Assignment-3

Q1)

Part (a)

The maximum expected value of the throughput theoretically speaking is 5 Mbps. The formula I have used is:

The bandwidth of network= min{ all the bandwidth of the links}

The bandwidth of a network is the maximum number of packets that can be transmitted through the network at a point in time. So let suppose I have one link in the network as L1 and another link as L2. Let bandwidth of L1 (say b1 packets can be transmitted at one point of time through L1) be less than the bandwidth of L2 (say b2 packets can be transmitted through L2 at a point of time). Let B1 packets pass through L1 and L2, but b2 packets can not be transmitted through L1. So the bandwidth of the entire network should be b1. If you observe b1= min{b1,b2}. Hence the link between n0-n1 has a bandwidth of 10 Mbps, and the link between n1-n2 has a bandwidth of 5 Mbps. So the bandwidth of the entire network should be min{5 Mbps, 10 Mbps } => 5 Mbps.

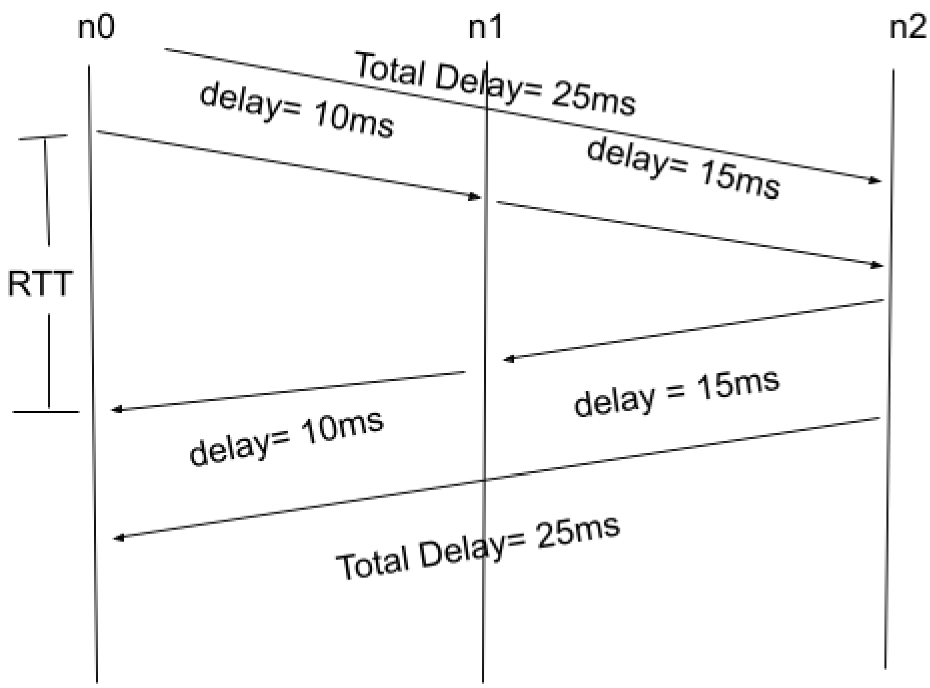
Part (b)

Answer: 21.4 packets

Bandwidth Delay Product(BDP) is the maximum amount of unacknowledged bandwidth that can prevail over a network at a given point in time.

BDP= BottleNeck Bandwidth * Round Trip Time (RTT)

Bottleneck Bandwidth= 5 Mbps



RTT= 25ms + 25ms= 50 ms

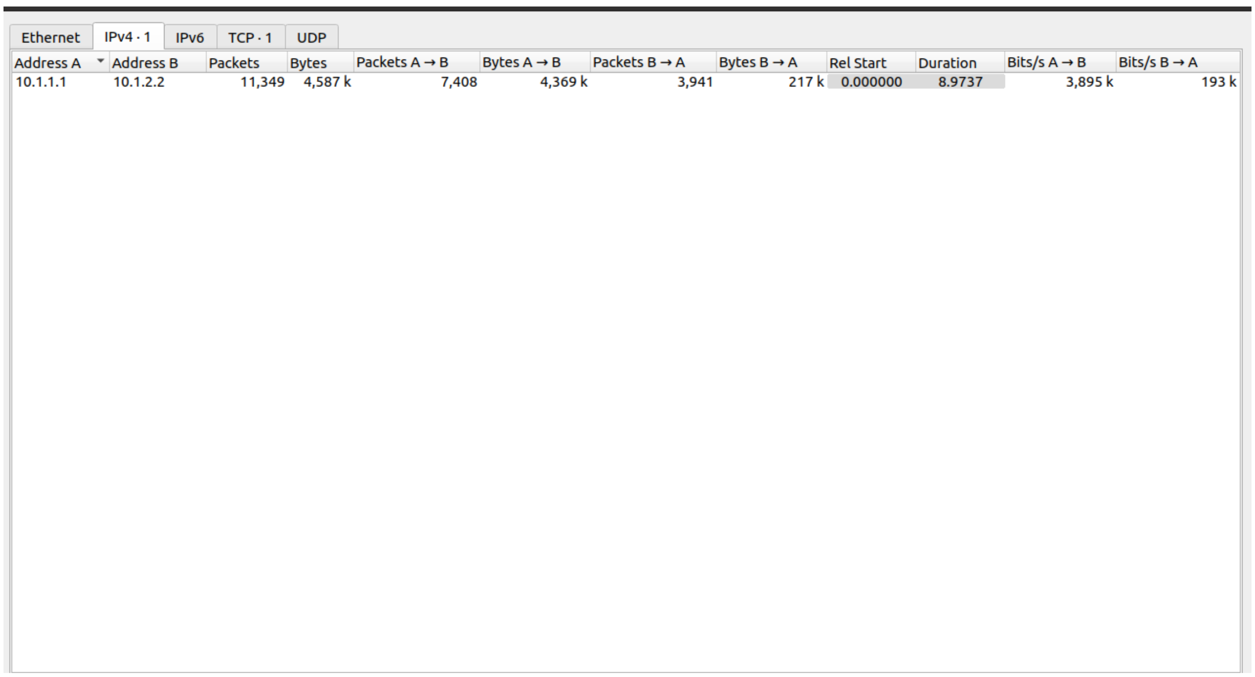
BDP= 5Mbps * 50ms = 0.25 Mb= 250000 bits

BDP=   250000 bits/ 1460 bytes {Packet size}

= 250000 bits/ 1680 bits{Packet size}

= approx 21.4 packets

Part(c)



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

Bits/s A -> B = 3895 k

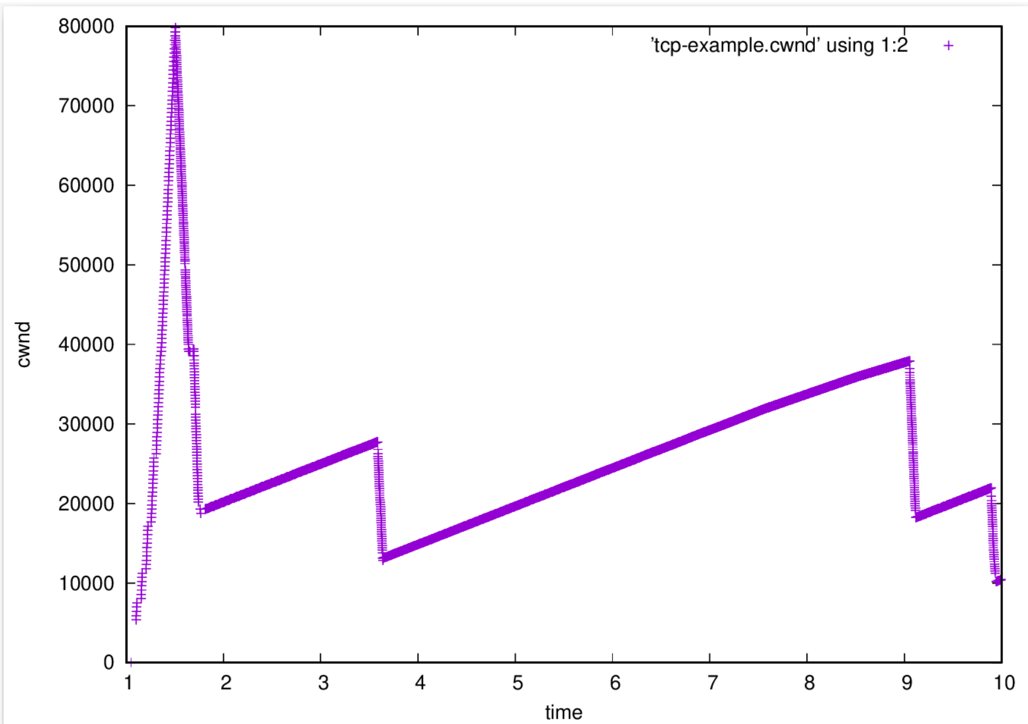
Bits/s B -> A = 193k

Average Computed throughput= 3895 k+ 193k= 4088 kbps = **4.088 Mbps**

Part(d)

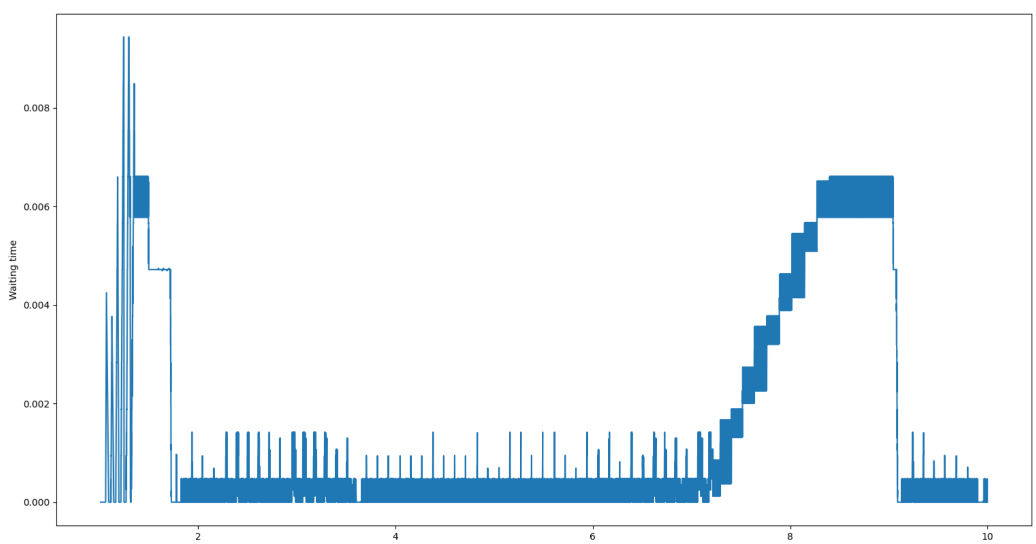
The achieved throughput in the stimulation is 4.088 Mbps, and the maximum expected throughput is 5 Mbps, as it is visible that the achievable throughput is less than the expected throughput. The difference is because of the queuing delays and the error that occur in the network. So when we calculated the expected throughput, we only considered the bandwidth of the links in the network. We didn’t take into account the delay that gets introduced to the network when the congestion occurs. Delays are also introduced when a packet has to wait long in the router’s queues or the end nodes. During the calculation of the expected throughput, we also didn’t take into account the delay caused due to an error in transmission. Hence we expected a greater throughput than the actual throughput as the actual throughput took into account all these delays.

Part (e)



'tcp-example.cwnd' using 1:2
cwnd
time
80000
70000
60000
50000
40000
30000
20000
10000
0
1
2
3
4
5
6
7
8
9
10

Part (f)



Waiting time
Dequeuing time
0.008
0.006
0.004
0.002
0.000
2
4
6
8
10

Part(g)

Yes, both the plots in part(e) and part(f) are related. The graph plotted at part(e) represents the congestion window size at different time intervals. The graph plotted in part(f) plots points when a packet is dequeued at a time "t" after waiting for time "w" from the time it was enqueued, so the points on the graph
corresponding to "t" on the x-axis and "w" on the y-axis.
The congestion window size is increased incrementally. So we can say that as the congestion window size increases, more numbers of unacknowledged packets can be sent in one RTT, this means that more packets at a time reach the router thus it increases the router’s queue length. This is precisely what we are seeing. So initially in the start stimulation is at slow start phase, thereafter each RTT the window size doubles, hence we see a sudden spike in cwnd values from time 1 to 1.5 time in the graph of part(e), we see a similar sudden rise in the graph of part(f) since more and more packets sent to the network increases hence the queue at the router’s end also increases. When the congestion in the network increases because of higher queue lengths in the network that is visible by the graph of the part(f) which reach a peak at time 1.5. The sender node detects this congestion and in order to reduce congestion, it reduces the cwnd size by half, because of which we see a sudden dip in the cwnd value after time 1.5. So similar kind of behavior is seen with the rest of the graph.
With this behavior, we can say that the change in congestion window size is proportional to the queuing delay. This is because when cwnd size increases, more packets are sent to the network at a time, which increases the size of the queues, so the packets have to wait for a long time at these big queues, hence it increases the queuing delay of the packets.

Question 2)

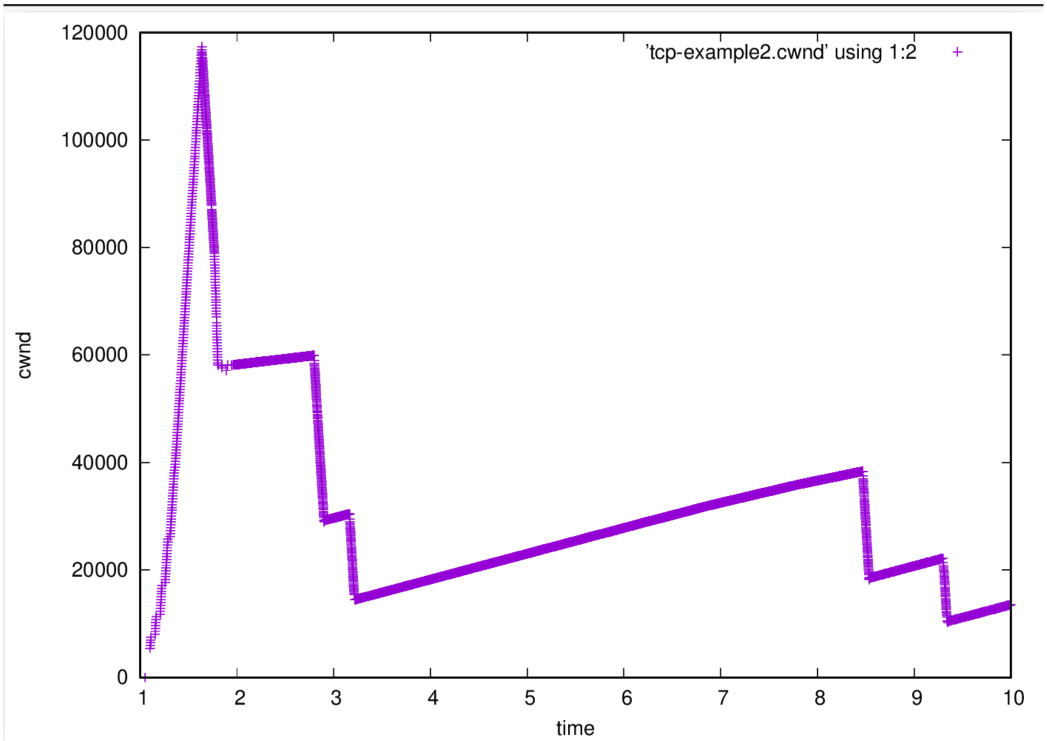
Part(a)

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

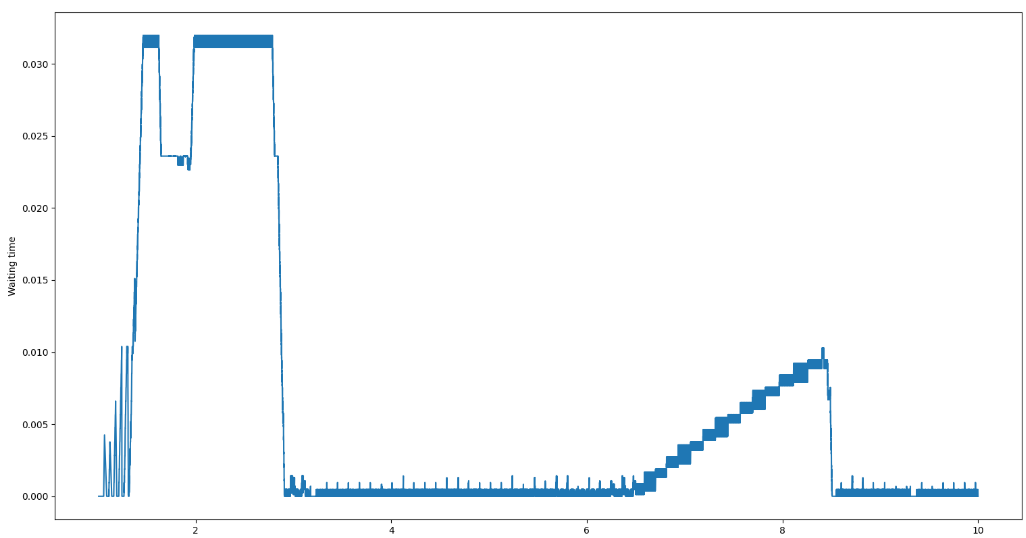
Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
10.1.1.1	10.1.2.2	11,800	4,745 k	7,656	4,514 k	4,144	230 k	0.000000	8.9746	4,024 k	205 k

Bits/s A -> B = 4024 k
Bits/s B -> A = 205 k
Average Computed throughput= 4024k+ 205k= 4229 kbps = **4.229 Mbps**

Part(b)


'tcp-example2.cwnd' using 1:2
cwnd
time
120000
100000
80000
60000
40000
20000
0
1
2
3
4
5
6
7
8
9
10

Part(c)


Waiting time
Dequeuing time
0.030
0.025
0.020
0.015
0.010
0.005
0.000
2
4
6
8
10

Part(d)

The cwnd plot of part(e) of Q1 takes a maximum value of 80000 and the network in that question had a queue length of 10. But in Question 2 the queue length is 50 and the cwnd plot of part(b) takes the maximum value of 120000. This simply implies that higher than queue size, the congestion window can take greater values before the congestion avoidance stage is reached. This implies that we can tolerate higher congestion. Hence higher queue size means higher tolerance to congestion. So when we are allowing higher congestion tolerance this means that larger size queues are formed in question 2 than in question 1. So the packets waiting in the queue for a longer time as compared to the waiting time in the queue for packets in question 1. So this implies that queuing delay is greater per packet for question 2 is greater than the queuing delay for question 1. Since this queuing delay directly indicates the congestion of a network, that's why we see higher congestion values for question 2 than in question 1and this is also reflected in their graphs.

Question 3)

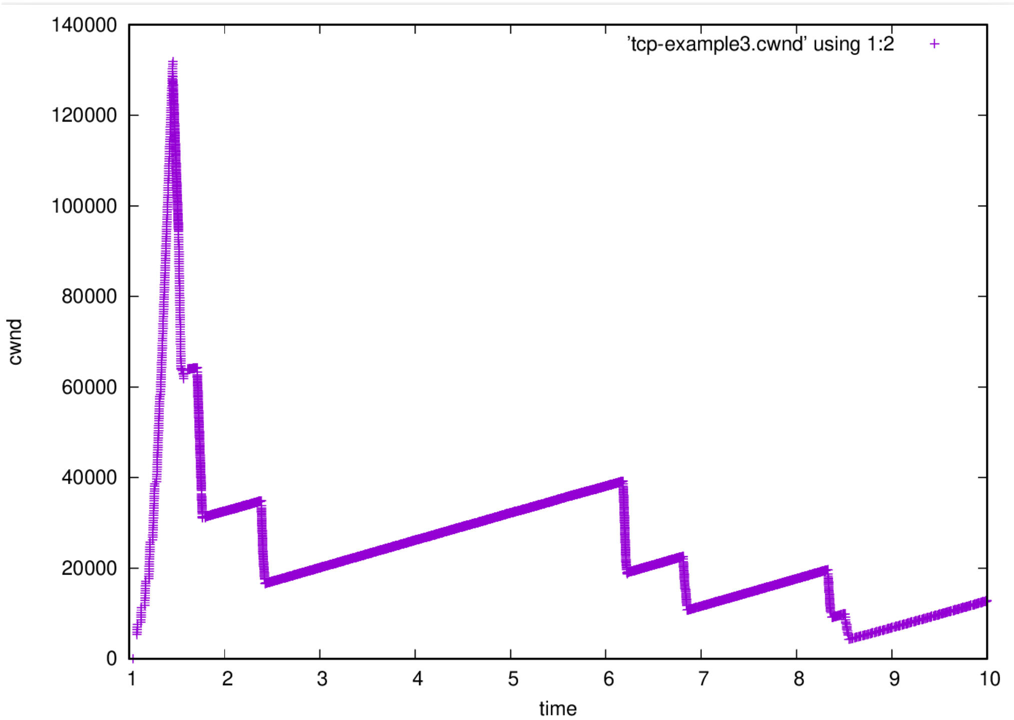
Part (a)

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

Address A	Port A	Address B	Port 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	49153	10.1.2.2	8080	13,868	5,562 k	8,981	5,292 k	4,887	270 k	0.000000	8.9746	4,717 k	240 k

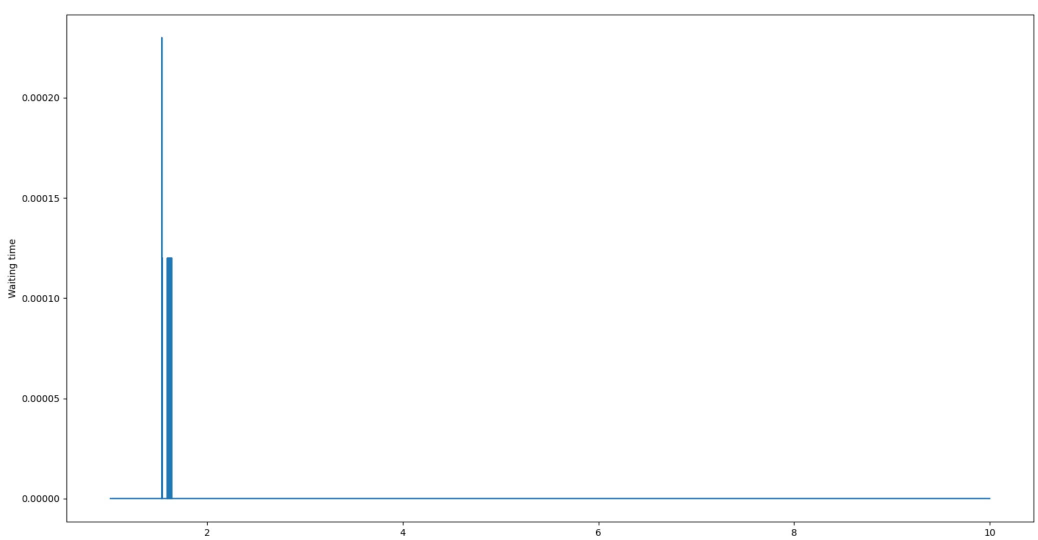
Bits/s A -> B = 4717 k
Bits/s B -> A = 240 k
Average Computed throughput= 4717k+ 240k= 4957 kbps
= **4.957 Mbps**

Part (b)



'tcp-example3.cwnd' using 1:2
cwnd
time
0
20000
40000
60000
80000
100000
120000
140000
1
2
3
4
5
6
7
8
9
10

Part(c)



Waiting time
Dequeuing time
0.00000
0.00005
0.00010
0.00015
0.00020
2
4
6
8
10

Part (d)

The network stimulation in Question 1 has a bandwidth of n0-n1 as 10 Mbps and n1-n2 is 5 Mbps. So the bandwidth of the incoming connection link is greater than the bandwidth of the outgoing link. So the sender can send a greater number of packets to n1 but n1 can send out packets at a lower rate than it is receiving them. This would increase the queue at n1 and more packets have to wait in a queue to send out by n1.

The network stimulation in question 3 had a bandwidth of n0-n1 as10 Mbps and n1-n2 as 10 Mbps. This implies that the bandwidth of the incoming connection is equal to the outgoing bandwidth link. So the sender can send an equal number of packets to n1 as n1 can send out packets to n2. Thus the packets would not get queued in n1. Hence the queue length of n1 would be significantly less.

So from the above situation, we can say that the queue length in n1 in question 1 stimulation is greater than the queue length of n1 in question 3. Due to this, it logically implies that the queuing delays of packets in question 1 are greater than the queuing delays of packets in question 3. This is what we can see in their respective graphs. As the queuing delay graph in question, 1 takes high values over different time periods but the queuing delay graph in question 2 takes very lower values.

Question 4:

Part(b)

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

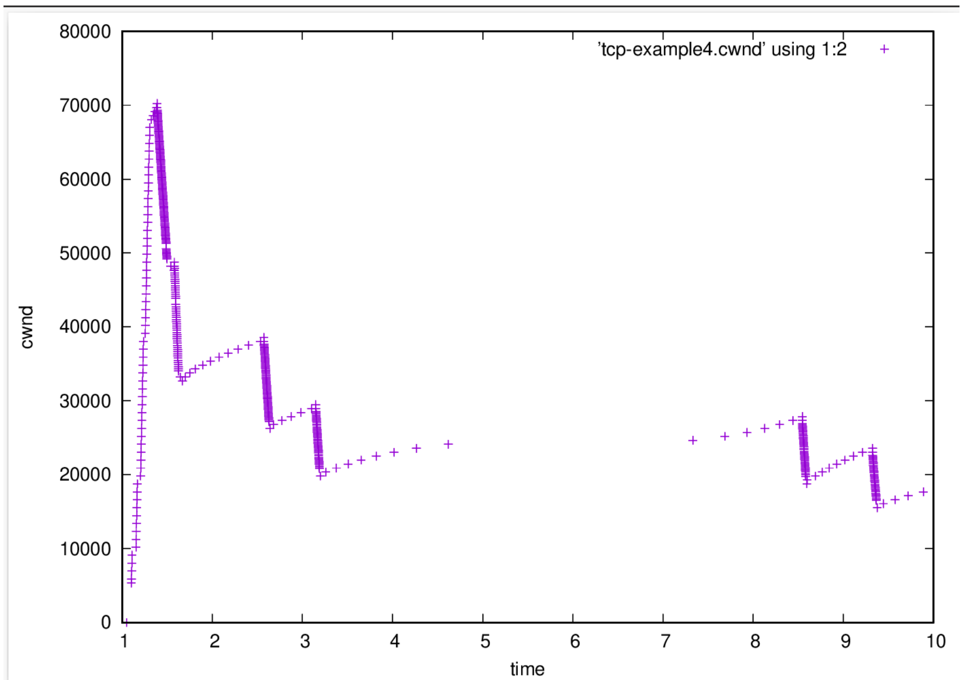
Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
10.1.1.1	49153	10.1.2.2	8080	11,944	4,816 k	7,774	4,585 k	4,170	230 k	0.000000	8.9743	4,087 k	205 k

Bits/s A -> B = 4087 k

Bits/s B -> A = 205 k

Average Computed throughput= 4087k+ 205k= 4292 kbps = **4.292 Mbps**

Part(b)



Part(c)

The main difference between the TCP New Reno and TCP Cubic is in the way they adjust the cwnd window. The slow start and fast recovery phases are the same for both but the congestion avoidance stage is different. In TCP new Reno in the congestion avoidance phase increases the congestion window size increases linearly as TCP new Reno follows the AIMD (Additive Increase and Multiplicative Decrease). But in TCP cubic the cwnd window size increases in cubic fashion. So throughput given by TCP cubic is greater than the TCP New Reno as when there are changes in the bottleneck bandwidth of the network, TCP Cubic is able to increase the congestion window size cubically rather than linearly which the TCP New Reno does. This is precisely what we see in the CWND graph of question 1 and question 3.