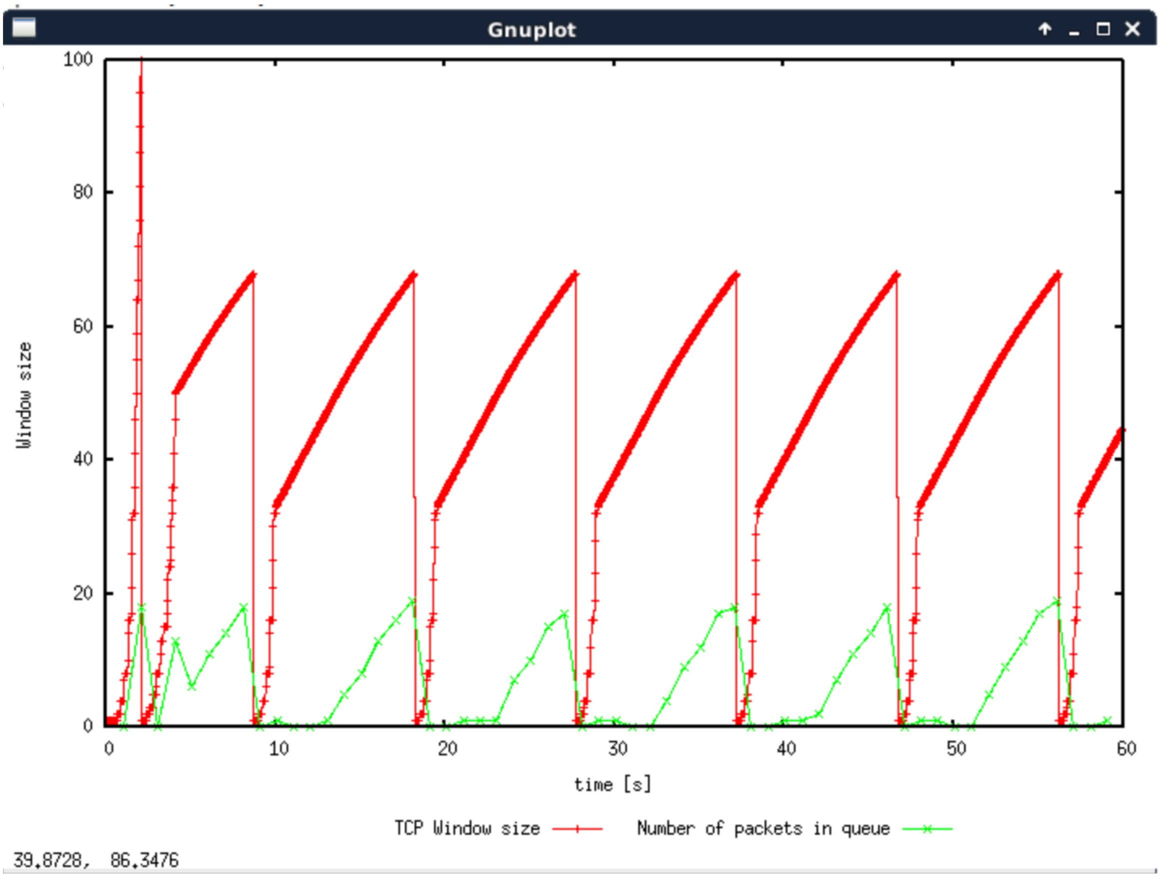
# COMP9331 LAB 05

### YUAN GAO Z5239220

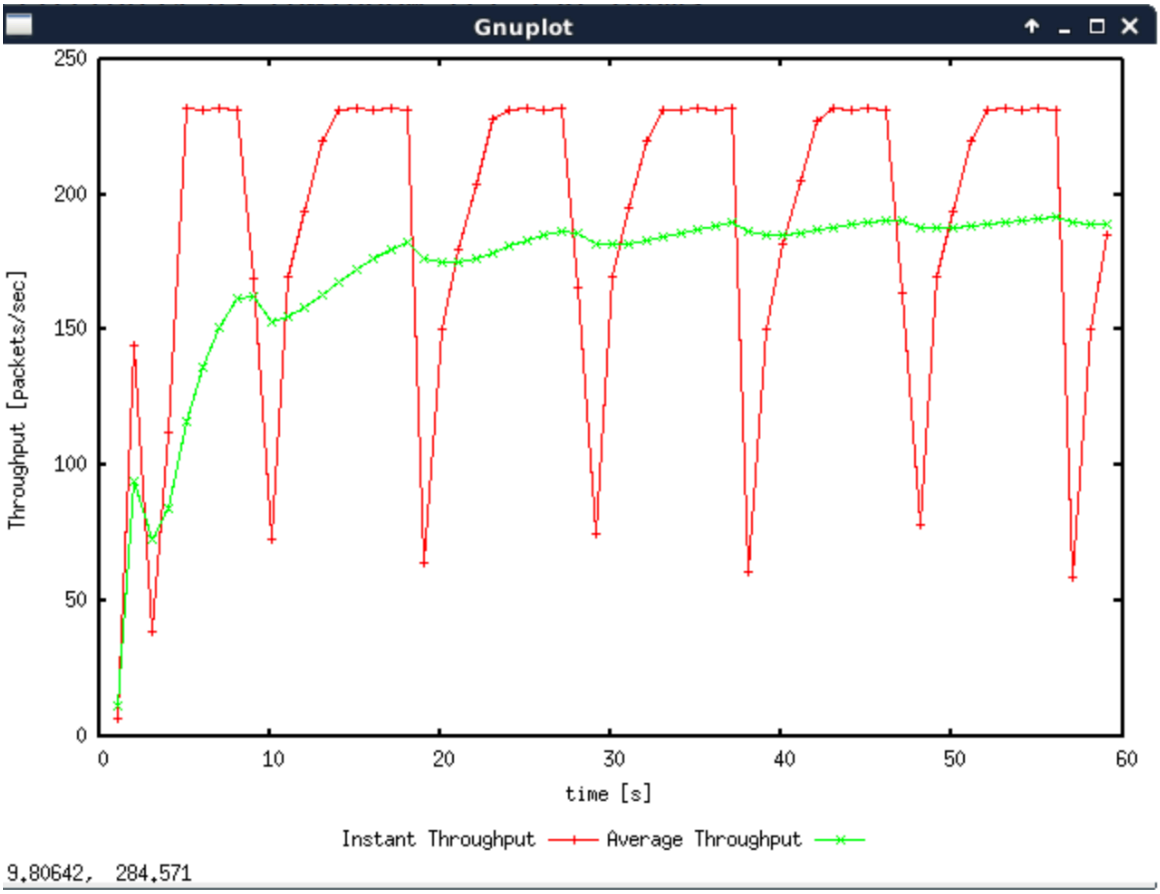
## Exercise 1: Understanding TCP Congestion Control using ns-2

#### Question 1




- The maximum size of the congestion window that the TCP flow reaches in this case is 100
- Because the maximum size of queue is 20, any additional packets whitch more than 20 will be dropped
- When it reached to 100MSS, the queue was full and the packets dropped whitch results in a congestion event. The sender stoped inctreasing the congestion window size and reseted the window size as 1 MSS, the threshold as 50 packets. Then, The connection enters slow start. When the congestion window reached threshold congestion avoidance phase such as AIMD was uesed to aviod congestion. Afterwards, repeat the process.

#### Question 2

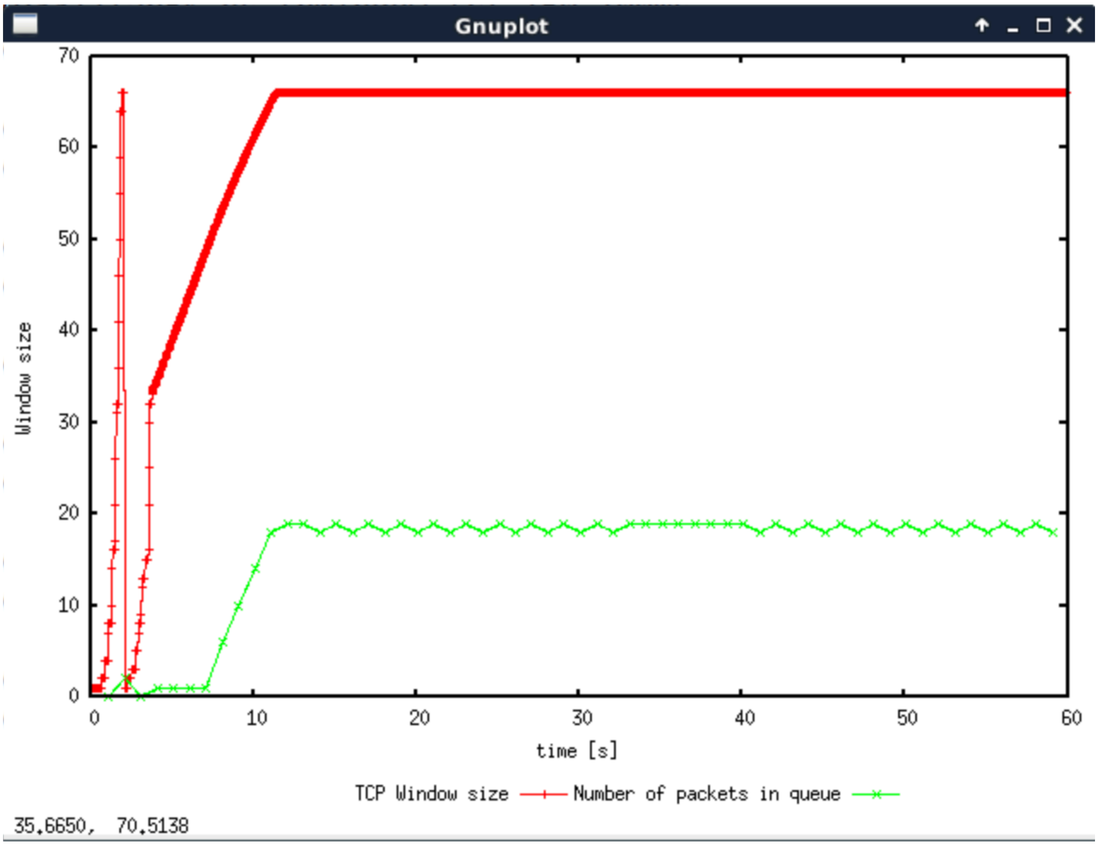



- The average throughput of TCP is around 190 pps.
- Acrodding to the graph, after 20s, the average throughput is relatively stable . Therefore, The size of pakets is $500 + 20 + 20 = 540\ btyes$

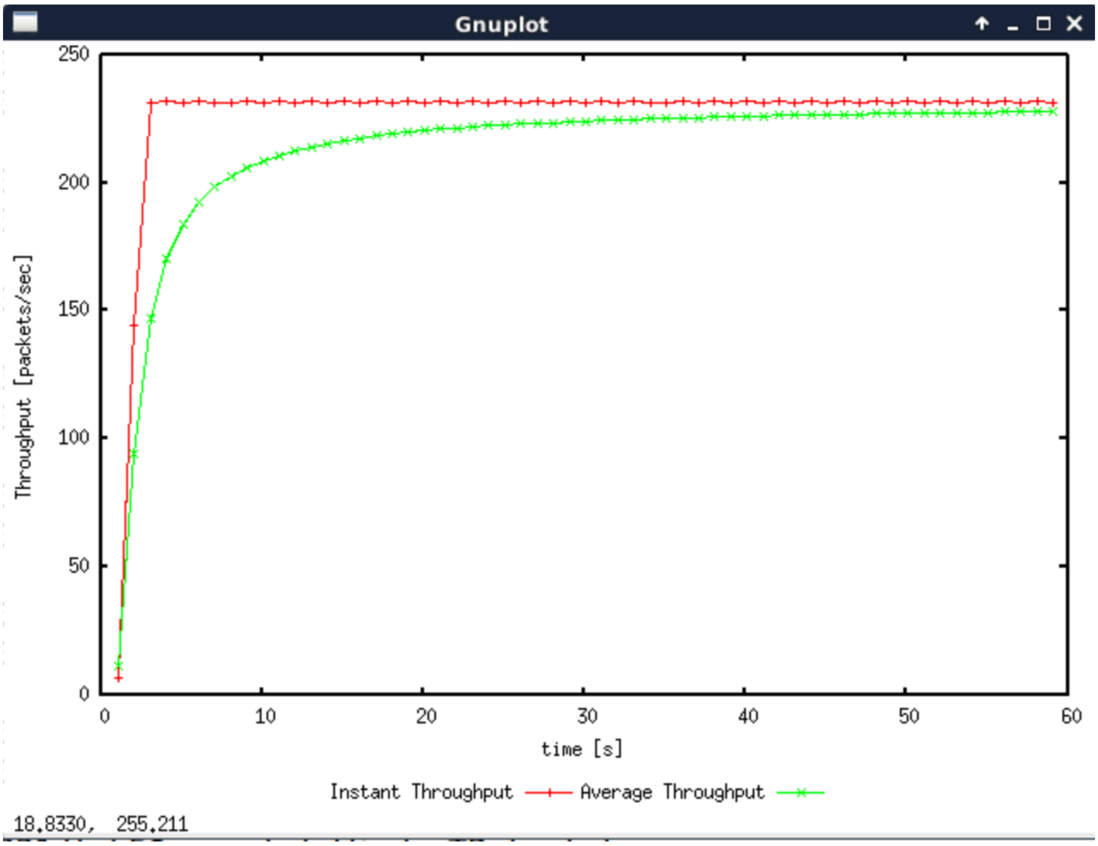
  The throughput is $540 \times 190 \times 8 = 820.8\ kbps$

**Question 3**

- When the maximum congestion window size $<=$ 66, TCP stops oscillating to reach a stable behaviour. The window size stops after returning to the slow start for the first time. When we reduce the congestion window to half, this is enough to stabilize the number of packets in the send queue. The queue has never full, so these no packers are dropped.




- When the initial max congestion window is set to 50,  TCP stabilized immediately.

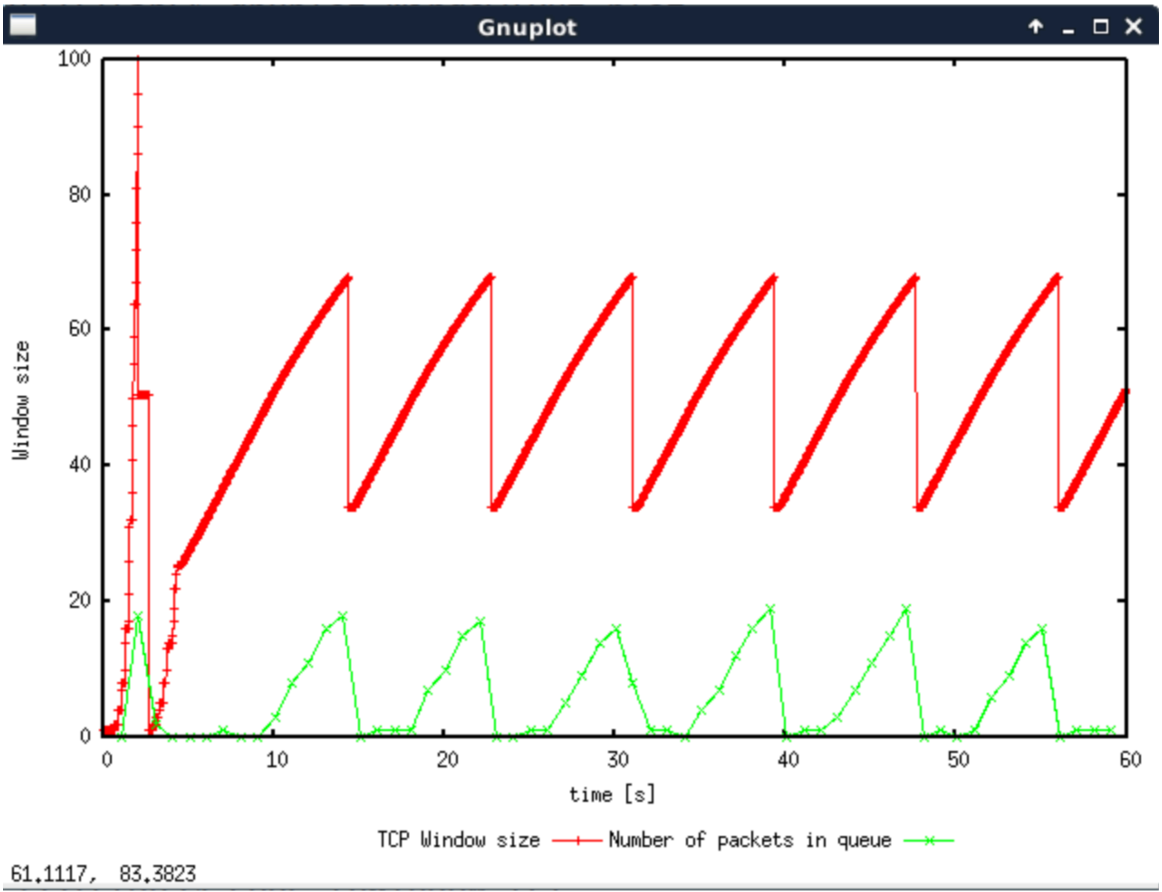



- The average packert throughput is around 225 pps.

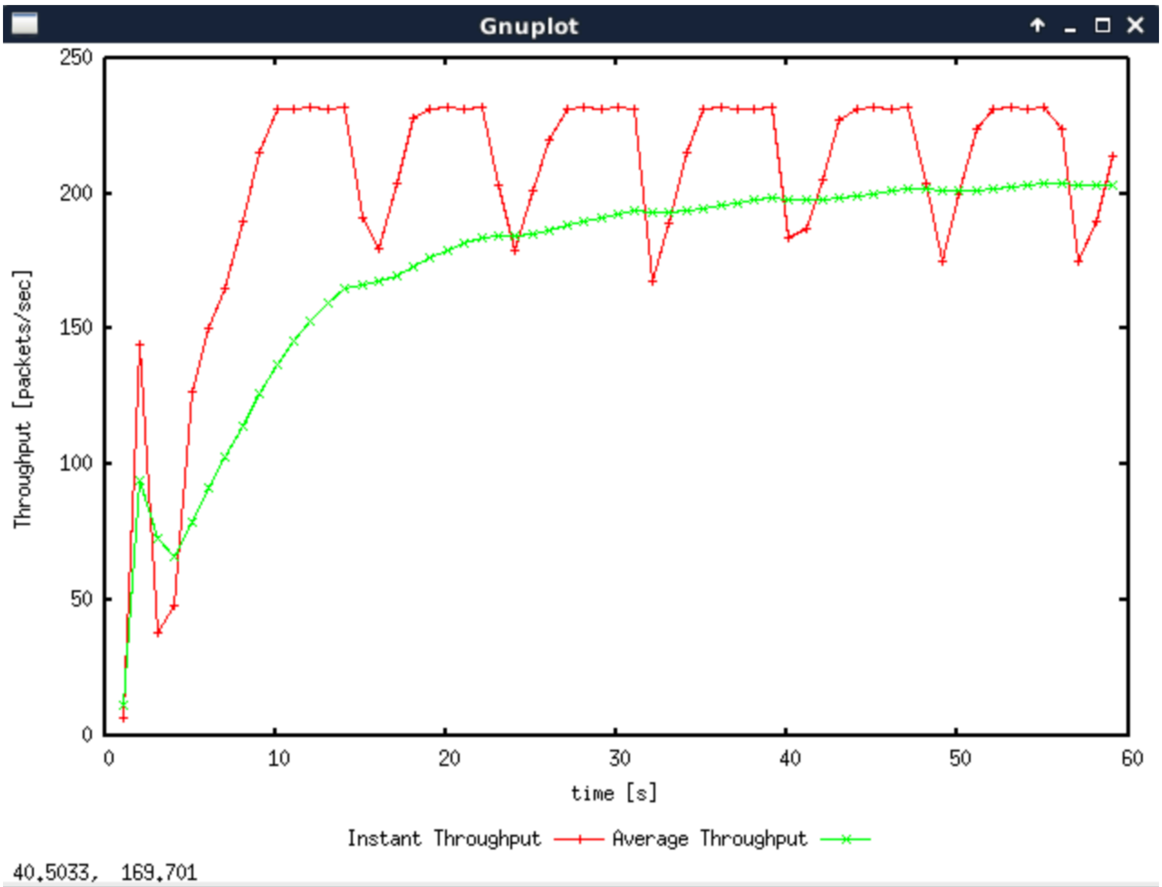
  The throughput is $255 \times 500 \times 8 = 900\ Kbps$

  It is almost equal to the link capacity.

**Question 4**

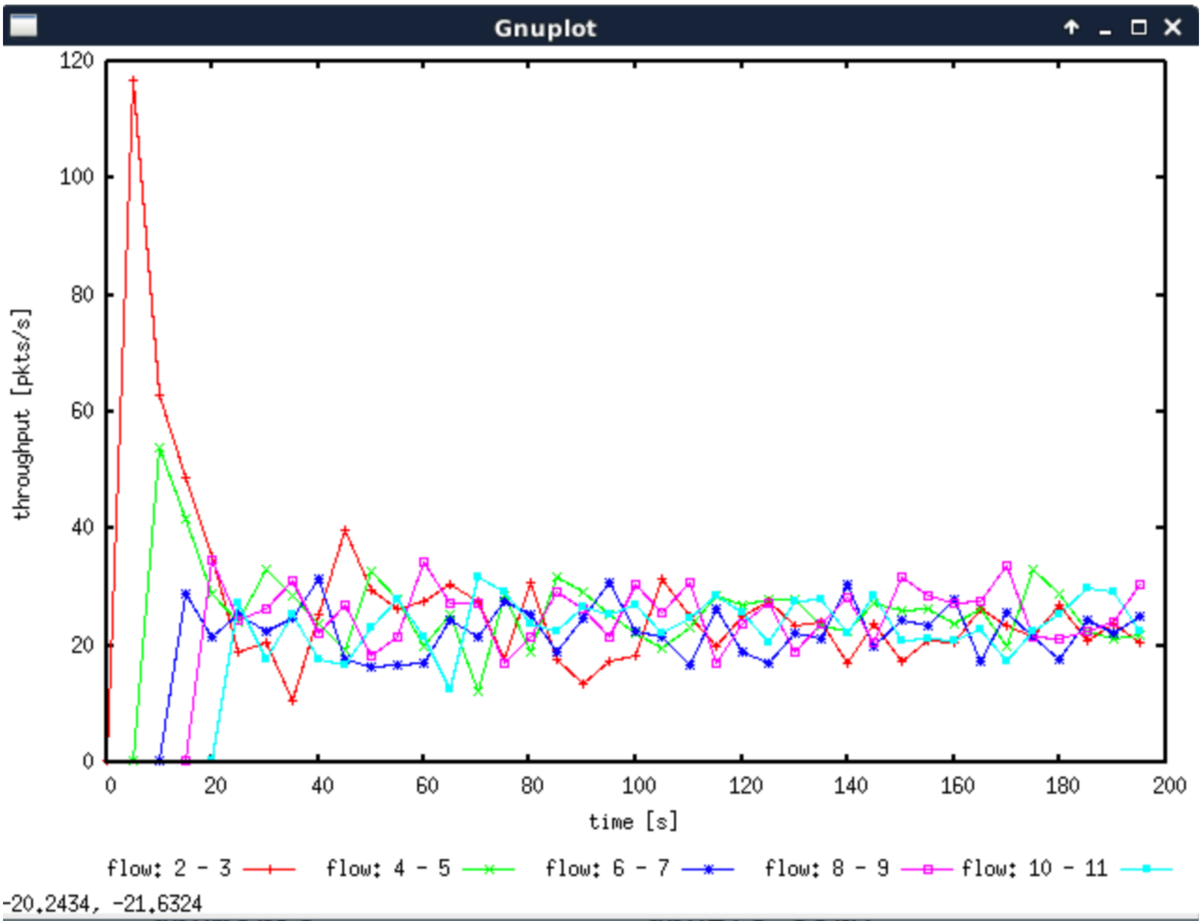



- The sender halves its current confestion window and increases it linearly, until losses. Afterwards, repeat the process. This because in TCP **Reno** method, most of the losses are detected due to triple duplicate ACKs. It is different to TCP **Tahoe** method. After each congestion event of TCP Tahoe, the window is reduced to 1.



> - In TCP **Tahoe**, there are around 190 pps. The throughput is $540 \times 190 \times 8 = 820.8\ kbps$
>
>   In TCP **Reno**, there are around 200 pps. The throughput is $540 \times 200 \times 8 = 863\ kbps$
>
>   The throughput of **Tahoe** is slightly higher.

## Exercise 2: Flow Fairness with TCP




### Question 1

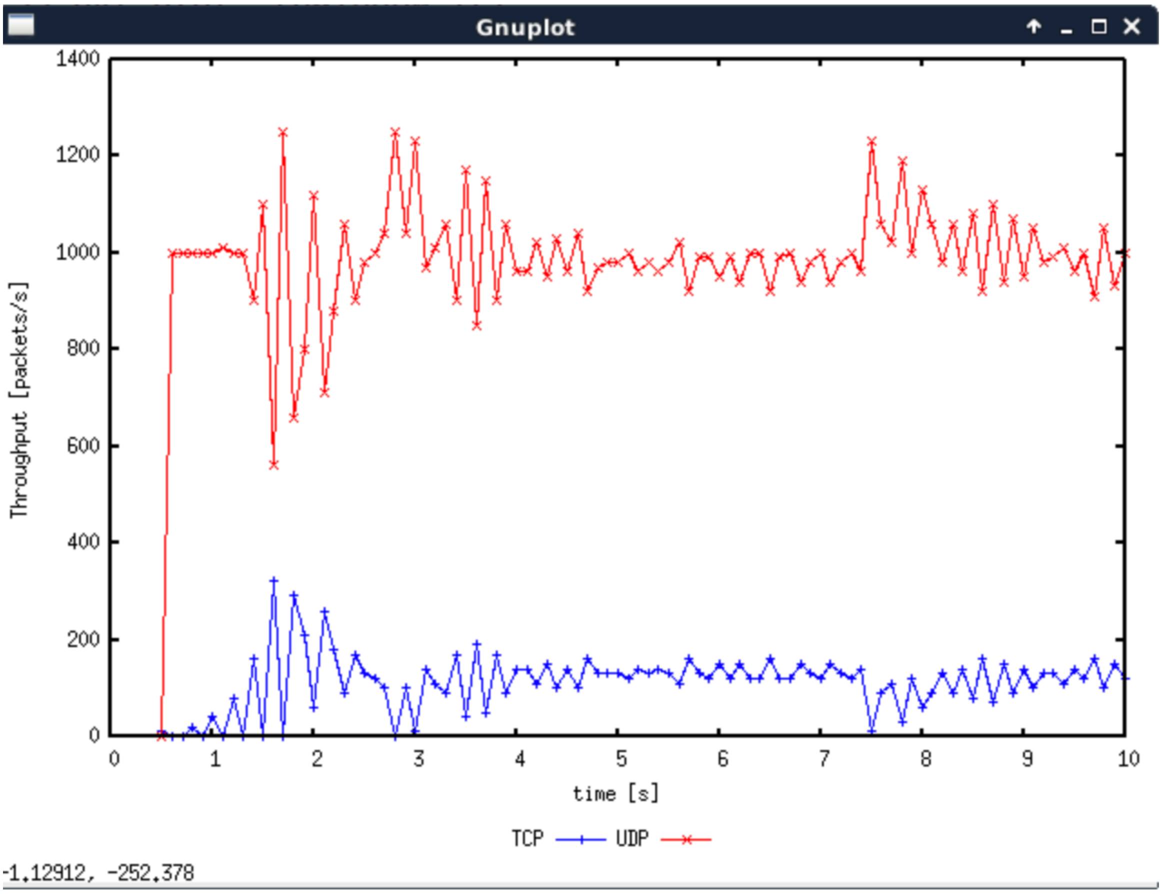
> - Yes, each flow get an equal share of the capacity of the common link. Accrodiong to the graph, the five connections have commenced. These have similar throughput.
>
>   Because when using the AIMD congestion control algortihm, window size can achieves long-term fairness.

### Question 2

> According to the graph, when a new flow is created, all go throughout pre-existing flows is reduced. This is because that during the slow start, when the a new flow adds in, the new flow will increase quickly abd result congestion. Then, every TCP connections will adapt the size of their congestion window to avoid network congestion. This behaviour is fair, Because once new traffic is joined in, all existing traffic is reduced accordingly.

# Exercise 3: TCP competing with UDP

### Question 1




- According to the picture, when the capacity of the link is 5 Mbps, the throughput of UDP is larger than TCP.

### Question 2

> This is becuse UDP dose not have congestion control. Hence UDP will transmit packets keeping a constant rate regardless of the situation of packets dropped. However, TCP has congestion control, packets dropped affect rate. Therefore, TCP flows will be forced to use a lower throughput due to more aggressive UDP flows.

### Question 3

- Advantages:

  UDP does not have restrained. it can keep transmitting as a higher speed. The delay would be low.

- Disadvantages:

  UDP is unreliable. Therefore , if using UDP transfer file, it need a reliable application.

  UDP will increase  the burden of the network.

- If everybody started using UDP instead of TCP, the network may encounter congestion collapse.