Mohd. Aftab Alam
02013302717

EXPERIMENT 6

Aim :

Simulate Telnet and FTP between n Sources – n Sinks (n = 1, 2, 3). Evaluate the effect of increasing Data Rate on Congestion.

TCP Tahoe :

- This is the original version of TCP Congestion Control as implemented by Jacobson
- Congestion Detection Mechanism is based on Packet Loss and End to End Delay
- Techniques used for Congestion Control :
  - ***Slow Start***
  - ***Congestion Avoidance***
  - ***Fast Retransmit***
  - ***Fast Recovery***
- **TCP Reno –** This is the most popular version of TCP congestion control mechanism today
- **TCP Vegas –** This is completely new implementation

To accommodate the uncertainty, TCP operates in two different modes / phases :

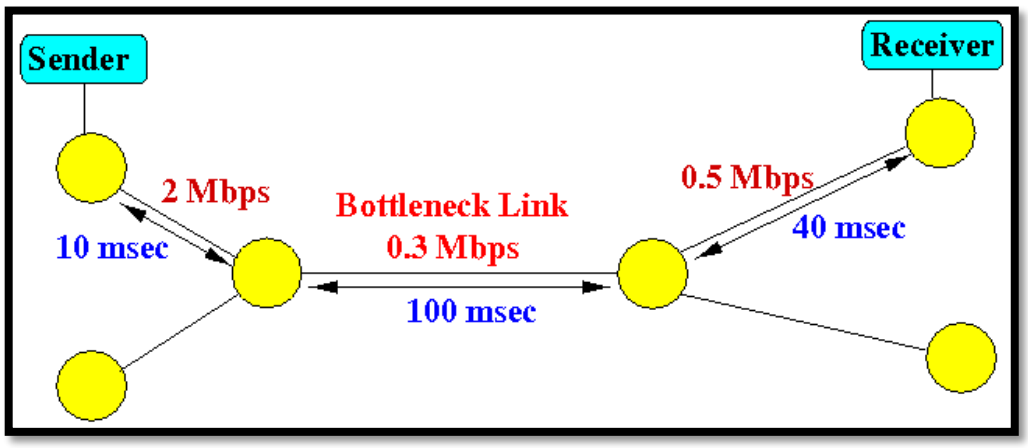
1. Slow Start Mode / Phase :

- This is the startup mode of operation of TCP
- In this mode / phase, TCP has an idea (guess) about the maximum transmission rate and TCP is trying to reach this transmission rate
- Although TCP has an idea (guess) about this maximum transmission rate, TCP will NOT transmit at this rate instantaneously
- Rather, TCP will try to reach this maximum transmission rate in a piece meal fashion
- In this phase, TCP will start by transmitting ONE packet and at each successfully transmission epoch, TCP will DOUBLE the number of packets (resulting in an exponential increase in number of packets in time).

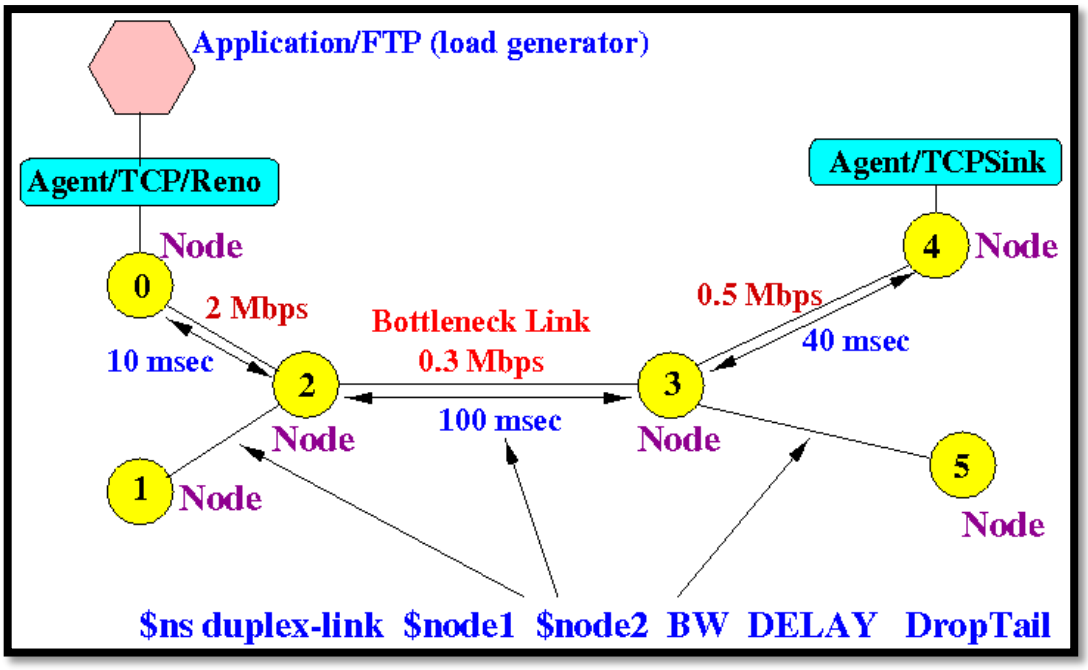
2. Congestion Avoidance Mode / Phase :

- This is the phase that begins AFTER the startup phase. The startup phase ends when TCP has reached the maximum transmission rate that it "believes" to be safe
- Because TCP has reached the maximum safe level, it would appear that there is still some more capacity available – it would be a shame NOT to use the available capacity
- But TCP has no idea what the new maximum capacity is so it must be careful
- In this phase, TCP will increase the number of packets much slower than in the startup phase (increase rate will not be exponential, but linear)

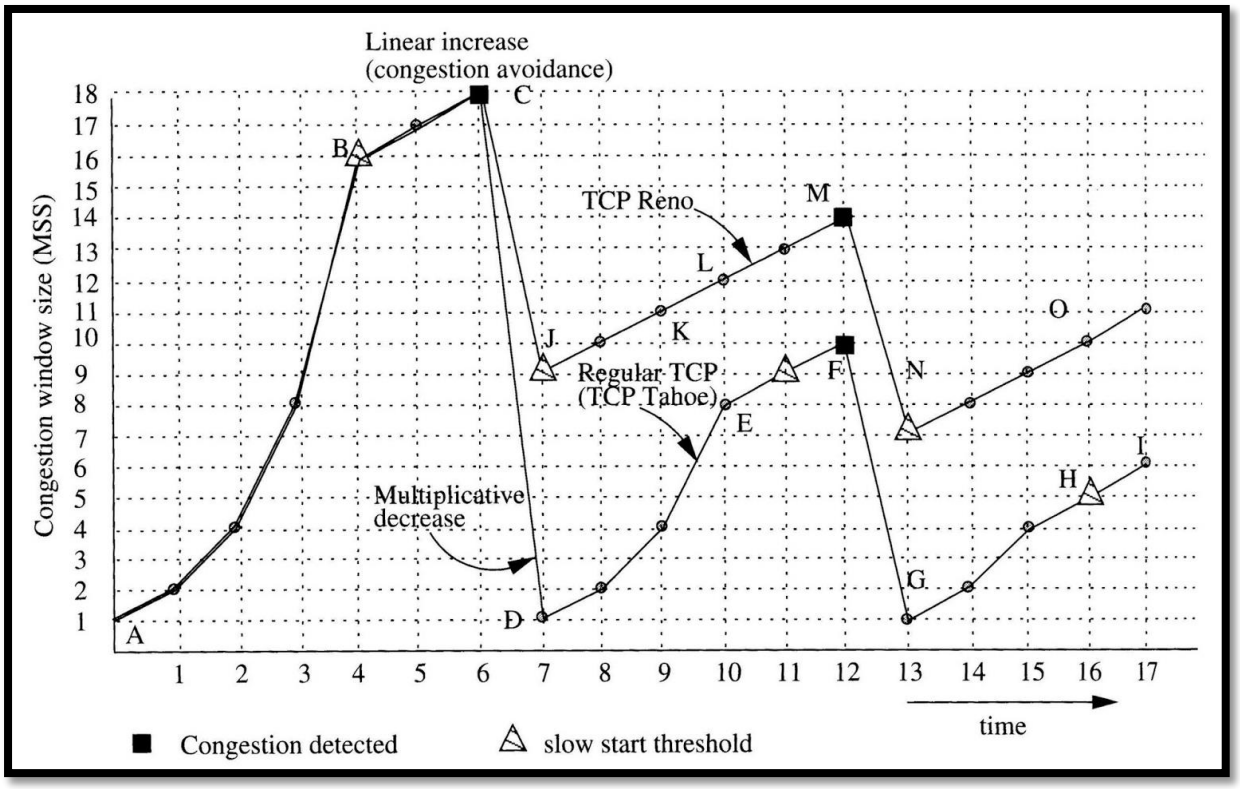
TCP Tahoe Demo – We will look at the operation of TCP Tahoe in this sample network :



Mohd. Aftab Alam
02013302717



Breakdown of the NS Components that Makeup the above Network



Congestion Window Plot

Mohd. Aftab Alam
02013302717

Code :

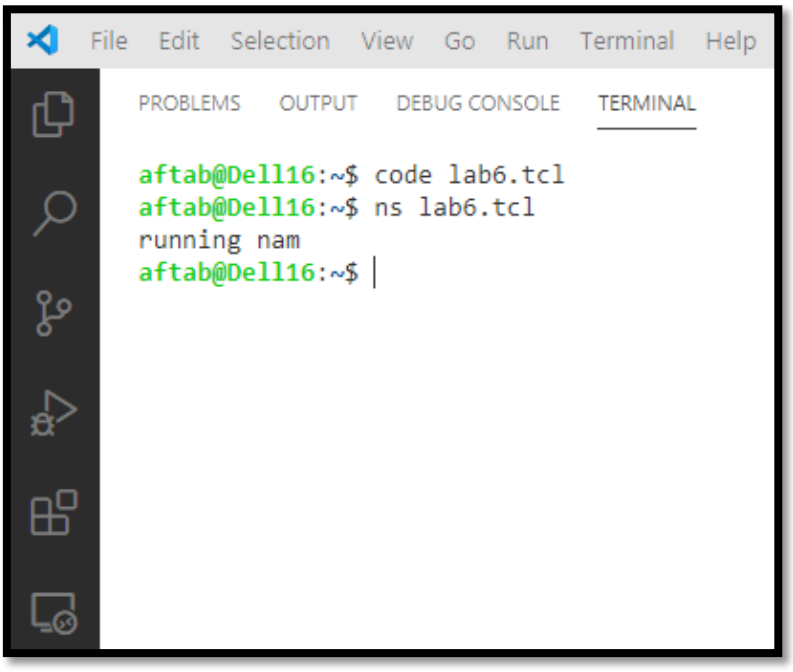
```
set ns [new Simulator]
$ns color 1 Blue
$ns color 2 Red
set file2 [open out.nam w]
$ns namtrace-all $file2
#Open the window trace file
set winfile [open WinFile w]
proc finish {} {
global ns file2
$ns flush-trace
close $file2
exec nam out.nam &
exit 0 }
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
set n4 [$ns node]
set n5 [$ns node]
$ns duplex-link $n1 $n2 2Mb 10ms DropTail
$ns duplex-link $n0 $n2 2Mb 10ms DropTail
$ns simplex-link $n2 $n3 0.3Mb 200ms DropTail
$ns simplex-link $n3 $n2 0.3Mb 200ms DropTail
$ns duplex-link $n3 $n4 0.5Mb 40ms DropTail
$ns duplex-link $n3 $n5 0.5Mb 30ms DropTail
#Monitor the queue for link (n0-n1) (for nam)
$ns duplex-link-op $n2 $n3 queuePos 0.1
#Give node position (for nam)
$ns duplex-link-op  $n0 $n2 orient right-down
$ns duplex-link-op  $n1 $n2 orient right-up
$ns simplex-link-op $n2 $n3 orient right
$ns simplex-link-op $n3 $n2 orient left
$ns duplex-link-op  $n3 $n4 orient right-up
$ns duplex-link-op  $n3 $n5 orient right-down
```

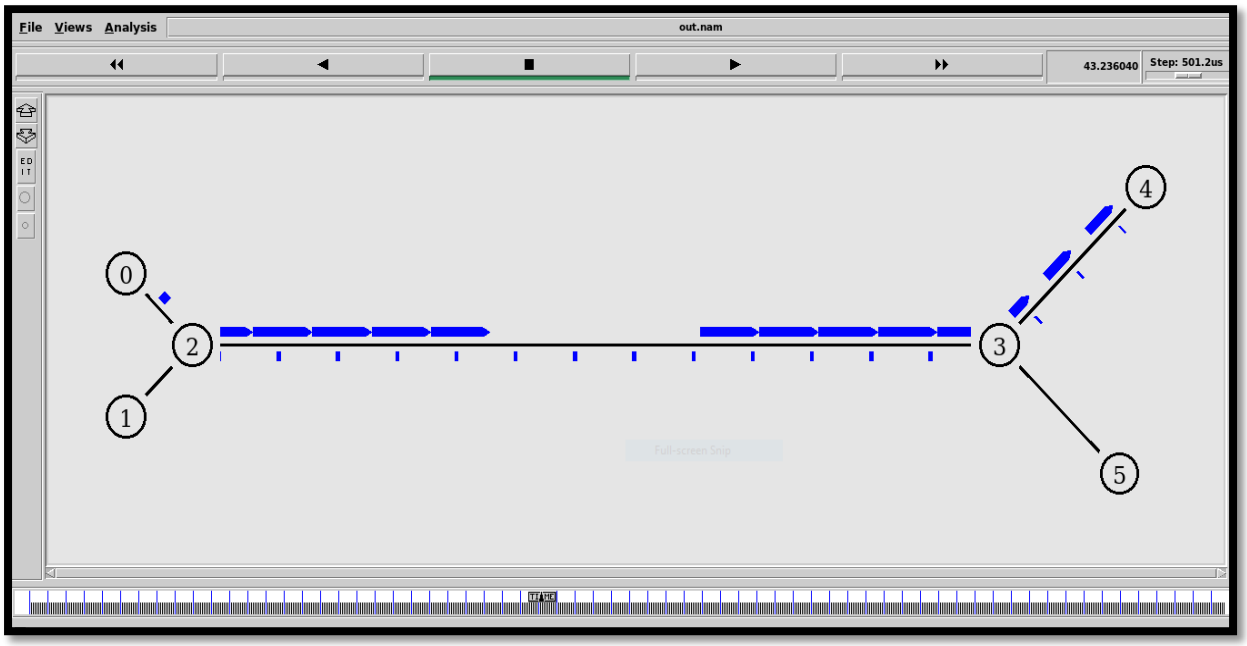
Mohd. Aftab Alam
02013302717

```
#Set Queue Size of link (n2-n3) to 10
$ns queue-limit $n2 $n3 10
#Setup a TCP connection
set tcp [new Agent/TCP]
$ns attach-agent $n0 $tcp
#Set sink [new Agent/TCPSink/DelAck]
set sink [new Agent/TCPSink]
$ns attach-agent $n4 $sink
$ns connect $tcp $sink
$tcp set fid_ 1
$tcp set window_ 8000
$tcp set packetSize_ 552
#Setup a FTP over TCP connection
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ftp set type_ FTP
#Schedule start/stop times
$ns at 0.1 "$ftp start"
$ns at 100.0 "$ftp stop"
#plotWindow (tcpSource,file) : write CWND from $tcpSource
#to output file $file every 0.1 sec
proc plotWindow {tcpSource file} {
global ns
set time 0.1
set now [$ns now]
set cwnd [$tcpSource set cwnd_]
set wnd [$tcpSource set window_]
puts $file "$now $cwnd"
$ns at [expr $now+$time] "plotWindow $tcpSource $file" }
#Start plotWindow
$ns at 0.1 "plotWindow $tcp $winfile"
#Set simulation end time
$ns at 125.0 "finish"
puts "running nam"
$ns run
```

Mohd. Aftab Alam
02013302717

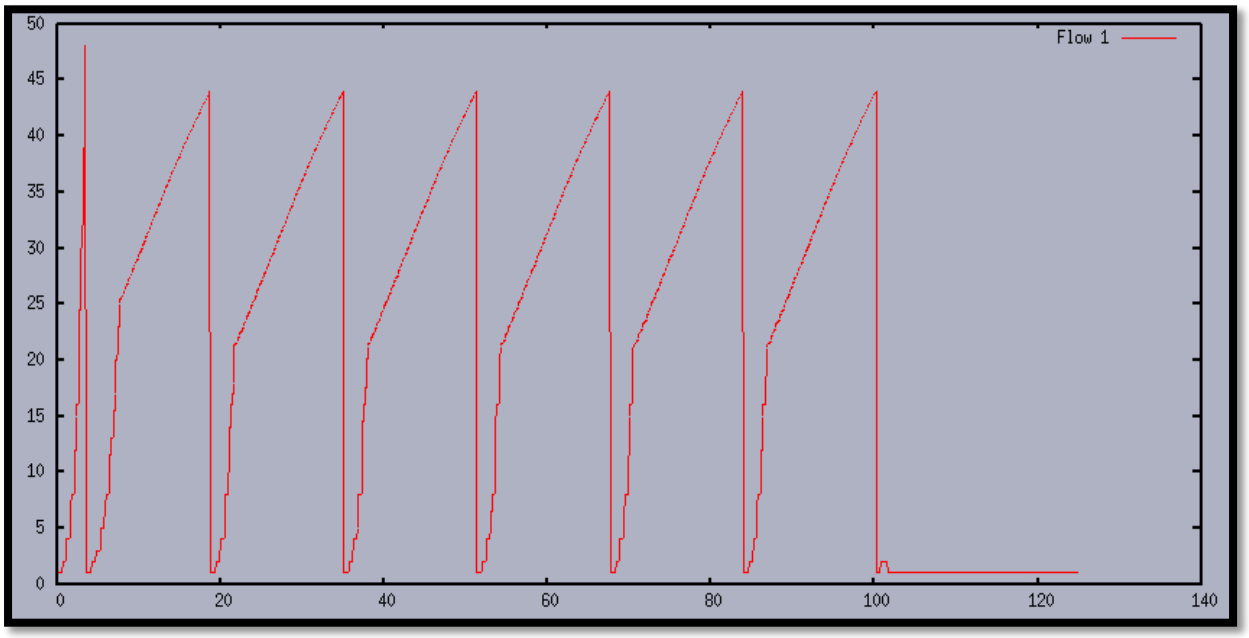
Screen Shots :





Mohd. Aftab Alam
02013302717

The Congestion Window (CWND) Plot for TCP is shown below :



You can see the operation of TCP Tahoe clearly from the above figure :

1. At approximately time 0, TCP Tahoe starts and it is in the slow start mode : the congestion window size increases exponentially
2. At approximately time 5, packet loss is detected
   TCP marks SSThresh = 25 (approximately) and begins another slow start
3. When it reaches CWND = 25 (approximately), the CWND increases linearly – here TCP Tahoe enters the congestion avoidance mode
4. At approximately time 19, TCP Tahoe detects packet loss and begins a slow start **SSThresHold is approximately 22**
5. TCP begins another slow start and so on...