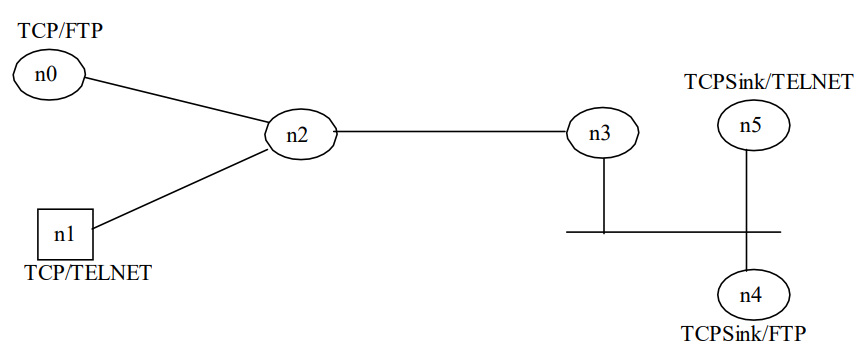
**Lab Experiment 6:**

**Implement an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source / destination.**

**Topology**

****

**Code-**

**#set ns Simulator**

set ns [new Simulator]

**#define color for data flow**

$ns color 1 Blue

$ns color 2 Red

**#open trace file**

set tracefile1 [open lab6.tr w]

set winfile [open winfile w]

$ns trace-all $tracefile1

**#open namtrace file**

set namfile [open lab6.nam w]

$ns namtrace-all $namfile

**#define finish procedure**

proc finish { } {

global ns tracefile1 namfile

$ns flush-trace

close $tracefile1

close $namfile

exec nam lab6.nam &

exit 0

}

**#create 6 nodes**

set n0 [$ns node]

set n1 [$ns node]

set n2 [$ns node]

set n3 [$ns node]

set n4 [$ns node]

set n5 [$ns node]

$n1 shape box

**#create link between nodes**

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

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

$ns simplex-link $n2 $n3 0.3Mb 100ms DropTail

$ns simplex-link $n3 $n2 0.3Mb 100ms DropTail

set lan [$ns newLan "$n3 $n4 $n5" 0.5Mb 40ms LL Queue/DropTail MAC/802\_3]

**#give node position**

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

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

$ns simplex-link-op $n3 $n2 orient left

$ns simplex-link-op $n2 $n3 orient right

**#set queue size of link(n2-n3)**

$ns queue-limit $n2 $n3 20

**#setup tcp connection**

set tcp [new Agent/TCP]

$ns attach-agent $n0 $tcp

set sink [new Agent/TCPSink]

$ns attach-agent $n4 $sink

$ns connect $tcp $sink

$tcp set fid\_ 1

$tcp set packetSize\_ 552

**#set ftp over tcp connection**

set ftp [new Application/FTP]

$ftp attach-agent $tcp

**#setup a TCP1 connection**

set tcp1 [new Agent/TCP]

$ns attach-agent $n1 $tcp1

set sink1 [new Agent/TCPSink]

$ns attach-agent $n5 $sink1

$ns connect $tcp1 $sink1

$tcp1 set fid\_ 2

$tcp1 set packetSize\_ 552

set telnet0 [new Application/Telnet]

$telnet0 attach-agent $tcp1

**#title congestion window1**

set outfile1 [open congestion1.xg w]

puts $outfile1 "TitleText: Congestion Window-- Source \_tcp"

puts $outfile1 "xUnitText: Simulation Time(Secs)"

puts $outfile1 "yUnitText: Congestion WindowSize"

**#title congestion window2**

set outfile2 [open congestion2.xg w]

puts $outfile2 "TitleText: Congestion Window-- Source \_tcp1"

puts $outfile2 "xUnitText: Simulation Time(Secs)"

puts $outfile2 "yUnitText: Congestion WindowSize"

proc plotWindow {tcpSource outfile} {

global ns

set time 0.1

set now [$ns now]

set cwnd [$tcpSource set cwnd\_]

puts $outfile "$now $cwnd"

$ns at [expr $now+$time] "plotWindow $tcpSource $outfile"

}

$ns at 0.1 "plotWindow $tcp $winfile"

$ns at 0.0 "plotWindow $tcp $outfile1"

$ns at 0.1 "plotWindow $tcp1 $outfile2"

$ns at 0.3 "$ftp start"

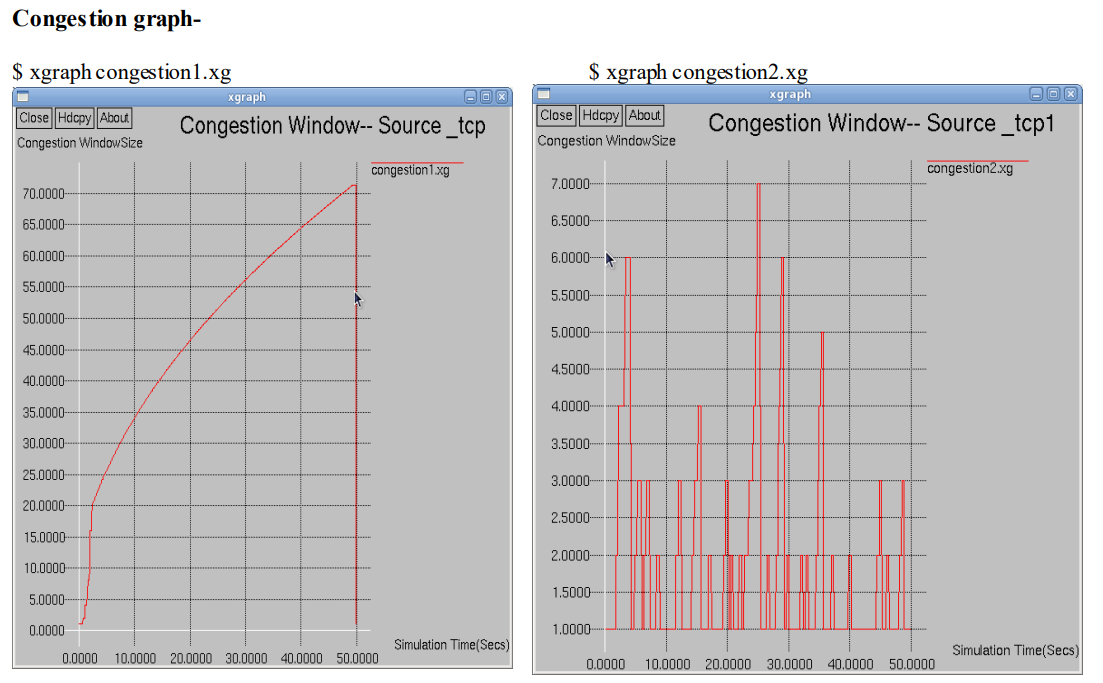
$ns at 0.5 "$telnet0 start"

$ns at 49.0 "$ftp stop"

$ns at 49.1 "$telnet0 stop"

$ns at 50.0 "finish"

$ns run



**Explanation**

The line **set winfile [open winfile w]** in Tcl is used to open a file named "winfile" in write mode.

set winfile: This command **creates a variable named winfile.**

[open winfile w]: This part of the command opens a file named "winfile" in write mode (w). The open command is used to open a file, and w specifies that the file should be opened for writing. If the file does not exist, it will be created; if it does exist, its contents will be truncated.

So, after executing this line, the variable winfile will hold the file handle for the opened file, and you can use it to write data to the file.

**set lan [$ns newLan "$n3 $n4 $n5" 0.5Mb 40ms LL Queue/DropTail MAC/802\_3]**

The line is **creating a local area network (LAN) and associating it with three nodes** ($n3, $n4, and $n5).

set lan: This command **creates a variable named lan** to store the reference to the newly created LAN.

$ns newLan: This command is a **custom function or procedure used to create a new LAN.**

"$n3 $n4 $n5": This **string specifies the nodes that are part of the LAN**. The nodes are $n3, $n4, and $n5.

0.5Mb: **Specifies the bandwidth of the LAN**, in this case, it's set to 0.5 megabits per second.

40ms: **Specifies the propagation delay** of the LAN, here set to 40 milliseconds.

LL: **Specifies the link layer type**, likely set to LL (Link Layer).

Queue/DropTail: **Specifies the type of queue** used in the LAN, here set to a drop-tail queue.

MAC/802\_3: **Specifies the Medium Access Control (MAC) layer type**, here set to **MAC/802\_3, which is commonly used for Ethernet.**

After this line is executed, **the lan variable will hold the reference to the newly created LAN object.** This LAN can be used to model communication and interactions between the specified nodes within the ns-2 simulation.

$tcp set fid\_ 1

$tcp set packetSize\_ 552

* These lines are setting **some parameters for a TCP agent** in an ns-2 simulation.
* $tcp set fid\_ 1: This line **sets the flow ID (fid\_) of the TCP agent ($tcp) to 1**. The flow ID is a **unique identifier assigned to a specific flow** or connection in the simulation. It helps distinguish between different flows of data.
* $tcp set packetSize\_ 552: This line **sets the packet size (packetSize\_) of the TCP agent ($tcp) to 552 bytes**. The packet size is the **size of the data payload in each packet** transmitted by the TCP agent.

set outfile1 [open congestion1.xg w]

puts $outfile1 "TitleText: Congestion Window-- Source \_tcp"

puts $outfile1 "xUnitText: Simulation Time(Secs)"

puts $outfile1 "yUnitText: Congestion WindowSize"

* These lines are used to **create and write header information to a file** named "congestion1.xg."
* set outfile1 [open congestion1.xg w]: This line **creates a file handle (outfile1) by opening a file named "congestion1.xg" in write mode (w).** The file handle will be used for writing data to the file.
* puts $outfile1 "TitleText: Congestion Window-- Source \_tcp": This line **writes a line to the file, specifying the title for the graph.** In this case, it sets the title to "Congestion Window-- Source \_tcp."
* puts $outfile1 "xUnitText: Simulation Time(Secs)": This line **writes a line to the file, specifying the x-axis unit text.** In this case, it sets the x-axis unit text to **"Simulation Time(Secs)."**
* puts $outfile1 "yUnitText: Congestion WindowSize": This line **writes a line to the file, specifying the y-axis unit text.** In this case, it sets the y-axis unit text to "**Congestion WindowSize."**
* These commands are preparing a file for storing data related to congestion window size over time in the context of a network simulation.

The **plotWindow procedure** is designed to be used for plotting the congestion window size of a TCP source over time.

proc plotWindow {tcpSource outfile} { ... }: This line **defines a procedure** named plotWindow that takes two parameters, tcpSource (presumably a TCP source agent) and outfile (the file handle for the output file).

global ns: This line **declares the use of the global variable** $ns, which typically represents the ns-2 simulator instance.

set time 0.1: **Sets the variable time to 0.1.** This is the **time interval at which the procedure will be called to update the plot.**

set now [$ns now]: **Gets the current simulation time** using $ns now and **assigns it to the variable now.**

set cwnd [$tcpSource set cwnd\_]: **Retrieves the congestion window size** (cwnd\_) from the TCP source agent ($tcpSource).

puts $outfile "$now $cwnd": **Writes a line to the output file,** containing the current simulation time and the congestion window size.

$ns at [expr $now + $time] "plotWindow $tcpSource $outfile": **Schedules the plotWindow procedure to be called again after the specified time interval ($time).**

This **creates a recursive loop**, effectively updating and writing the congestion window size over time.

This procedure is designed to be called **periodically to capture the congestion window size at different simulation times.**