UCS1511 - Networks Lab Name: *Krithika Swaminathan*
AY: *2022-23* Roll No.: *205001057*

Assignment 9 – Congestion Control

Date: 28/10/2022

Aim:

To write a TCL script to simulate the different congestion control mechanisms.

Algorithm:

TCP/Tahoe:

1. Create 3 nodes and the links between the nodes as:
    a. 0 → 1 - 10Mb 10 ms duplex link
    b. 1 → 2 - 2Mb 10 ms duplex link
2. Align the nodes properly.
3. Setup a TCP/Tahoe connection over 0 and 2 and its flow id, window size, packet.
4. Show the simulation in network animator and in the trace file.

TCP/Reno:

1. Create 3 nodes and the links between the nodes as:
    a. 0 → 1 - 10Mb 10 ms duplex link
    b. 1 → 2 - 2Mb 10 ms duplex link
2. Align the nodes properly.
3. Setup a TCP/Reno connection over 0 and 2 and mention the same flow id, window size, packet used for TCP/Tahoe.
4. Show the simulation in network animator and in the trace file.

Code:

TCP/Tahoe:

```
set ns [new Simulator]

$ns color 1 Blue
$ns color 2 Red
```

```
set nf [open out.nam w]
@ns namtrace-all $nf

proc finish {} {
   global ns nf
   $ns flush-trace
   close $nf
   exec nam out.name & exit 0
}

set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]

$ns duplex-link $n0 $n1 10Mb 10ms DropTail
$ns duplex-link $n1 $n2 2Mb 10ms DropTail

$ns queue-limit $n0 $n1 10

$ns duplex-link-op $n0 $n1 orient right
$ns duplex-link-op $n1 $n2 orient right

set tcp [new Agent/TCP]
$tcp set class_ 2
$ns attach-agent $n0 $tcp

set sink [new Agent/TCPSink]
$ns attach-agent $n2 $sink

$ns connect $tcp $sink

$tcp set packetSize_ 1000
$tcp set window_ 65000
$tcp set fid_ 1

set cbr [new Application/Traffic/CBR]
$cbr set packetSize_ 500
$cbr set interval_ 0.001
$cbr attach-agent $tcp

$ns at 0.1 "$cbr start"
$ns at 4.5 "cbr stop"
```

```
$ns at 4.5 "$ns detach-agent $n0 $tcp; $ns detach-agent $n2 $sink"
$ns at 5.0 "finish"

$ns run
```

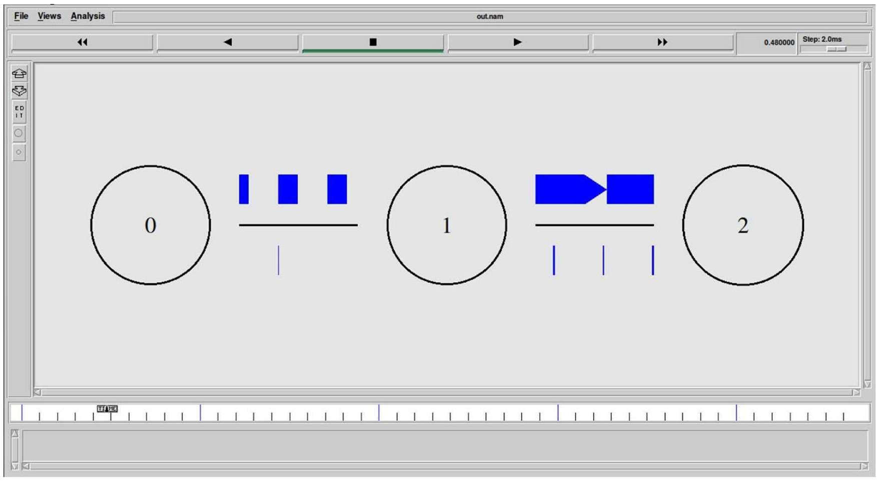
TCP/Reno:

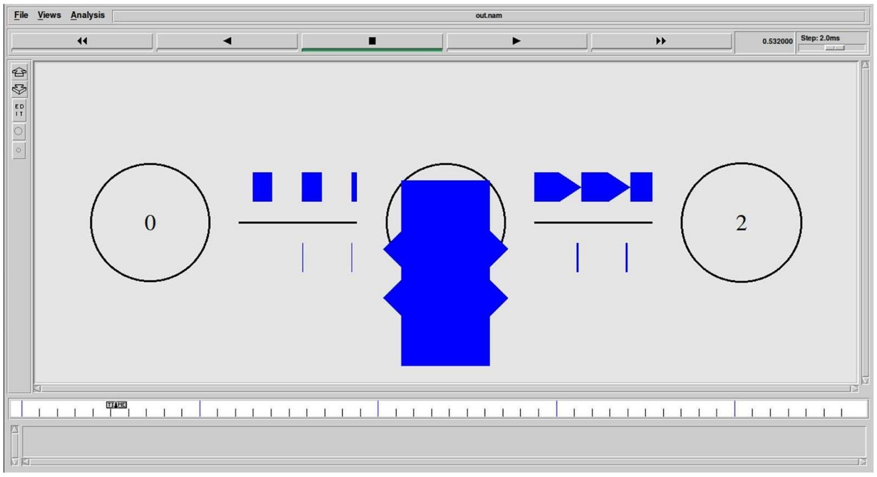
```
set ns [new Simulator]

$ns color 1 Blue
$ns color 2 Red

set nf [open out.nam w]
@ns namtrace-all $nf

proc finish {} {
   global ns nf
   $ns flush-trace
   close $nf
   exec nam out.name & exit 0
}

set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]

$ns duplex-link $n0 $n1 10Mb 10ms DropTail
$ns duplex-link $n1 $n2 2Mb 10ms DropTail

$ns queue-limit $n0 $n1 10

$ns duplex-link-op $n0 $n1 orient right
$ns duplex-link-op $n1 $n2 orient right

set tcp [new Agent/TCP/Reno]
$tcp set class_ 2
$ns attach-agent $n0 $tcp

set sink [new Agent/TCPSink]
$ns attach-agent $n2 $sink

$ns connect $tcp $sink
```

```
$tcp set packetSize_ 1000
$tcp set window_ 65000
$tcp set fid_ 1

set cbr [new Application/Traffic/CBR]
$cbr set packetSize_ 500
$cbr set interval_ 0.001
$cbr attach-agent $tcp

$ns at 0.1 "$cbr start"
$ns at 4.5 "cbr stop"
$ns at 4.5 "$ns detach-agent $n0 $tcp; $ns detach-agent $n2 $sink"
$ns at 5.0 "finish"

$ns run
```

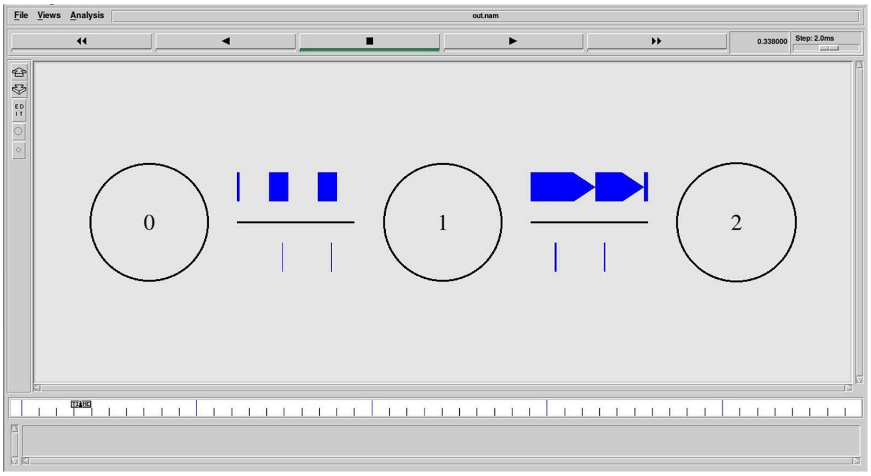
Output:

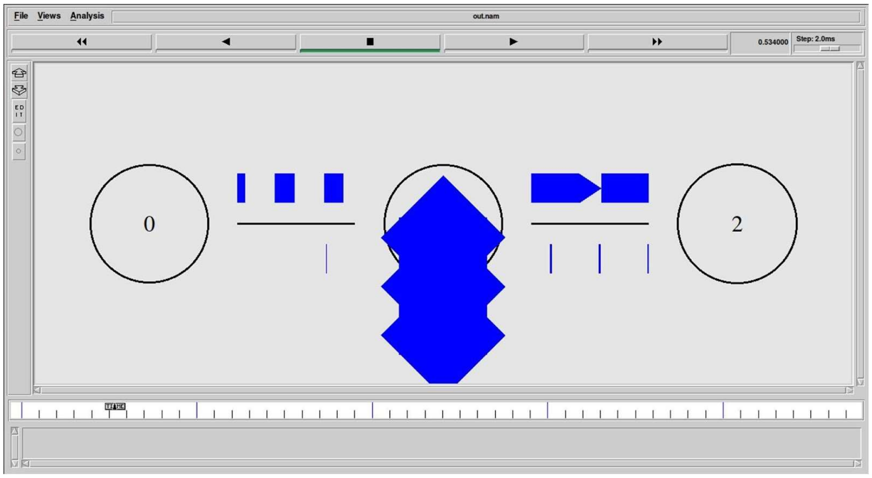
TCP Tahoe:





TCP Reno:





Learning outcomes:

- Congestion control algorithms were understood.
- The implementation of congestion control algorithms using NS2 was understood.