UNSW-COMP-9331

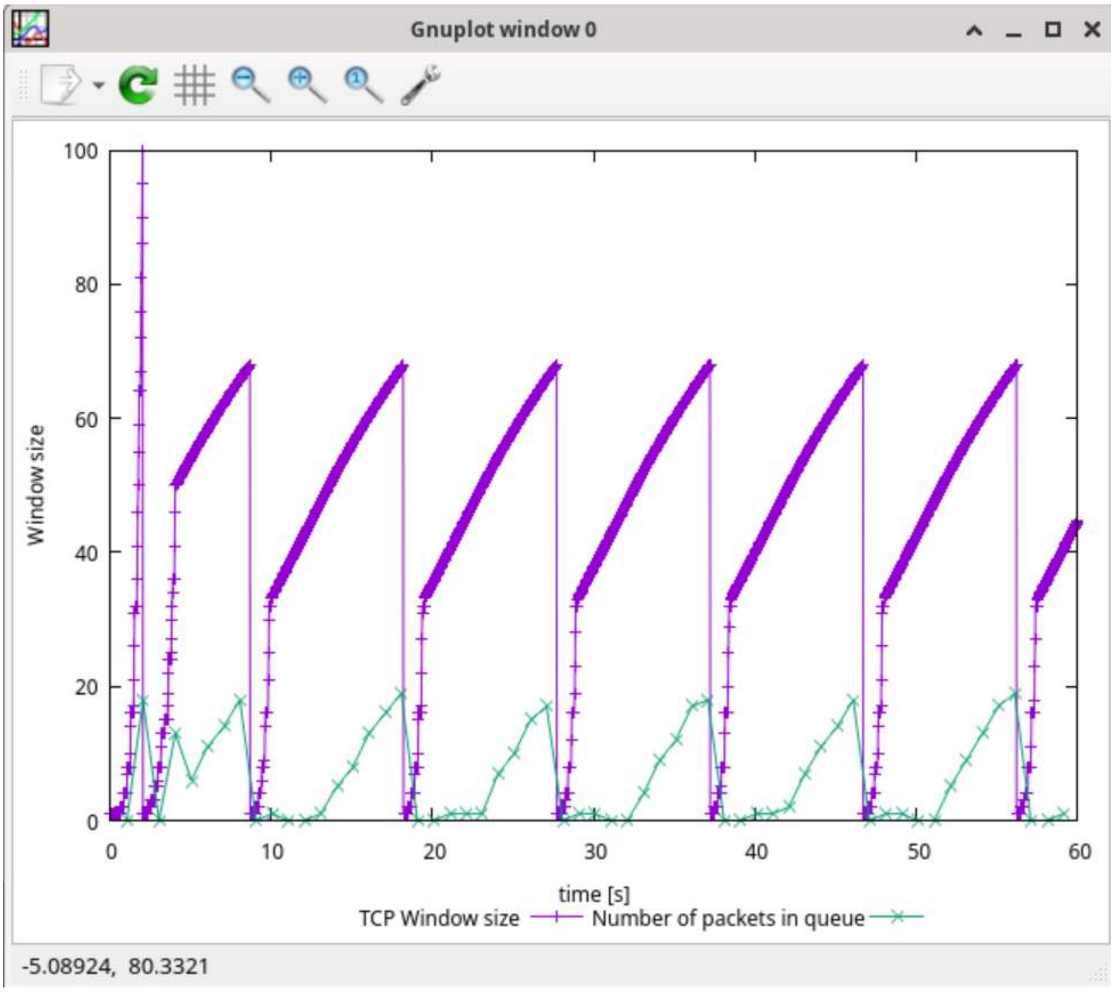
Tian Huang

20215-Nov-16

LAB 5

Exercise 1:

Question 1:



Part A:

Maximum value about 68-70 packets

50 packets in 1 Mbps link with 200ms RTT + 20 packets queue

Part B:

TCP detects the packet loss

The congestion windows drop to 1

The slow start threshold set to half of window size when loss

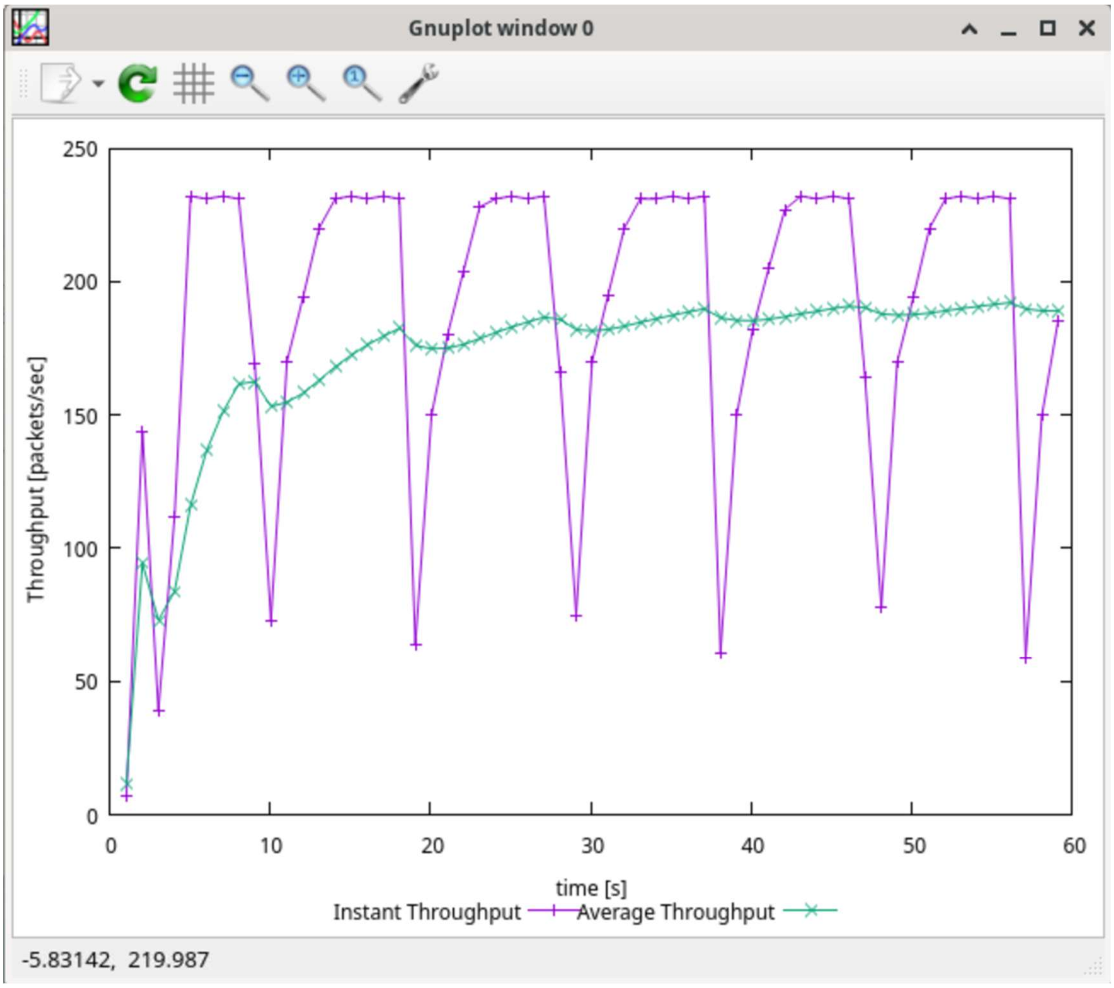
Part C:

When window collapses 1, sender slow start, congestion window grow

When it reach slow start threshold, switch to congestion avoidance

When windows increase and achieve 70 packets, queue fill up, queue fills, packet loss, windows drop to 1, everything repeat

Question 2:

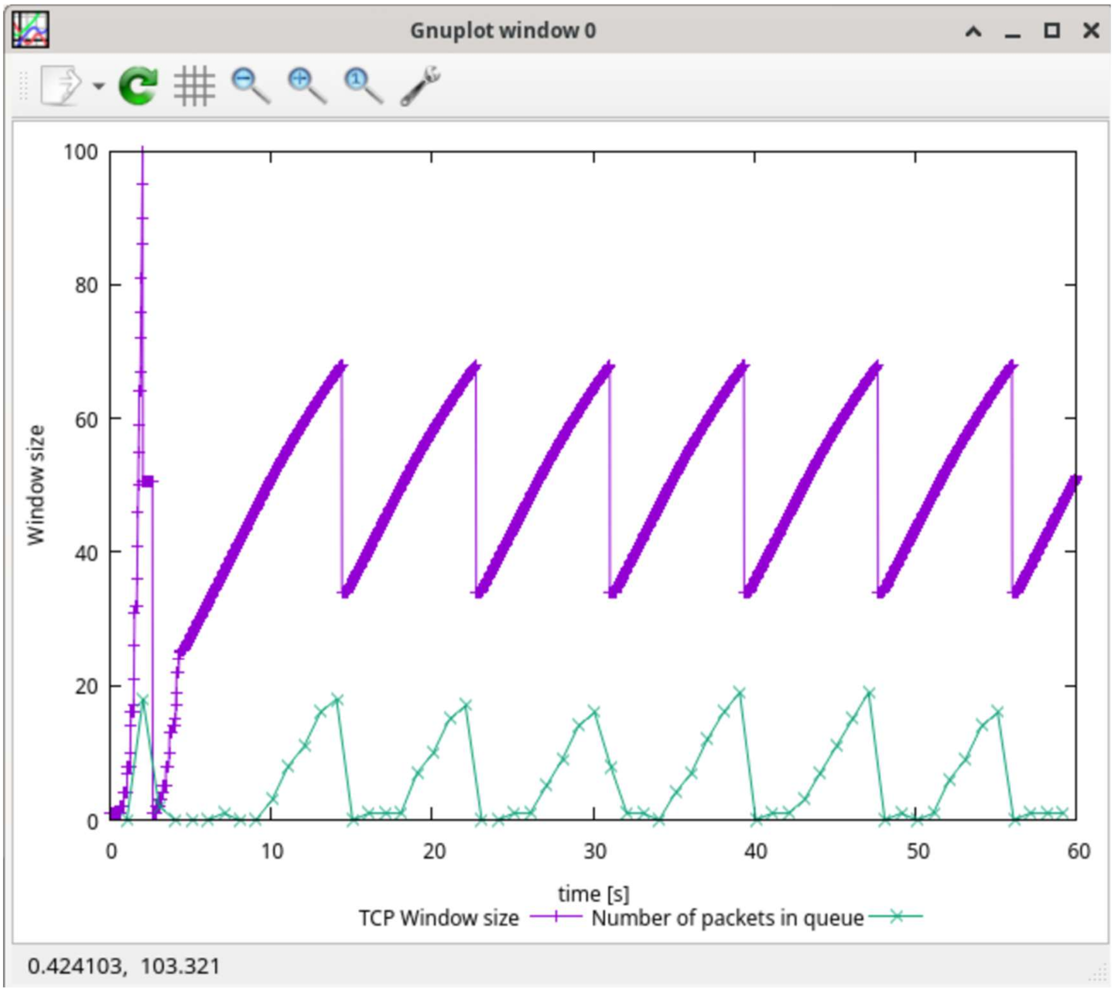


Average throughput 188 packet/s

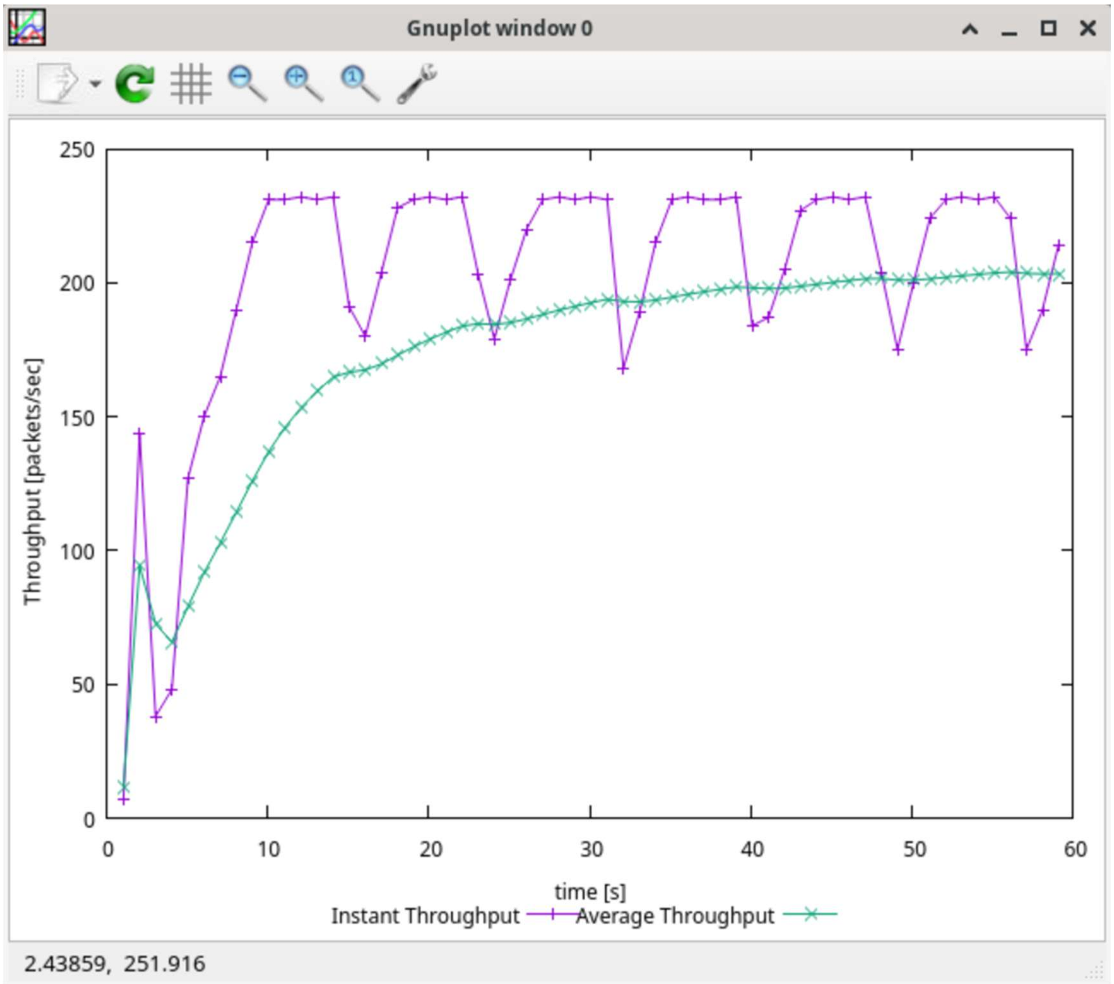
640 bytes = 4320 bits

188 * 4320 = 812160 bps = 812 kbps

Question 3:



Gnuplot window 0
100
80
60
40
20
0
Window size
0
10
20
30
40
50
60
time [s]
TCP Window size
Number of packets in queue
0.424103, 103.321



In Tahoe, the window drop to 1 packet every loss

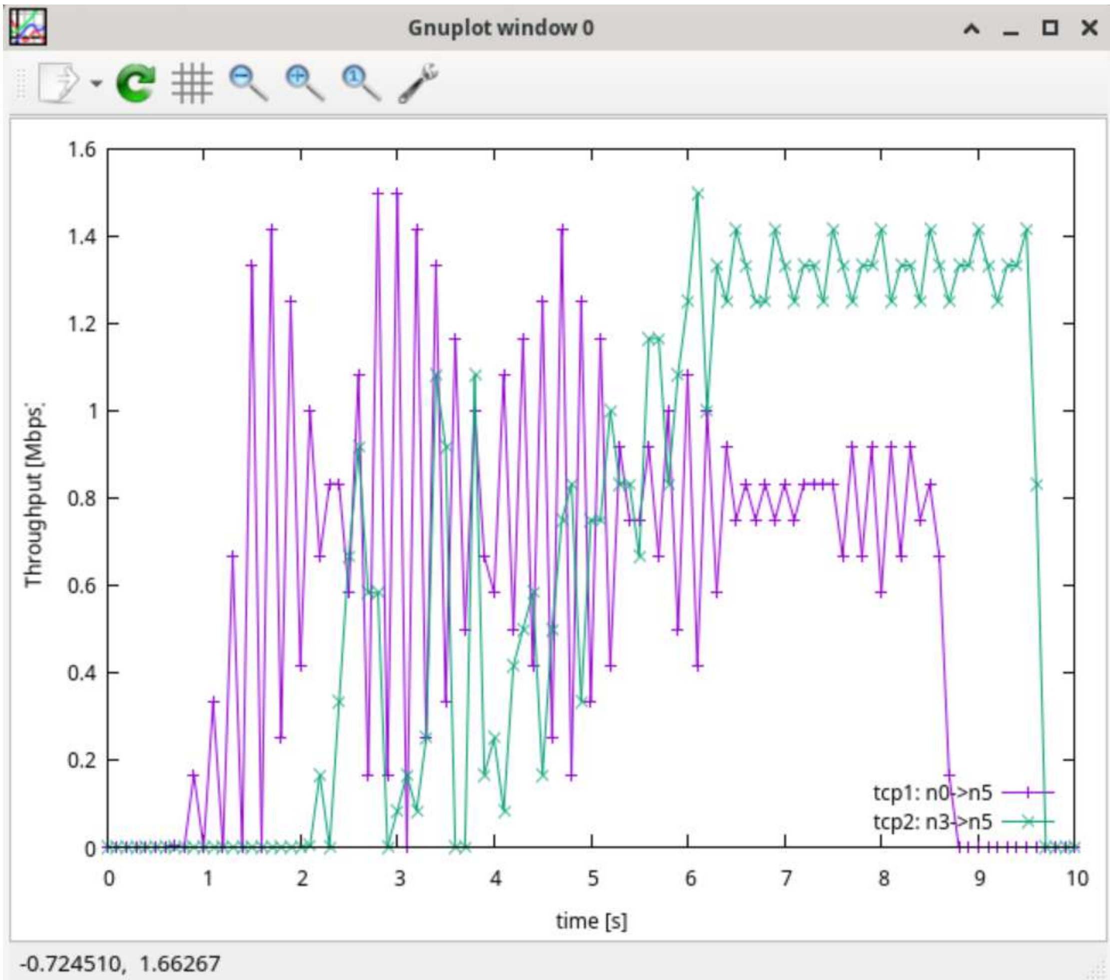
In Reno, when triple duplicate ACK, the congestion window drop to half of current size. The sender then fast recovery instead of slow start

Tahoe average throughput 188 packet/s or 812 kbps

Teno average throughput 205 packets/s or 864 kbps

Reno have higher throughput in that network condition

Exercise 2:



Question 1:

In 6-8s. the TCP from n7 to n0 is active, the flow shares link between n1 and n0 with tcp1. Both flows compete for same link capacity, tcp1 increase congestion, queue delay go higher, and occasional packet drop. So tcp1 throughput drop

Tcp2 does not traverse the congested n0 to n1, the path is independent and have less competition, so the throughput keep higher and stable

Question 2:

Tcp1 starts 0.5s and slow start. During slow start, the congestion windows increase exponentially, make throughput grow fast, and maybe over the capacity. At the same time, queue size and RTT is not stable causing shocks of throughput. So the shock here is tcp slow start situation.

Exercise 3:

Question 1:

N0 to n5

N2 to n5

It will not change

Question 2:

1.0s: n1 to n4 is down

0-1-4-5 will not change to 0-1-2-3-5, packet will be drop

2-3-5 will not been changed

1.2s n1 to n4 is back

It will not change.

Question 3:

1.0s: n1 to n4 down

DV detects the down, change to 0-1-2-3-5

It will change

1.2s: n1 to n4 is back

DV detects the change, 1-4-5 is only hops, it will change back

Question 4:

0-1-4-5: 1+3+1 = 5

0-1-2-3-5: 1+1+1+1 = 4

N1 to n4 will not been selected, because of high cost

N0 to n5 change to 0-1-2-3-5

DV detects the cost and switch to 0-1-2-3-5

2-3-5 will not been changed

Question 5:

Path A 0-1-4-5 1+2+3=6

Path B 0-1-2-3-5 1+1+1+3 = 6

The cost is same for both path

Multipath allows using multiple paths

N0 to n5 will go from both a and b

It will be multipath forwarding

DV will keep multiple paths

Traffic is more average in 2 ways