NS-2 simulation report

Team10
Li Wan UIN 824009186
Sama Avani UIN 724004372

1.Setup

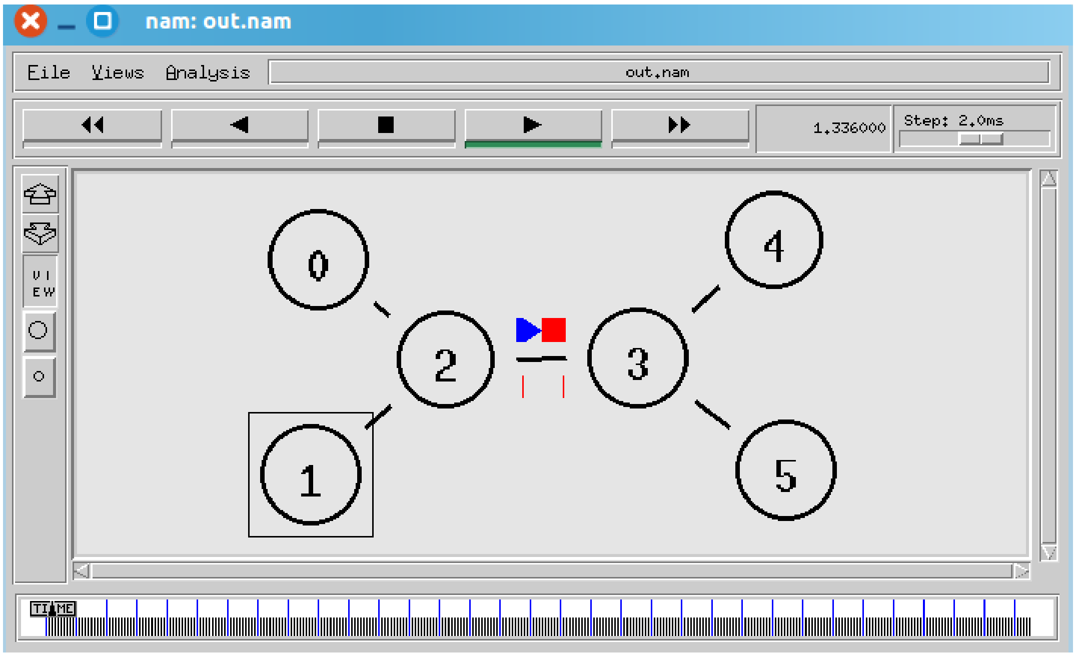
Ubuntu 14.04 (32 bits), NS-2, Xgraph, NAM, Tcl/Tk

2.Procesure

First, we need to install NS-2, NAM, Xgraph Packages on the Ubuntu. We then stat our simulation at the time 0, the intial process will be ran in the time 30.0 and the record process will start to work simultaneously. The whole simulation process will end at time 180 by calling the finish process. During the time 30-180, we call the record process to calculate the bandwidth from scr1-3 to sink 1-3 respectively. Here we simulate the following 2 scenarios:

Scenario 1:

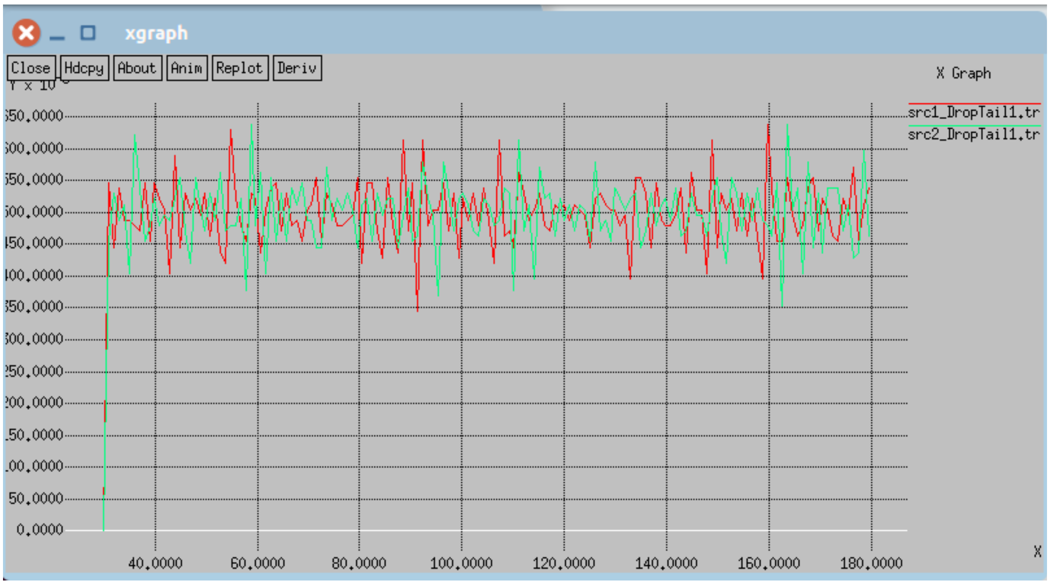
The topology is as shown in the graph below. We use two different buffer management strategies DROPTAIL and RED respectively. We set the time interval is 1 second in order to calculate the throughput.



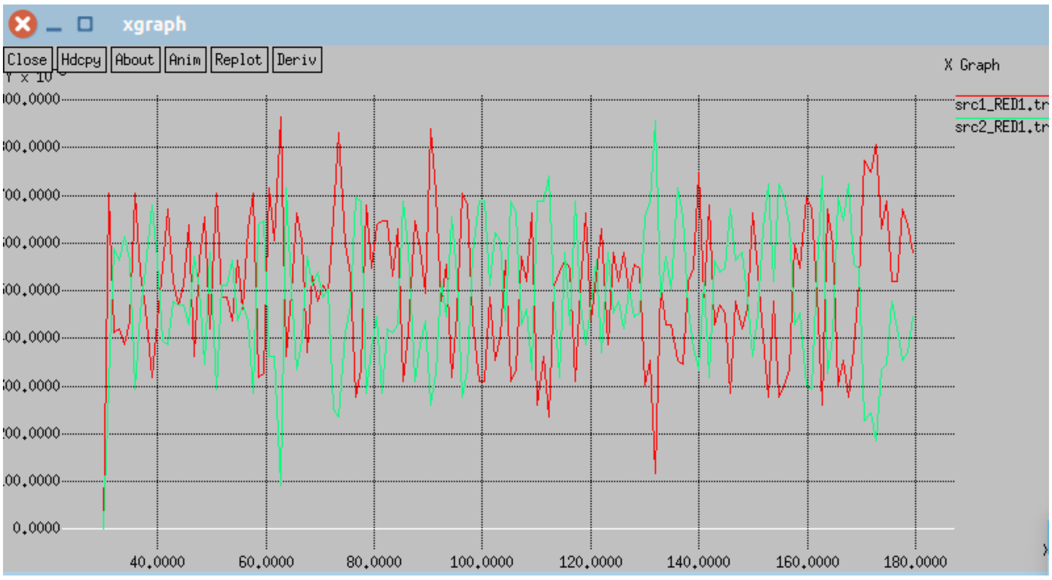
Average throughput for DropTail and RED

	Average Throughput Src1	Average Throughput Src2
DROPTAIN	0.4970 Mbps	0.4963 Mbps
RED	0.5055 Mbps	0.4868 Mbps

a) DROPTAIL buffer management: Instantaneous throughput

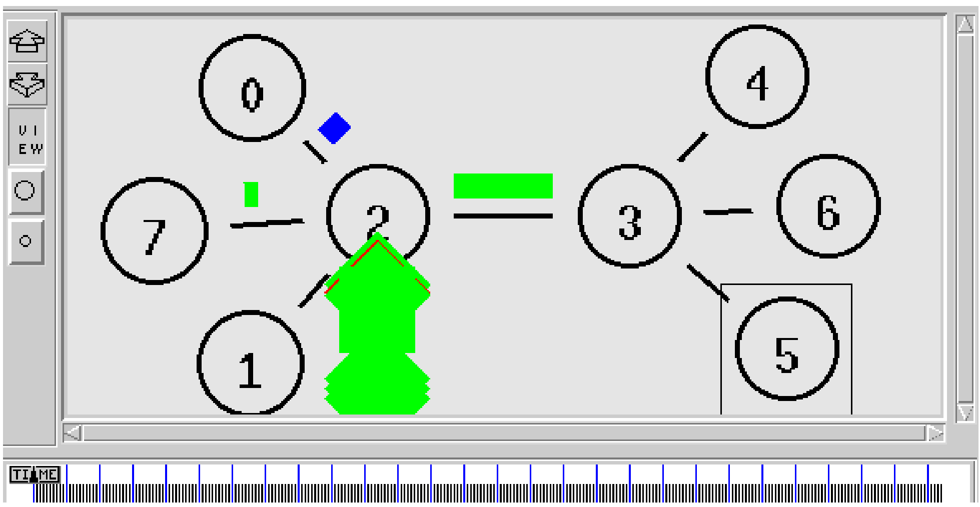


b) RED buffer management: Instantaneous throughput



Scenario 2:

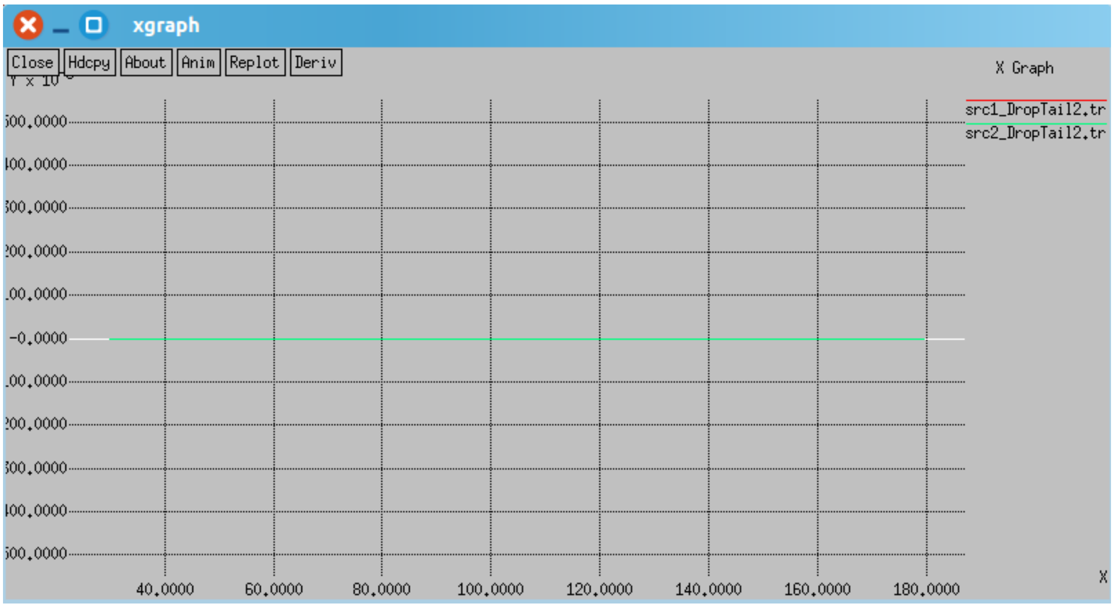
Here, we add two other nodes in this case, and add UDP source node. Now the topology is as shown below

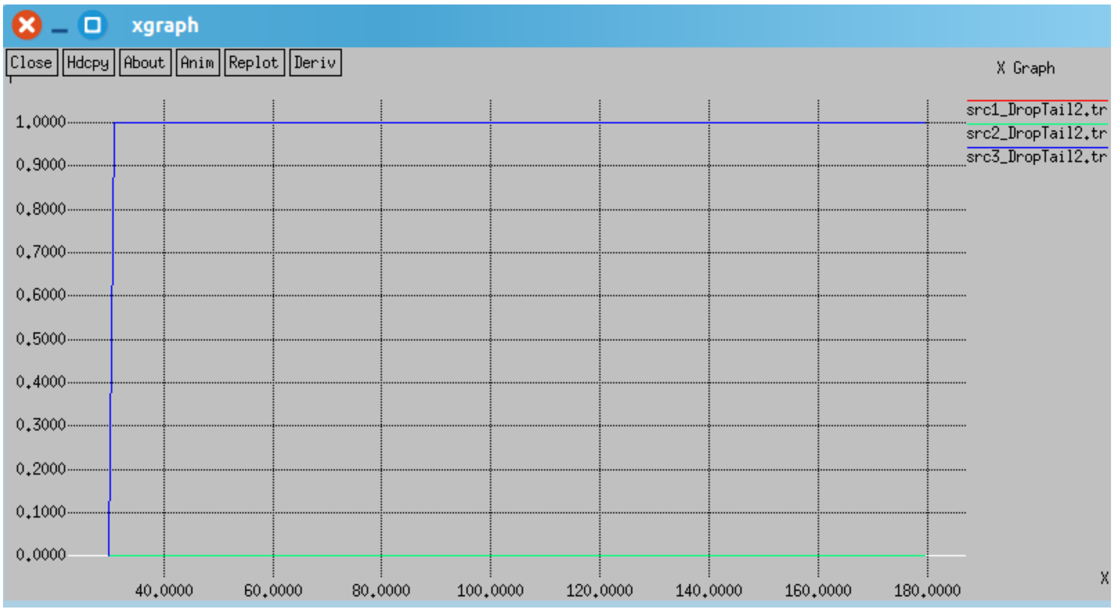


Average throughput for DropTail and RED

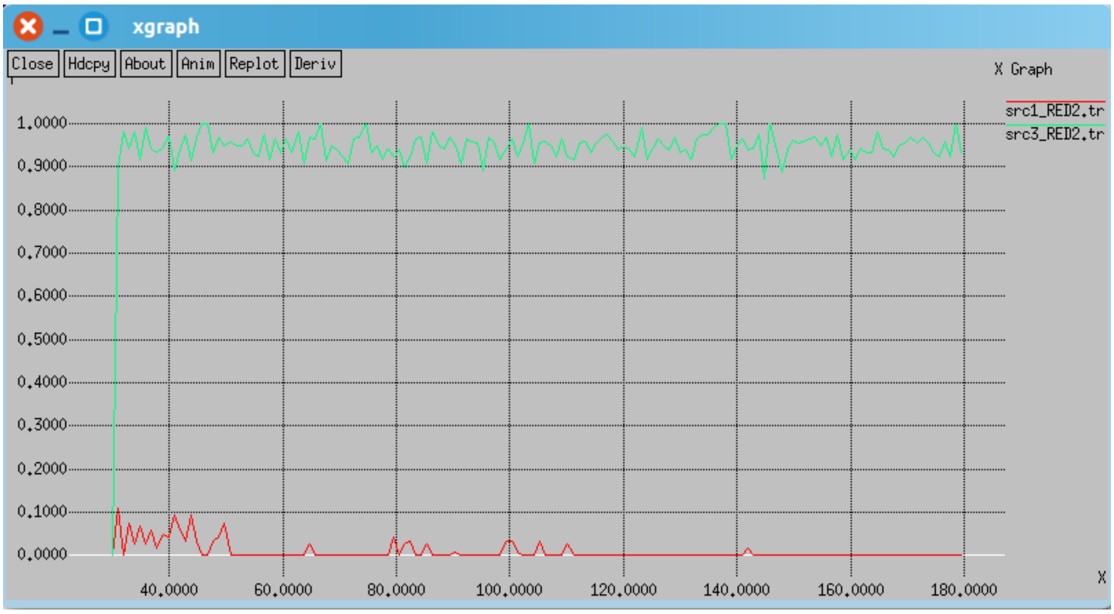
	Average Throughput Src1	Average Throughput Src2	Average Throughput Src3
DROPTAIL	0 Mbps	0 Mbps	0.9934 Mbps
RED	0.00818 Mbps	0.04014 Mbps	0.94261 Mbps

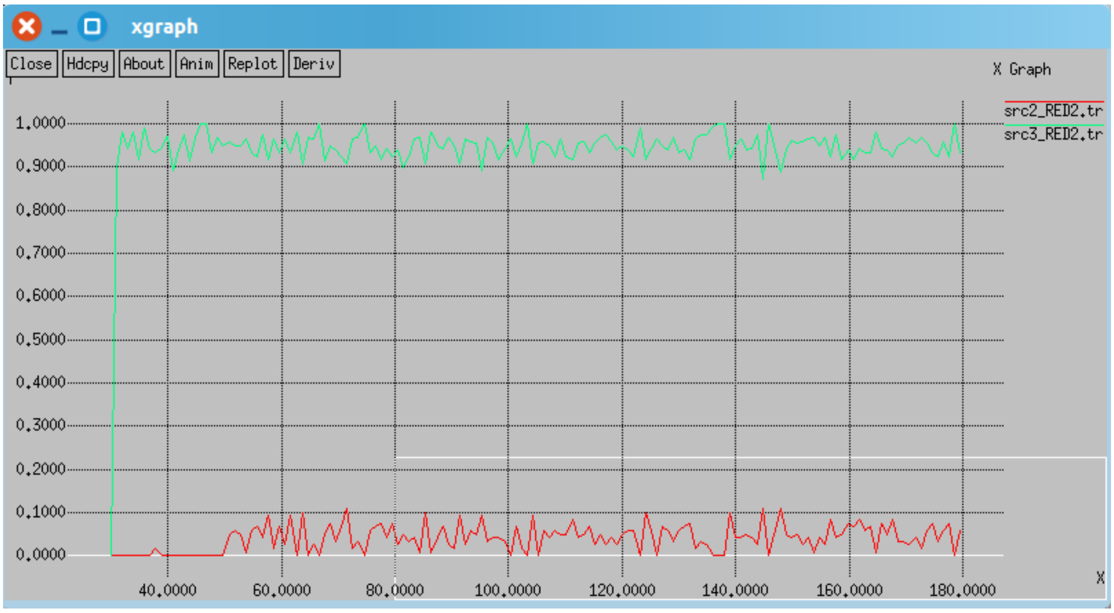
1) DROPTAIL buffer management: Instantaneous throughput





2) RED buffer management: Instantaneous throughput





3. Final analysis and conclusion:

DropTail is a passive Queue Management algorithm which only sets a maximum length for each queue at router. Routers decide when to drop packets. It uses the FIFO algorithm and the traffic is treated equally, which has the same priority. When queue buffer is filled to its maximum capacity, the queue will not receive any packets arrived afterword. That is, DropTail will keep discarding the packet until the queue has enough room for the new packets.

However, in RED, Dropping is based on the threshold values; min threshold and max threshold. RED monitors the average queue size, and checks whether it lies between some minimum threshold and maximum threshold. If it does, then the arriving packets will dropped with probability p=p(avg) which is an monotonous function with average queue size. If aveg

exceeds the maximum threshold, all the packets arrived will be discarded.
For the Scenario1, DROPTAIL, the router cannot distinguish the two incoming links, and drop the packet when the queue is full. As long as the packets are dropped, the TCP agent will halve the congestion window, and in this way, two links will finally get the average throughputs near 500 Kbps.

As for Scenario1 RED, the dropping probability of RED is related with the size of queue. If we assume the count is the number of packets which are arriving consecutively and not discarded since the last discard packets. When the count increases, then the dropping probability will also increase. So when one source node H1 transmits for a large number of packets, then it has a high probability that the queue will discard its packet. Thus H1 will halve its congestion windows and the transmission rate will go down, which in return will increase the transmission rate of H2. When the H2's packets go up, the queue management will decrease its rate. In this case, the instantaneous throughputs of RED will goes up and down more drastic than the DropTail.

As for senario2 DropTail, we introduce an addition UDP link. Because UDP never react to the packet loss as TCP, thus whenever there is packet loss, UDP just sends the packets as before. However TCP agents will halve their congestion windows. Since the send rate of UDP is equal to the bandwidth of link (R1, R2), so packets loss would always happen as long as TCP agents send packets. As a result, TCP would decrease their congestion window size to 0 and all bandwidth would be occupied by UDP, which explains why the throughput of src1 and src2 are 0, and scr3 is 1Mbps.

As Scenario2 RED, although TCP agents will decrease their window size and transmit rate, the RED would drop the packets of UPD with a higher probability. As a result, the TCP agent would be able to send more packets than the previous scenario, but UDP still occupied the huge part of the bandwidth. The final result showed the UDP occupies most of the bandwidth and TCP occupies a small part.