Report: LAB5.2

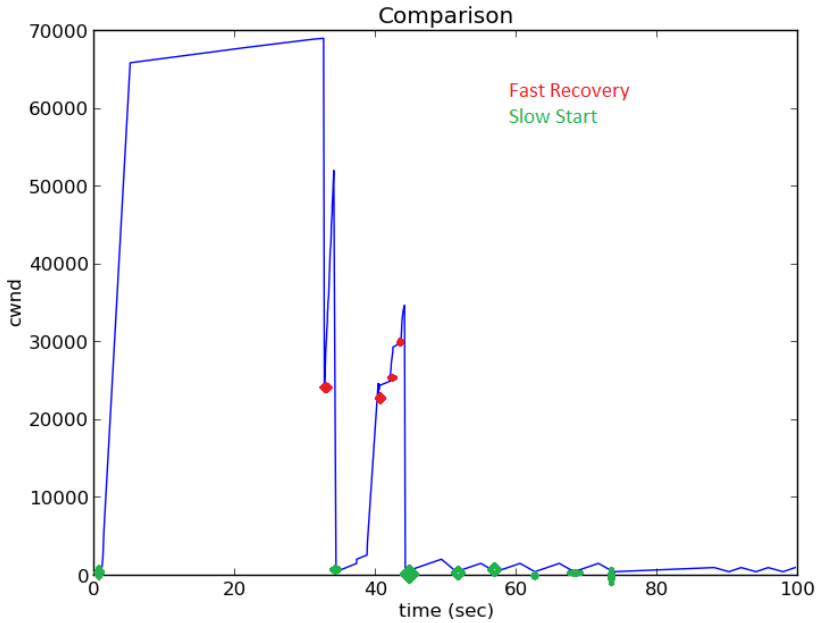
Abhishek Yadav, CS12B032

March 20, 2015

Question: 1

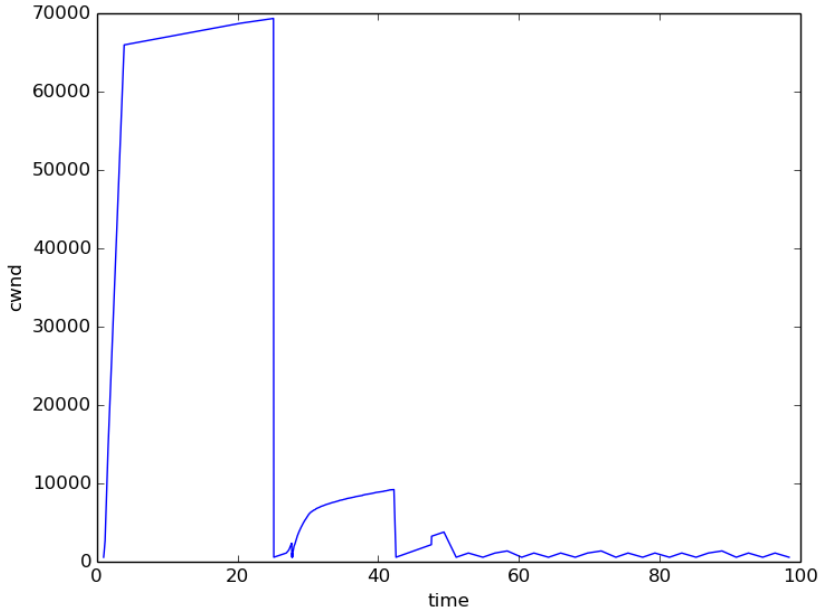
1. after finishing the initial setup for the dumbbell topology of the network, **TCP** application is started at 1 sec and **UDP** application at 20 sec at the rate of 250 kbps so as clog the half of the link capacity.

   At 30 sec changed the rate of the UDP application to 500 kbps so as to clog the whole of the dumbbell bridge's capacity.
   corresponding plot of the changes in the congestion window size vs time:-

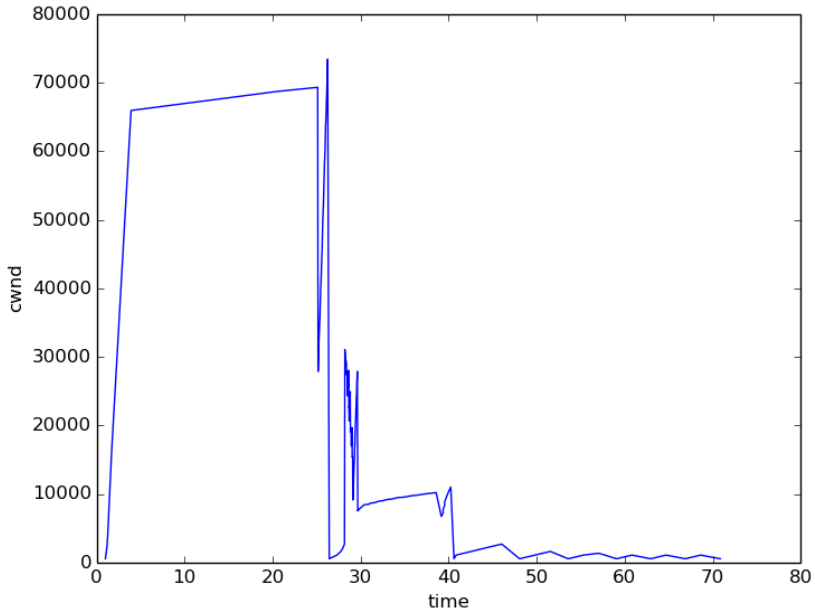


After changing the default TCP as TcpTahoe, TcpNewReno and TcpReno corresponding plots showing the variations of cwnd with time are:-

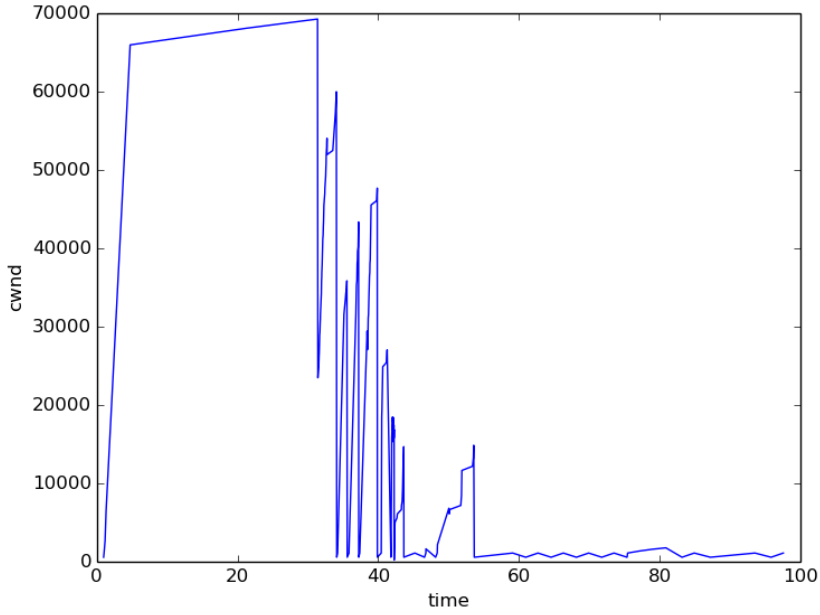
TcpTahoe:



TcpReno:


80000
70000
60000
50000
40000
30000
20000
10000
0
cwnd
0
10
20
30
40
50
60
70
80
time

TcpNewReno


70000
60000
50000
40000
30000
20000
10000
0
cwnd
0
20
40
60
80
100
time

2. For this part of the experiment, modified the previous setup with the following changes:

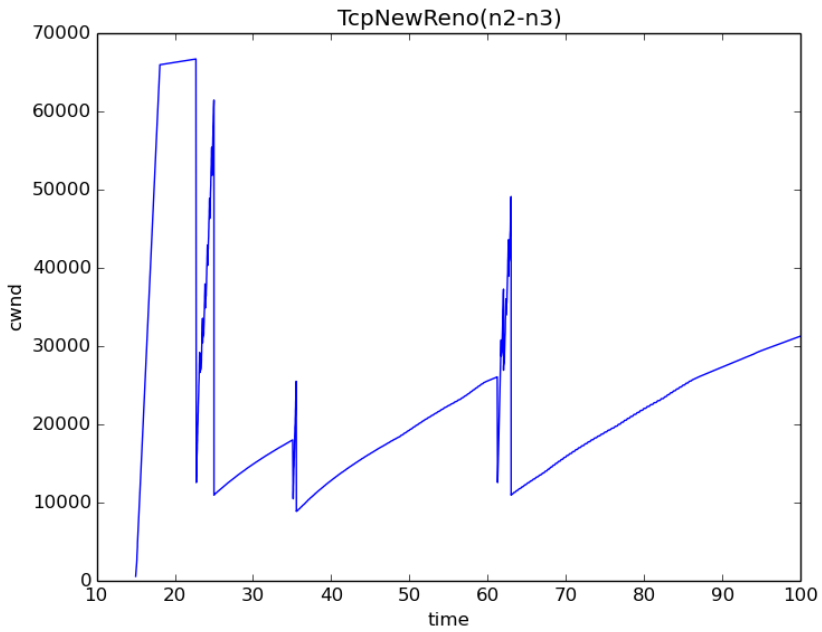
   (a) Installed a **TcpReno** socket instance on node1 that connects to node3.

   (b) Installed a **TcpNewReno** socket instance on node2 that connects to node3.

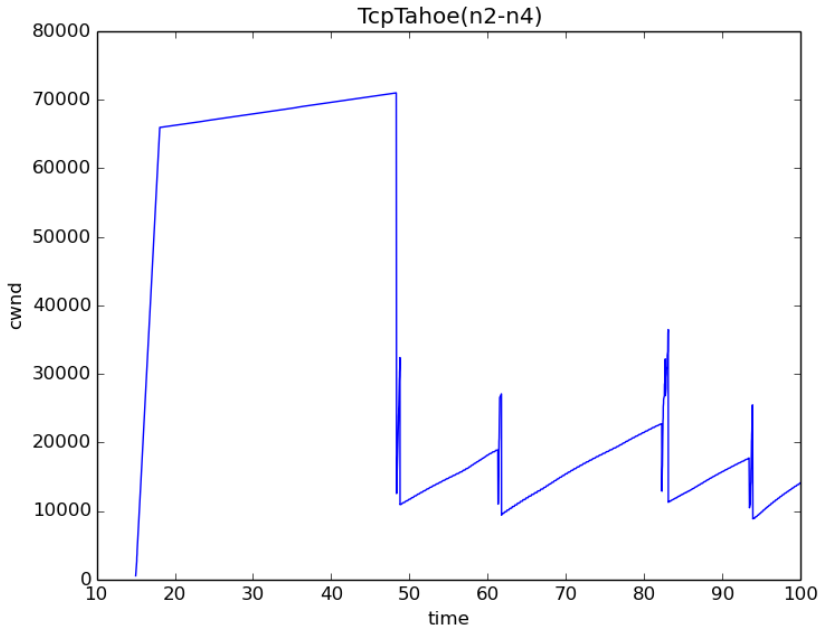
   (c) Installed a **TcpTahoe** socket instance on node2 that connects to node4

Experiment: Started node1-node3 flow at time 1 sec, and that of node2-node3 and node2-node4 at time 15 sec. corresponding plots are listed below

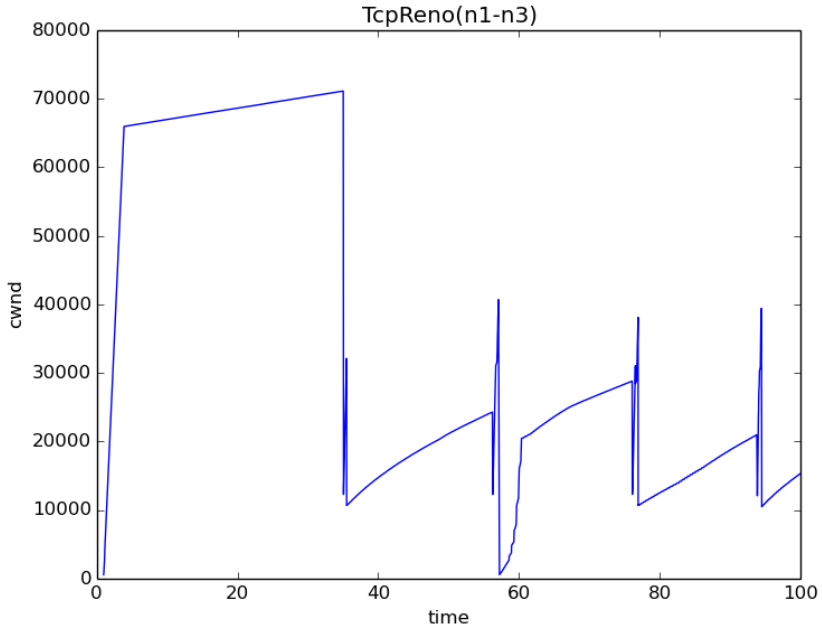
TcpNewReno



TcpTahoe



TcpReno



Observation: In case of **n2-n3** and **n2-n4** flows there is no cwnd

since these flows start at 15 sec. In all three scenarios the congestion window threshold is observed to be around 70000. And finally (n2-n3) link shows fast recovery after hitting the threshold.

Question: 2

TcpLab the changes made in the code of TcpLab and observed cwnd vs time plot is listed below.

- $m\_cWnd = 2 * m\_ssThresh$
- $m\_cWnd* = m\_segmentSize$
- $m\_cWnd = 10 * m_cwnd + m\_segmentSize$