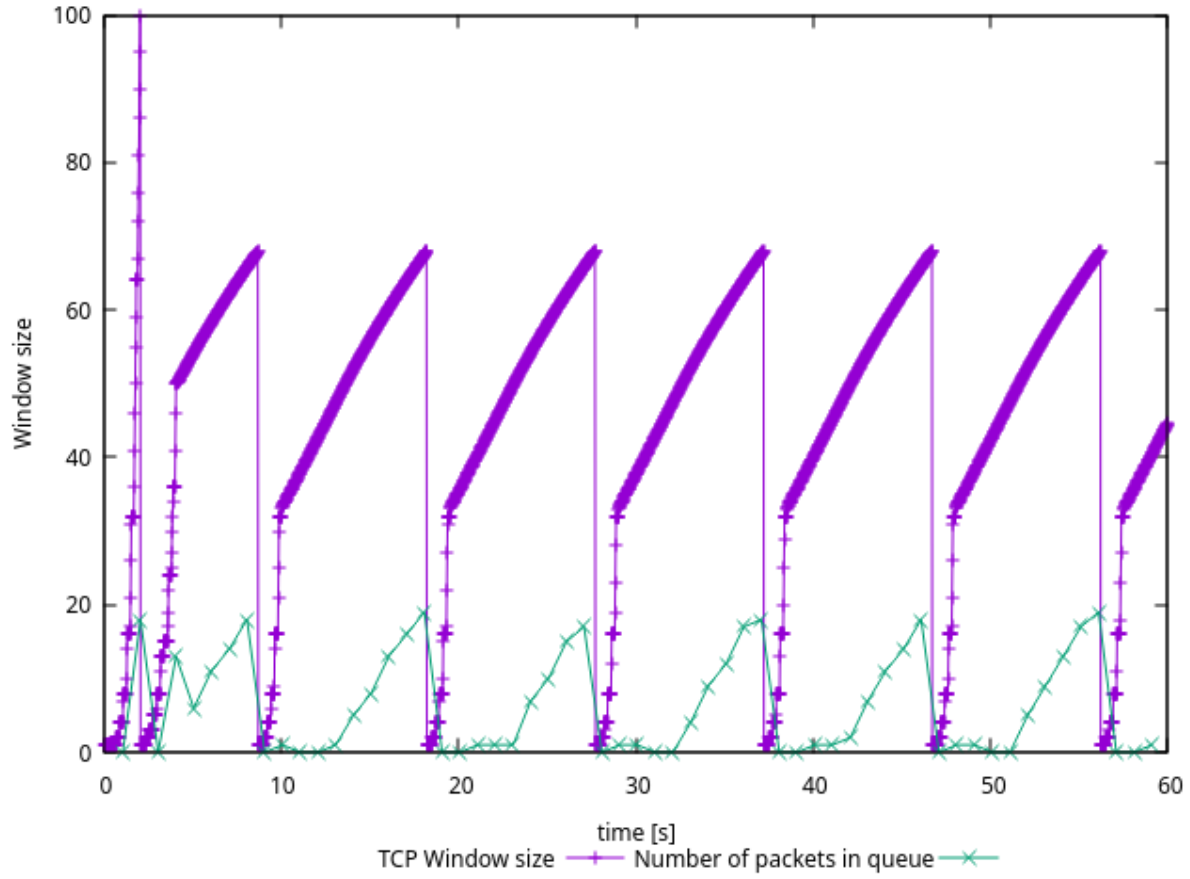
Exercise 1: Understanding TCP Congestion Control using ns-2 (4 Marks)

Question1:



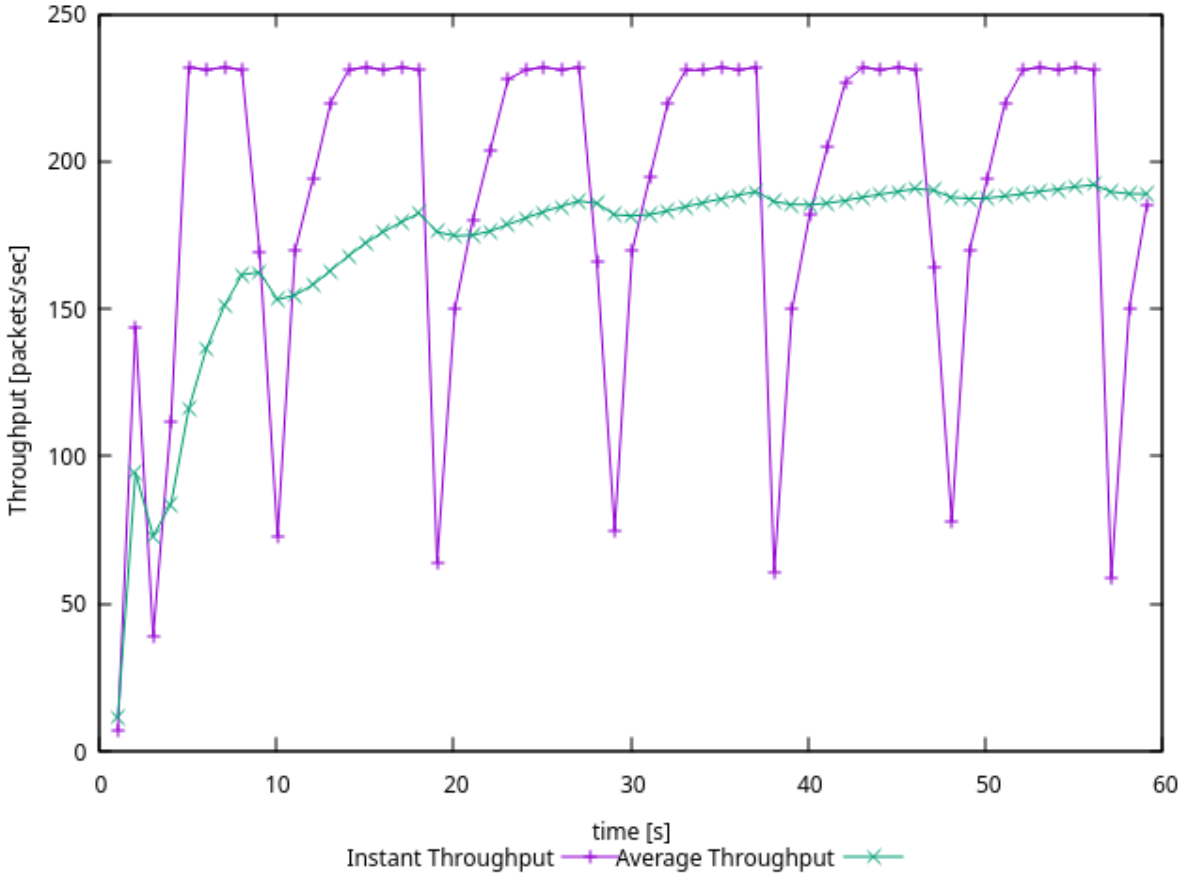
(a) In this case, what is the maximum size of the congestion window that the TCP flow reaches?

We have already set the maximum initial window size to be 150MSS. We can observe from the graph that the maximum congestion window is 100MSS

(b) What does the TCP flow do when the congestion window reaches this value? Why?
When it reaches 100 MSS, it encounters loss, either triple dup ACK or timeout, it halved the thread of its failing size to 50MSS and set the cwnd equal to 1 where TCP-Tahoe will set cwnd equal to 1 on triple dup ACK and timeout.

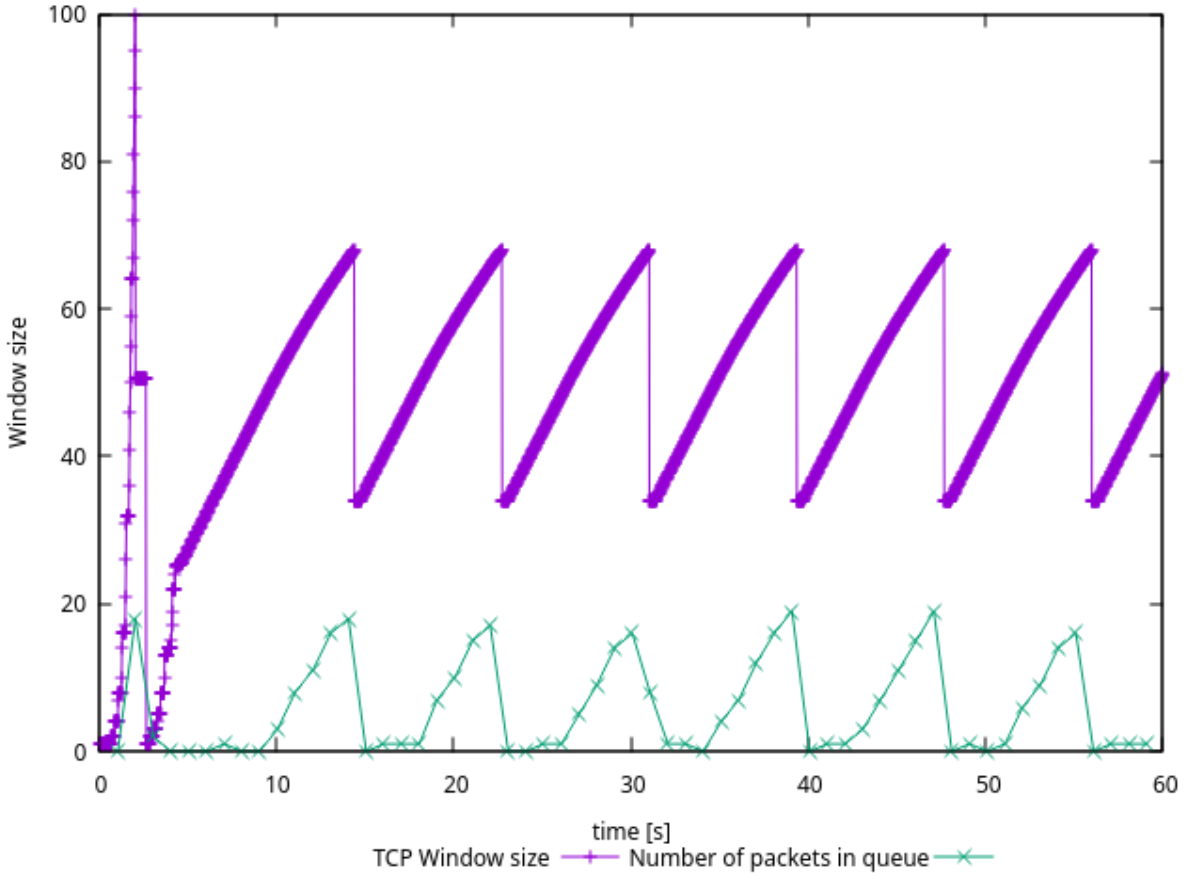
(c) What happens next?
It will start with cwnd = 1 with a slow start and when it reaches threshold, it will change to additive increase state until loss occurs again.

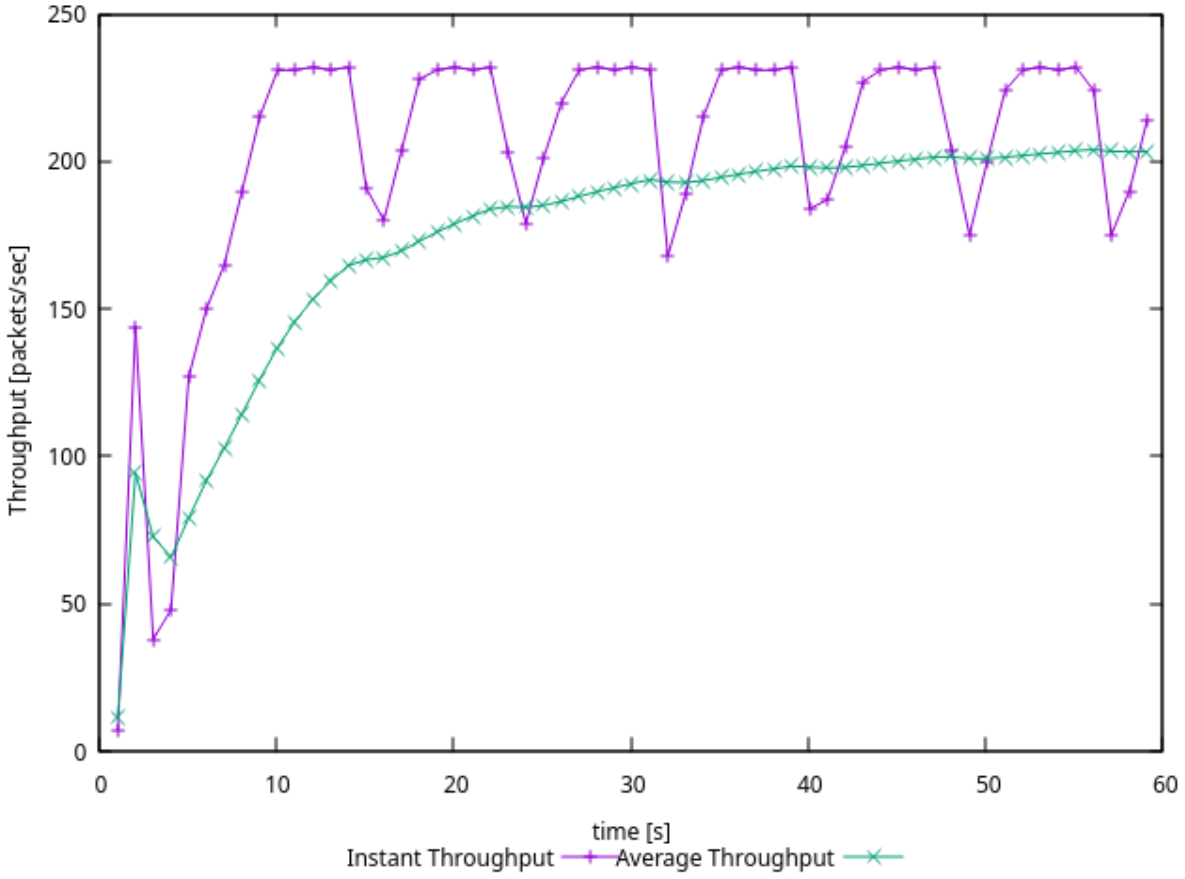
Question 2: From the simulation script we used, we know that the packet's payload is 500 Bytes. Keep in mind that the size of the IP and TCP headers is 20 Bytes each. Neglect any other headers. What is the average throughput of TCP in this case? (both in number of packets per second and bps)



The average throughput is around 190 packet per second. The size of packets is 500+20+20 = 540 bytes, therefore the average throughput is 540*190*8 = 821kbps

Question 3: Repeat the steps outlined in Questions 1 and 2 but for TCP Reno. Compare the graphs for the two implementations and explain the differences. (Hint: compare the number of times the congestion window returns to zero in each case). How does the average throughput differ in both implementations?

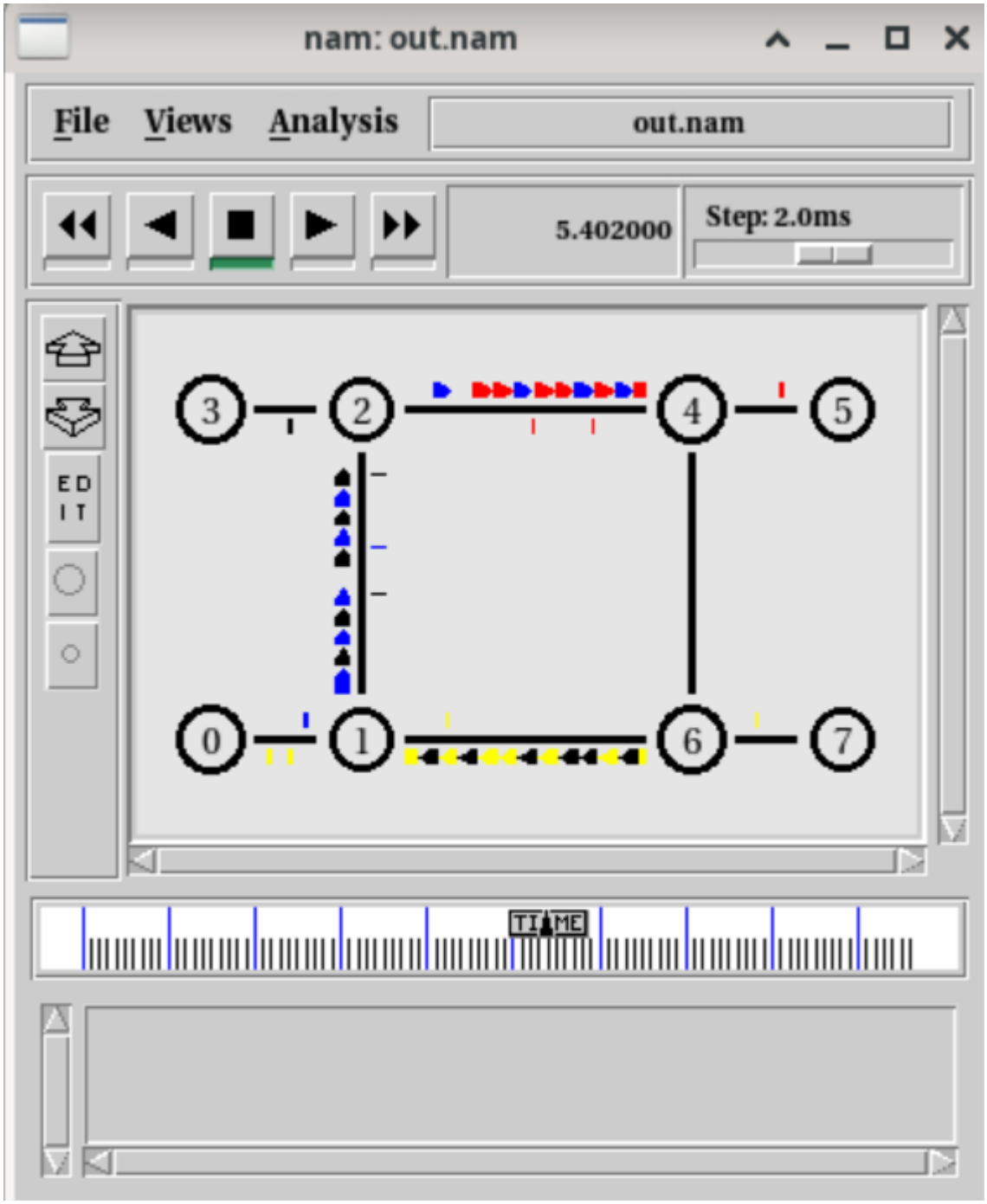

Window size
time [s]
TCP Window size
Number of packets in queue

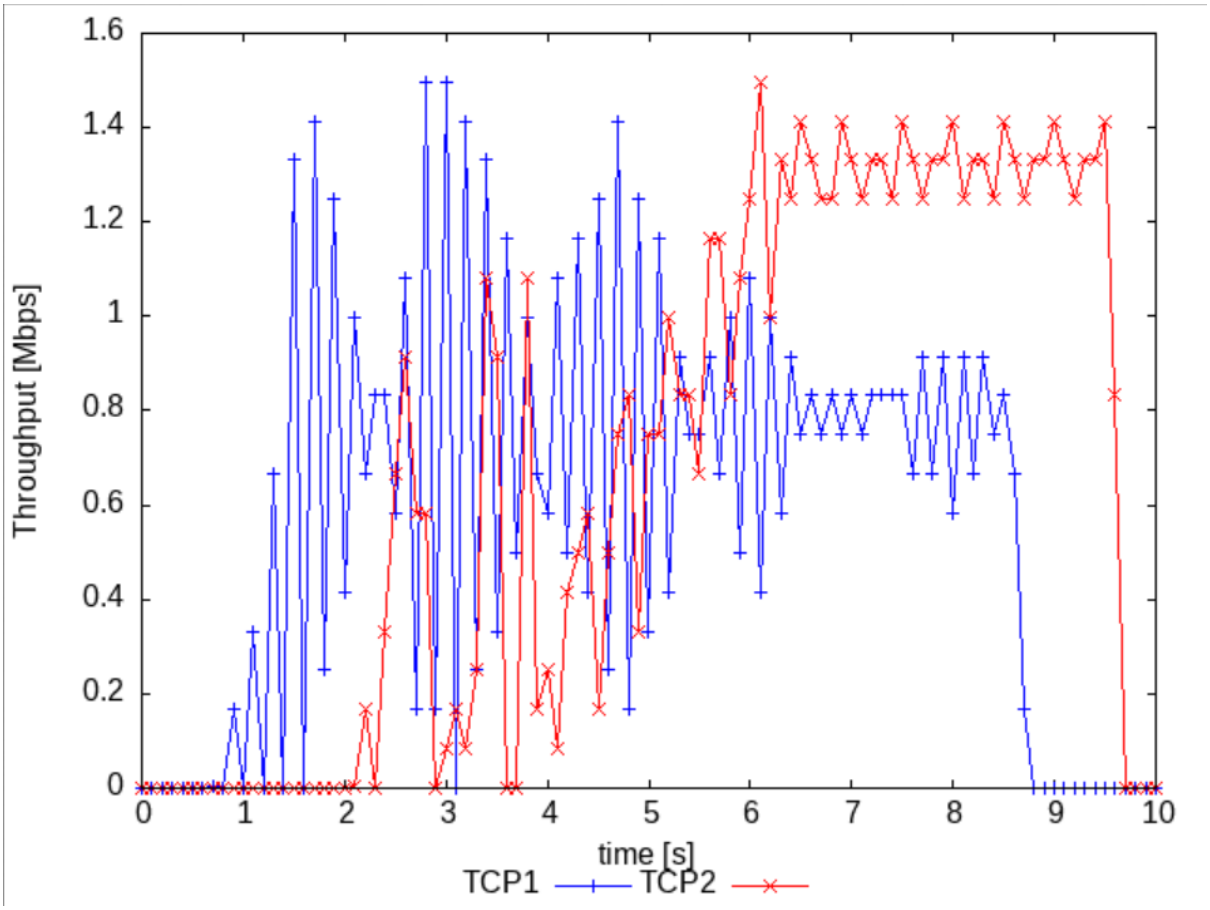

Throughput [packets/sec]
time [s]
Instant Throughput
Average Throughput

TCP-Tahoe will set cwnd = 1 when it meets either triple dup ACK or time out, TCP -Reno will only set cwnd = 1 when it meets time out, where it will half cwnd when it meets triple dup ACK. Therefore, the number of times the congestion window returns to zero in TCP-Tahoe will be more than the number of times the congestion window returns to zero in TCP-Reno.

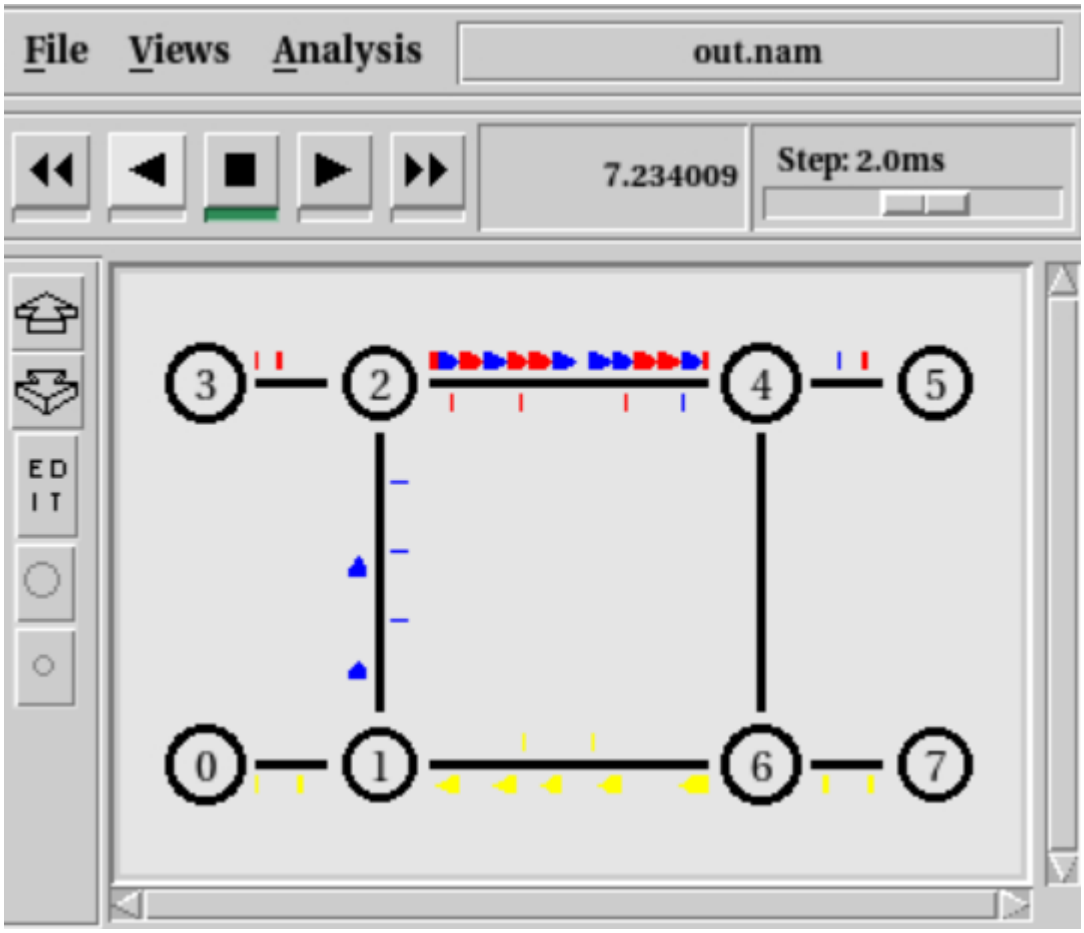
Since the number of times the congestion window returns to zero in TCP-Reno is less than in TCP-Tahoe, the average throughput of TCP-Reno is higher than TCP-Tahoe which is around 200.

Exercise 2: Setting up NS2 simulation for measuring TCP throughput (3.5 marks)


nam: out.nam
File
Views
Analysis
out.nam
5.402000
Step: 2.0ms
EDIT
3
2
4
5
0
1
6
7
TIME



Question 1: Why is the throughput achieved by flow tcp2 higher than tcp1 between 6 sec to 8 sec?



During 6s to 8s, there is no congestion in path 1->2 and path 2->4. The cwnd size is already in a consistent state(addictive increase).

For TCP2, it travels from 3->2 (high speed) 2->4(slow speed) 4->5(high speed)

For TCP1, it travels from 0->1 (high speed) 1->2(slow speed) 2->4(slow speed) 4-5(high speed).

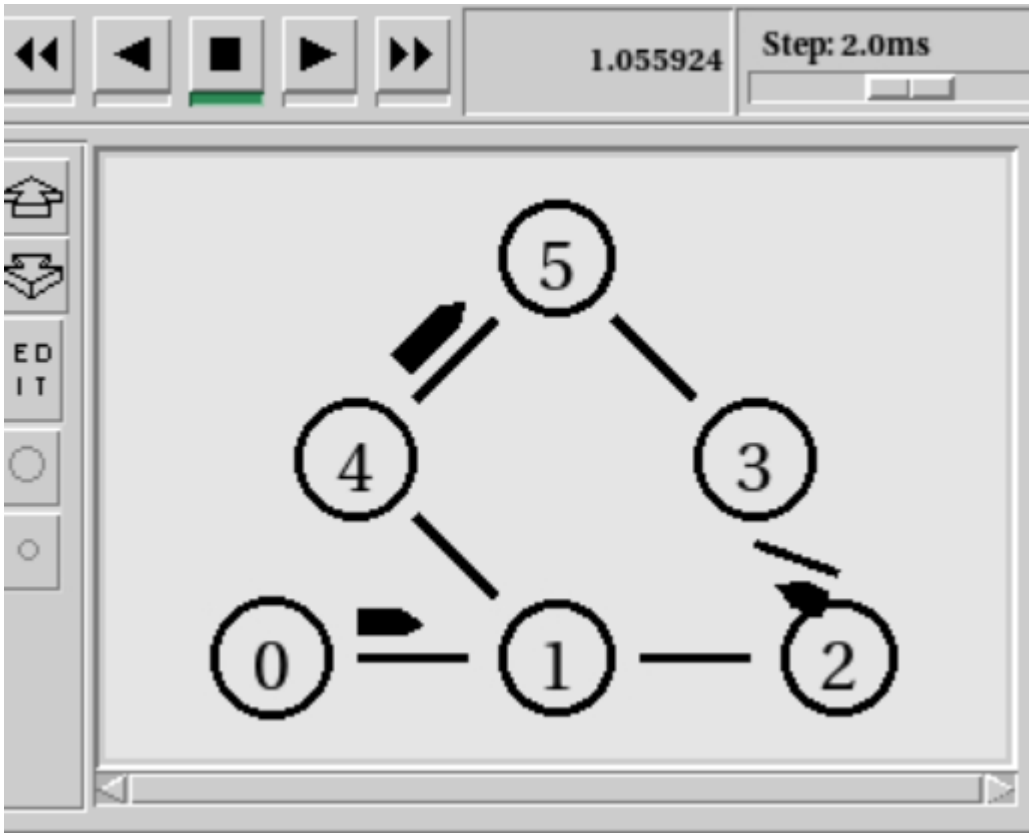
It took longer for TCP 1 to finish transmission, TCP1 and TCP2 use one common path 2->4, TCP2 will arrive node 2 faster as 3->2 is a fast transmit but 0->1->2 is slower, caused a smaller throughput for TCP1 under a consistent transmission environment.

Question 2: Why does the throughput for flow tcp1 fluctuate between a time span of 0.5 sec to 2 sec?

It was in slow start state, the cwnd changed exponentially, trying to reach an optimised cwnd.

Exercise 3: Understanding the Impact of Network Dynamics on Routing (2.5 marks)

Question 1: Which nodes communicate with which other nodes? Which route do the packets follow? Does it change over time?



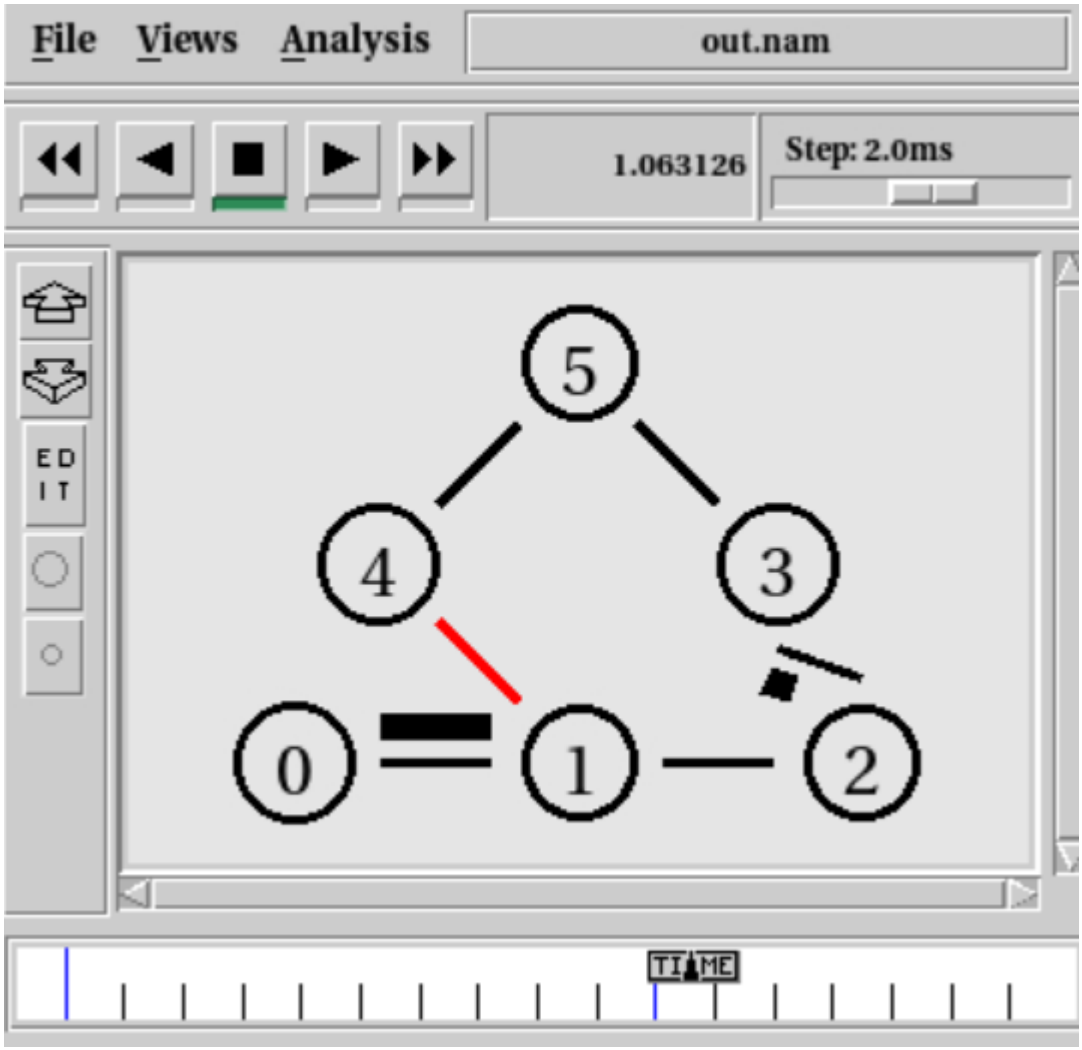
Node 0 communicate with node 1, node 1 communicate with node 4, node 4 communicate with node 5. Node 2 communicate with node 3, node 3 communicate with node 5.

0->1->4->5

2->3->5

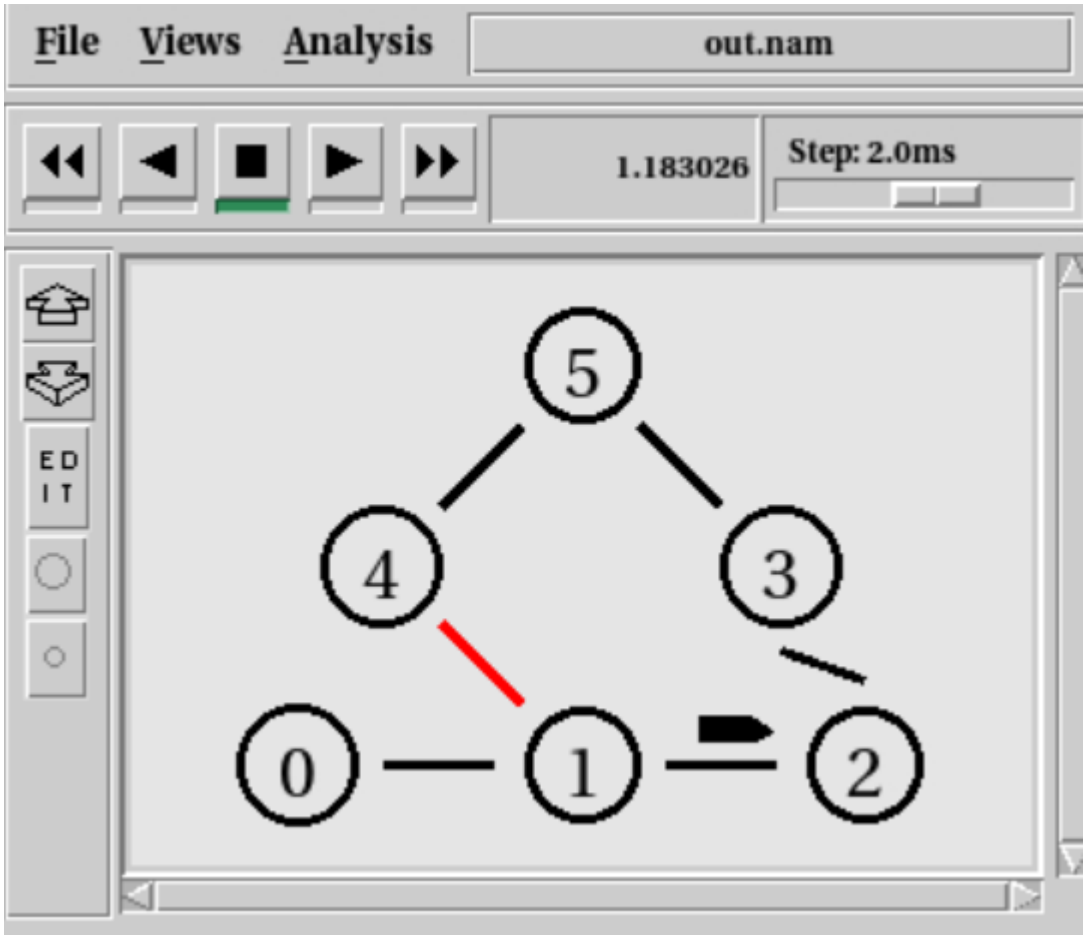
They don’t change over time

Question 2: What happens at time 1.0 and time 1.2? Does the route between the communicating nodes change as a result?

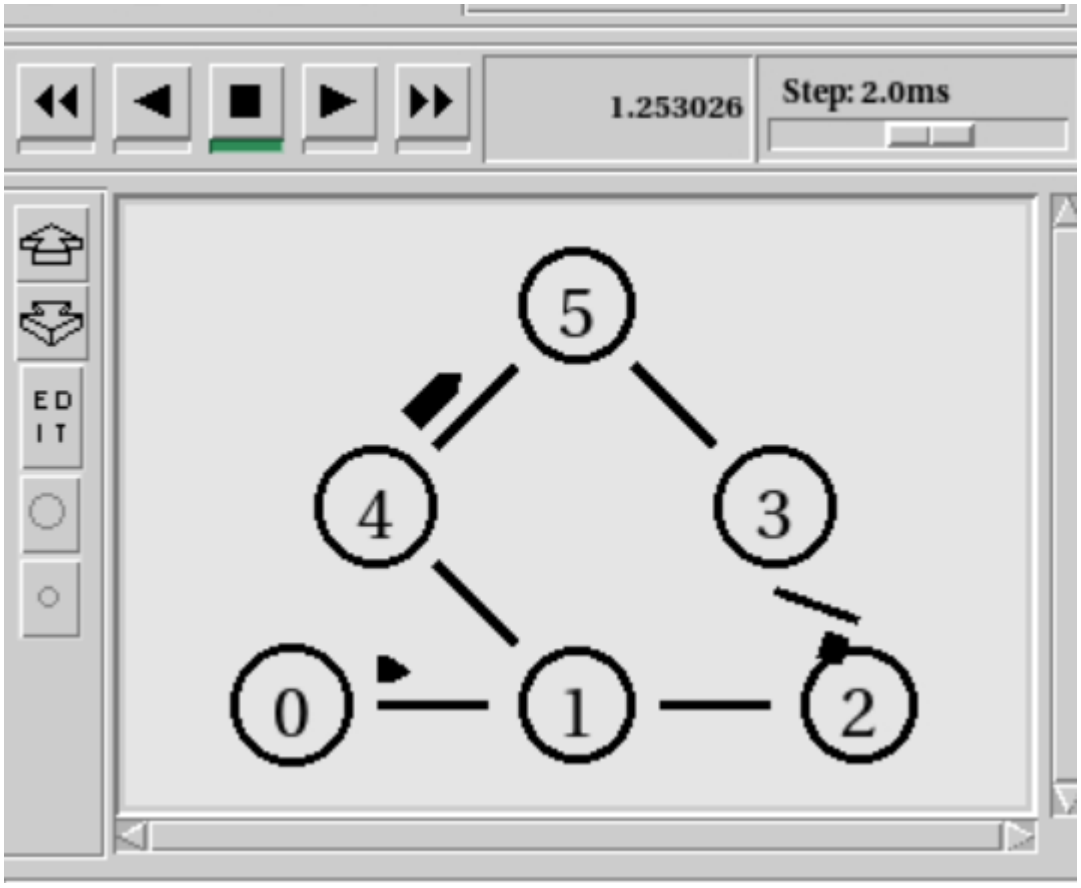


At time 1.0, 1->4 is down, at time 1.2 it restores connection. The route is changed as all the data sent from node 0 lost.

Question 3: Did you observe additional traffic compared to Step 3 above? How does the network react to the changes that take place at time 1.0 and time 1.2 now?

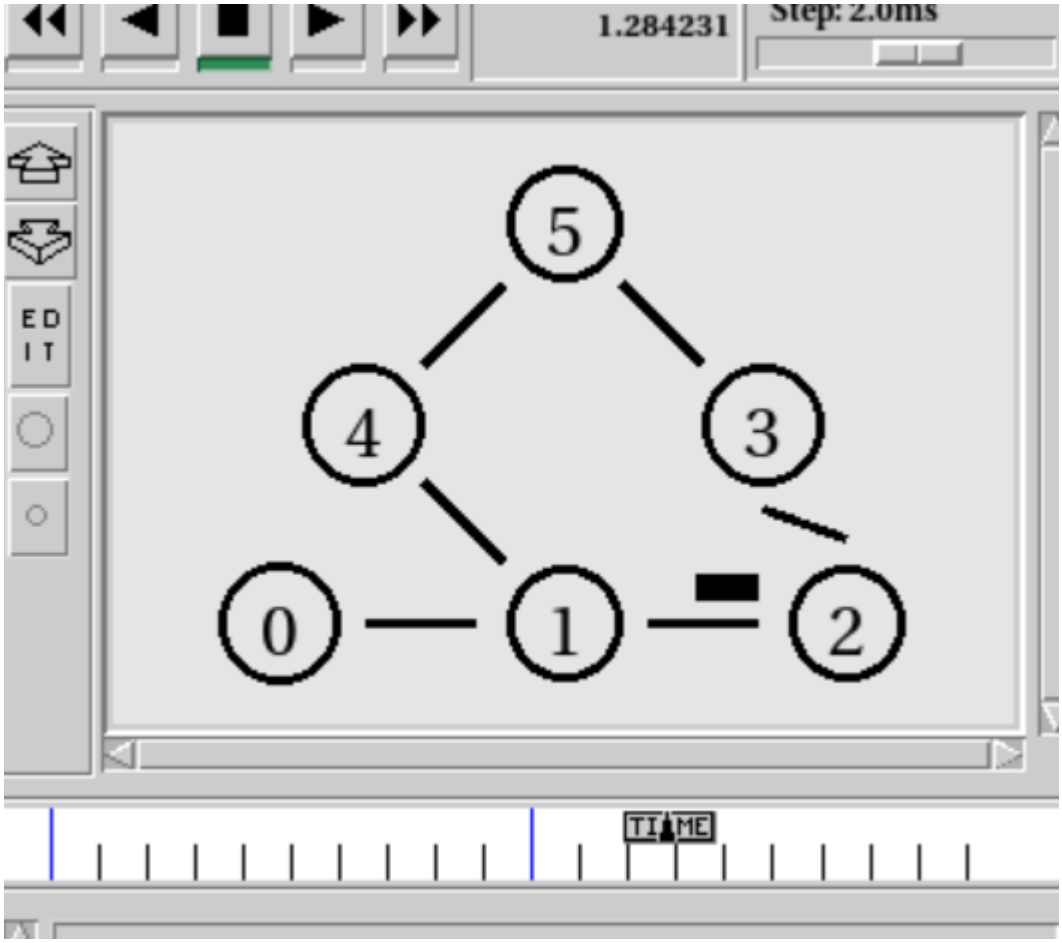


Yes, now after 1->4 is down, the new protocol leads the connection to a new route 0->1->2->3->5, and the data is not lost.



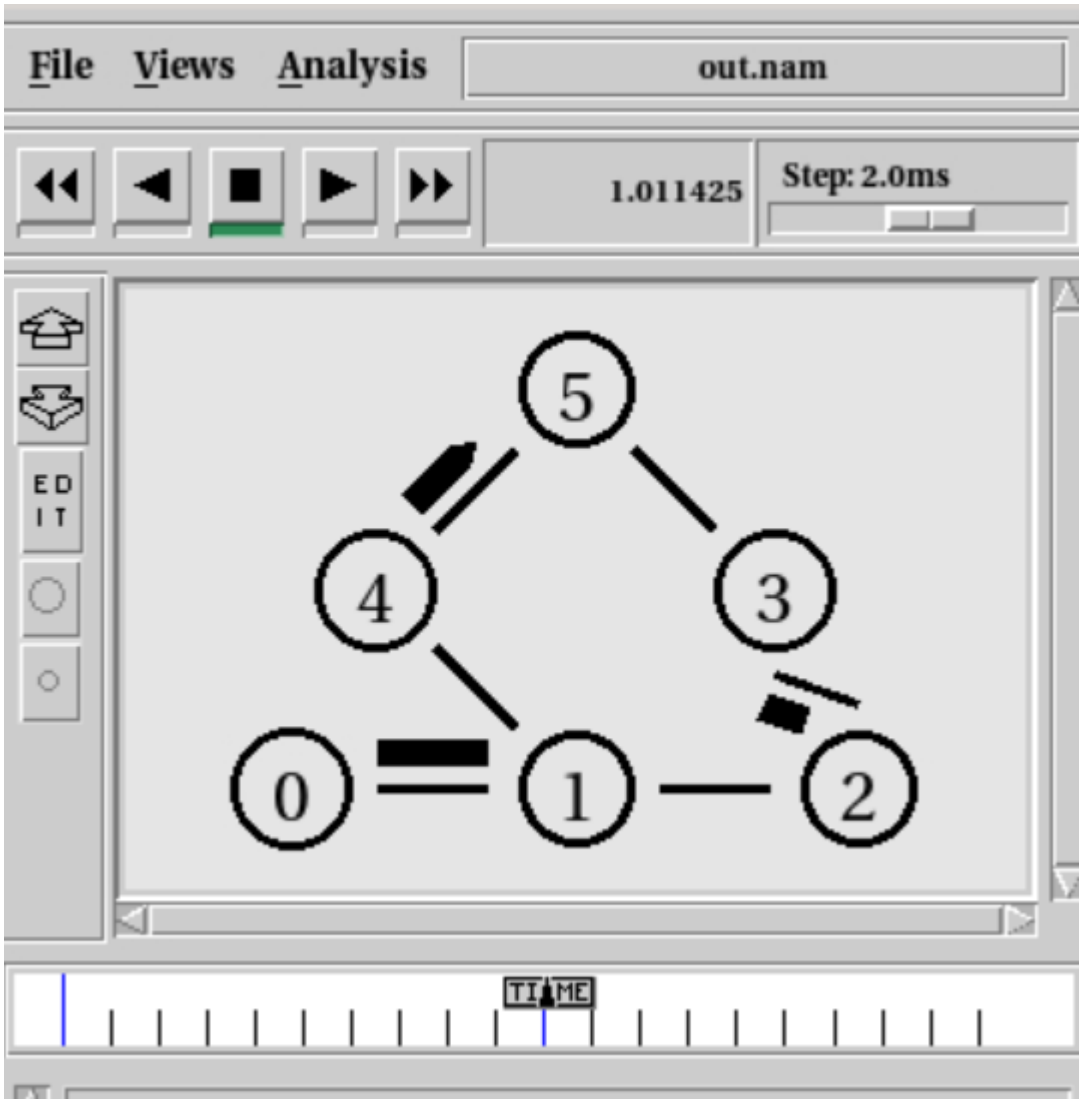
And at time 1.2, the link restores, data from node 0 go back to the original route again.

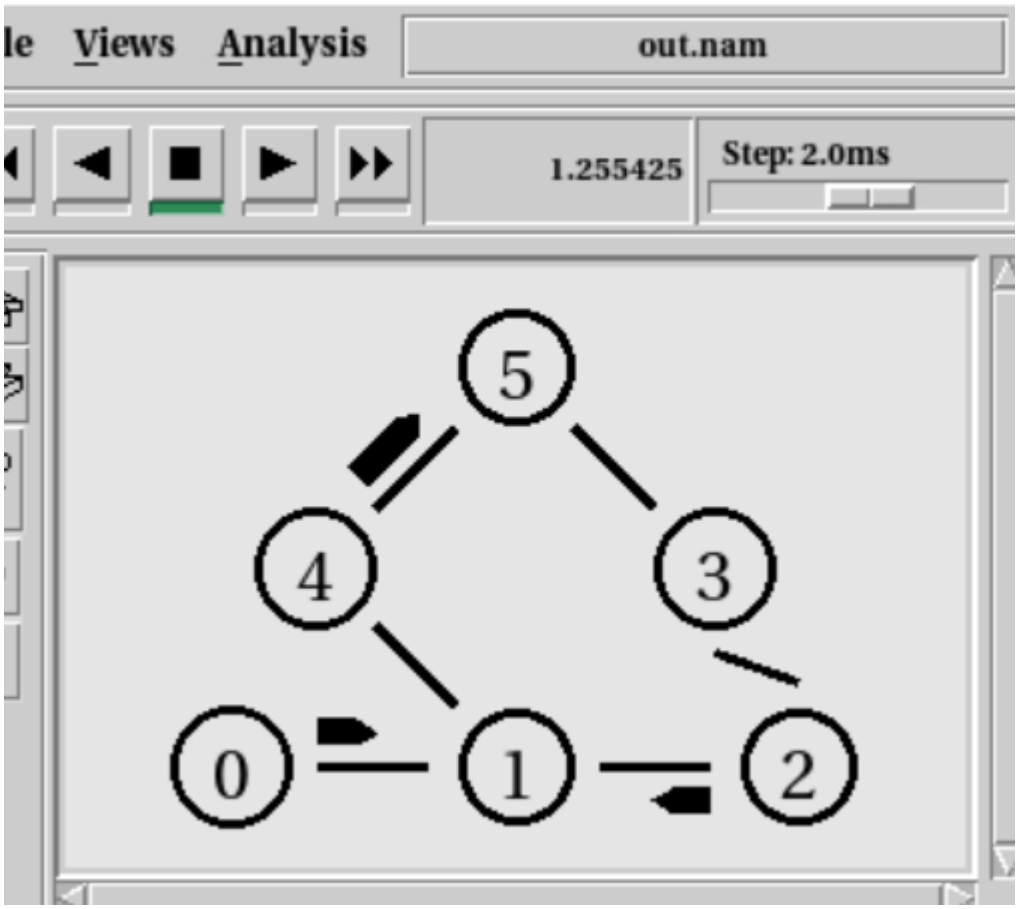
Question 4: How does this change affect the routing? Explain why.



The default cost of a link is 1, now we set the cost of 1->4 to be 3 now, the protocol will lead us to the cheapest cost route, which is 0->1->2->3->5 now.

Question 5: Describe what happens and deduce the effect of the line you just uncommented.


File
Views
Analysis
out.nam
1.011425
Step: 2.0ms
ED
IT
5
4
3
0
1
2
TIME



The node 2 is now transmitting the data using two routes 2->1->4->5 and 2->3->5 where they have the same cost, 1+2+1 = 4 and 1+3 = 4.

```
Node set multiPath_ 1
Enable the multipath with same cost.
```