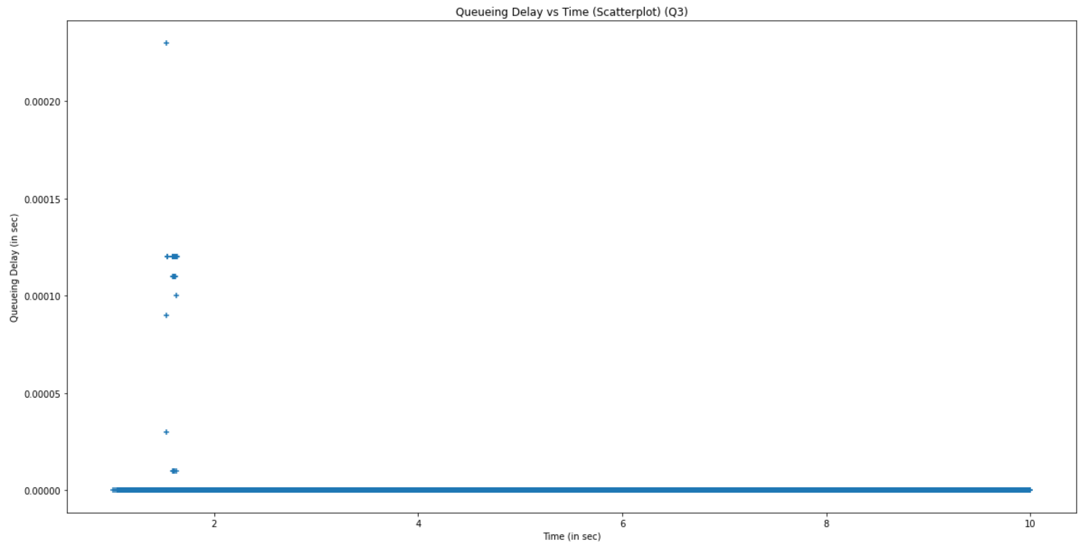
For N0 Scatter plot :



For N1 Lineplot:



For N1 scatter plot:



(D) On comparing the queuing delay plots for Q1 and Q3 for the respective nodes, I observed that corresponding to N0 the queueing delay is higher in the starting phase of about 0-2 seconds. This is because we have increased the bandwidth of the N1-N2 link(which was earlier the bottleneck bandwidth link) to 10Mbps. The bottleneck bandwidth is now 10 Mbps. Moreover, the delay corresponding to that has also been lowered. Both of these factors will contribute to more number of packets being transmitted and get accumulated in the queue. Thus the traffic at the queue is increased and the queueing delay is higher. We can also see with respect to queue at node one, there will be a queue formed in the case of question 1 because the link bandwidth between n0-n1 is higher than the bandwidth for n1-n2. But in the case of question 3, the bandwidths and delays of both are the same and the queueing delay is mostly zero.

Question - 4

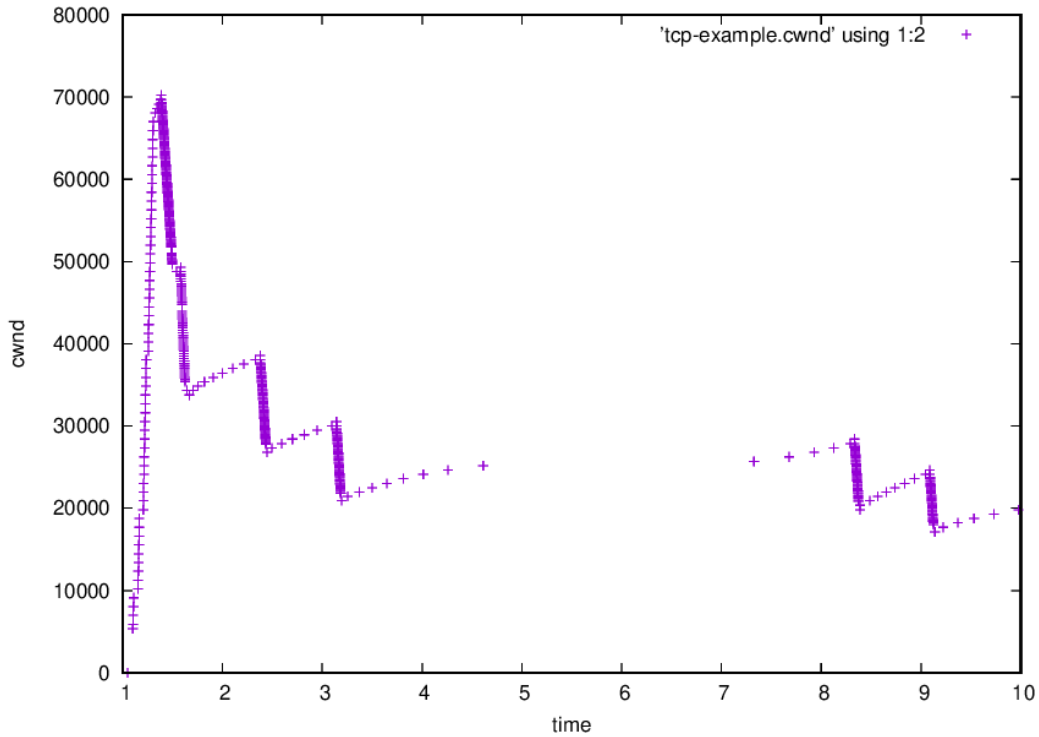
Parameters used:

(A) Average throughput of TCP transfer = 4,195 kbps = 4.195 Mbps

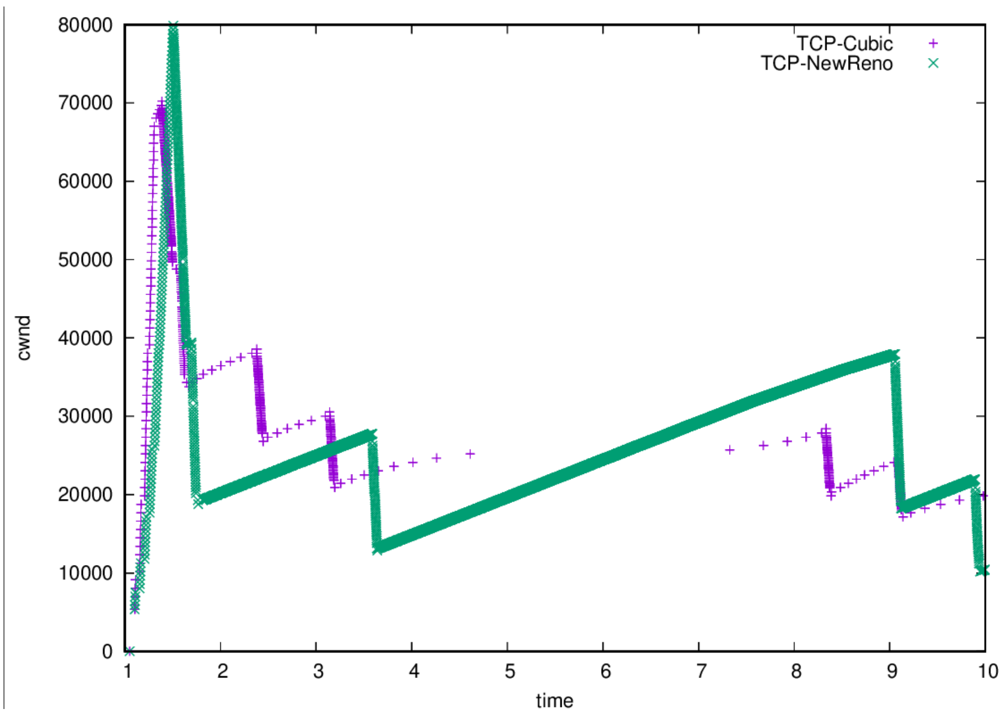
Ethernet | IPv4 · 1 | IPv6 | TCP · 1 | UDP

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
10.1.1.1	10.1.2.2	13,868	5,562 k	8,981	5,292 k	4,887	270 k	0.000000	8.9746	4,717 k	240 k

(B) Plot of CWND with time



(C) Plotting both of them together for a better comparison



Comparing both the plots for Q1 and Q4,

In both TCP Cubic and TCP New Reno, the first part of the slow start phase is similar because the same algorithm is followed in the slow start phase. Some differences are observed in the phase of congestion avoidance. In Q1 (TCPNewReno) the congestion window is increased linearly (it uses AIMD - Additive Increase Multiplicative Decrease). TCP cubic on the other hand uses a cubic function to increase the window. the increase in window size is a cubic function of distance between current time and the point in time where window sizes will reach its next maximum value. Thus due to this, the increase is faster when it is farther from its next maximum value and slower when it is closer to the next maximum value. The fast recovery phase in both the TCP variants has a minor difference in the terms that the factor by which window is reduced is not exactly half in the case of TCP Cubic but it is half in case of TCP New Reno.