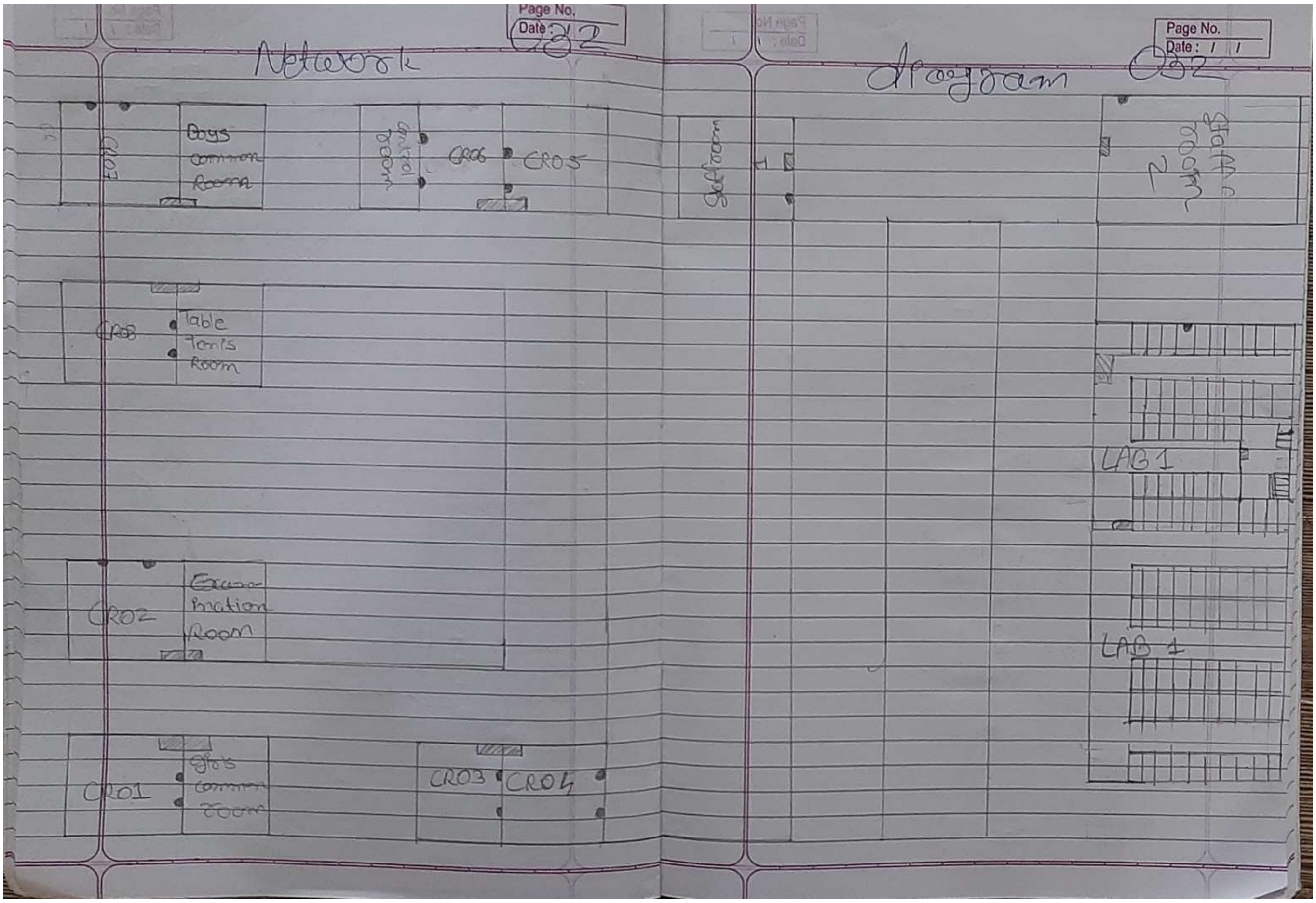
BABU MADHAV INSTITUTE OF INFORMATION TECHNOLOGY, UTU
Integrated M.Sc.(IT) / B.Sc.(IT)

060010715 - Wireless Networks
Experiment List: 01

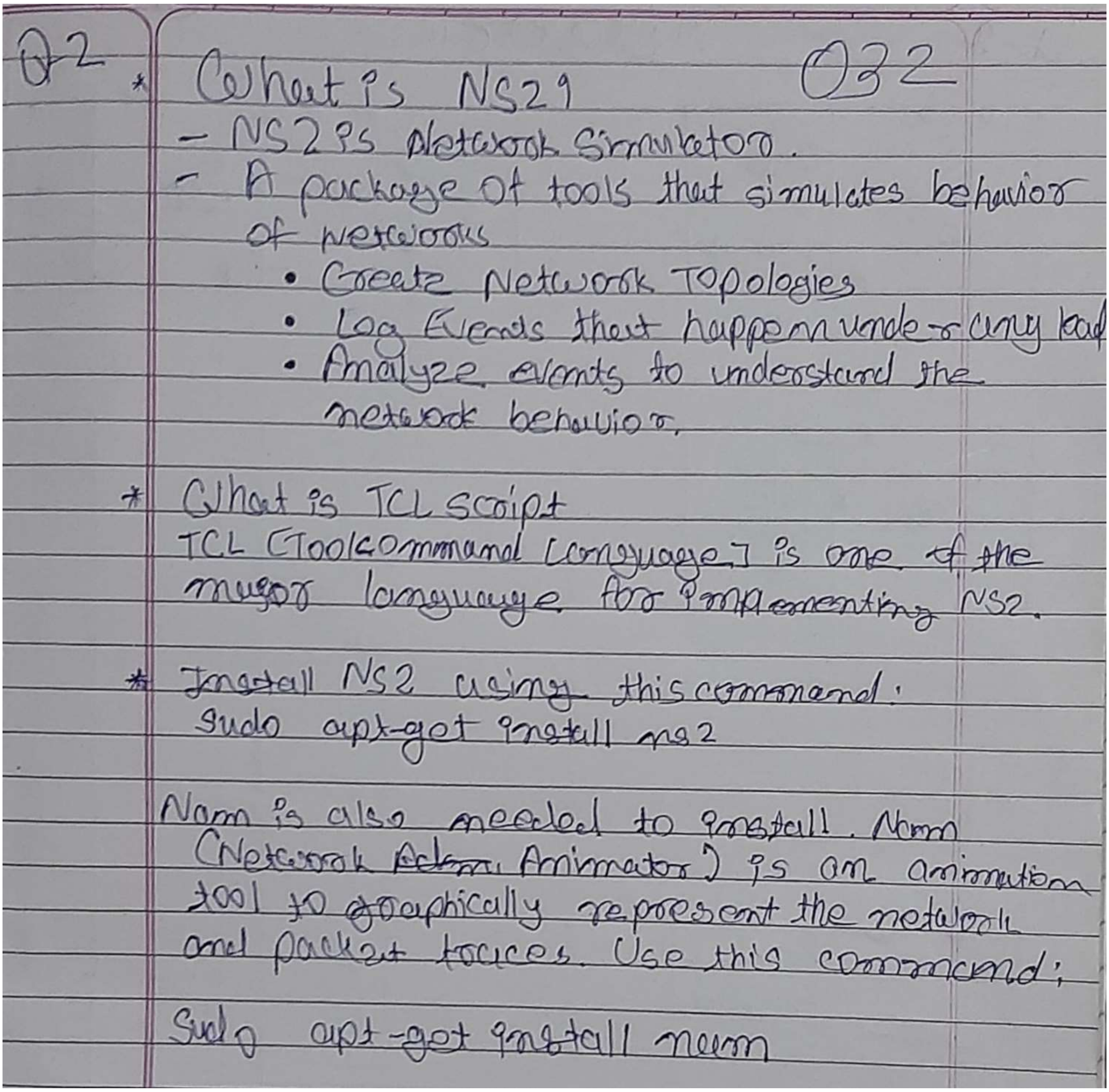
/* Enrollment No. : 201906100110032
Name : Choksi Abhishek Premalkumar
Class : M.Sc(IT) – 7th
Date : 19-07-2022
*/

1. Network Diagram



2. Study of NS2 and TCL Script. Installation of NS2

What is NS2?



Q-2 032

* What is NS2?

- NS2 is Network Simulator.
- A package of tools that simulates behavior of networks
  - • Create Network Topologies
  - • Log Events that happen under any load
  - • Analyze events to understand the network behavior.

* What is TCL script

TCL [Toolcommand Language] is one of the major language for implementing NS2.

* Install NS2 using this command:

sudo apt-get install ns2

Nam is also needed to install. Nam (Network ~~Adm~~. Animator) is an animation tool to graphically represent the network and packet traces. Use this command;

sudo apt-get install nam

3. Create Simple Node Topology with Traffic

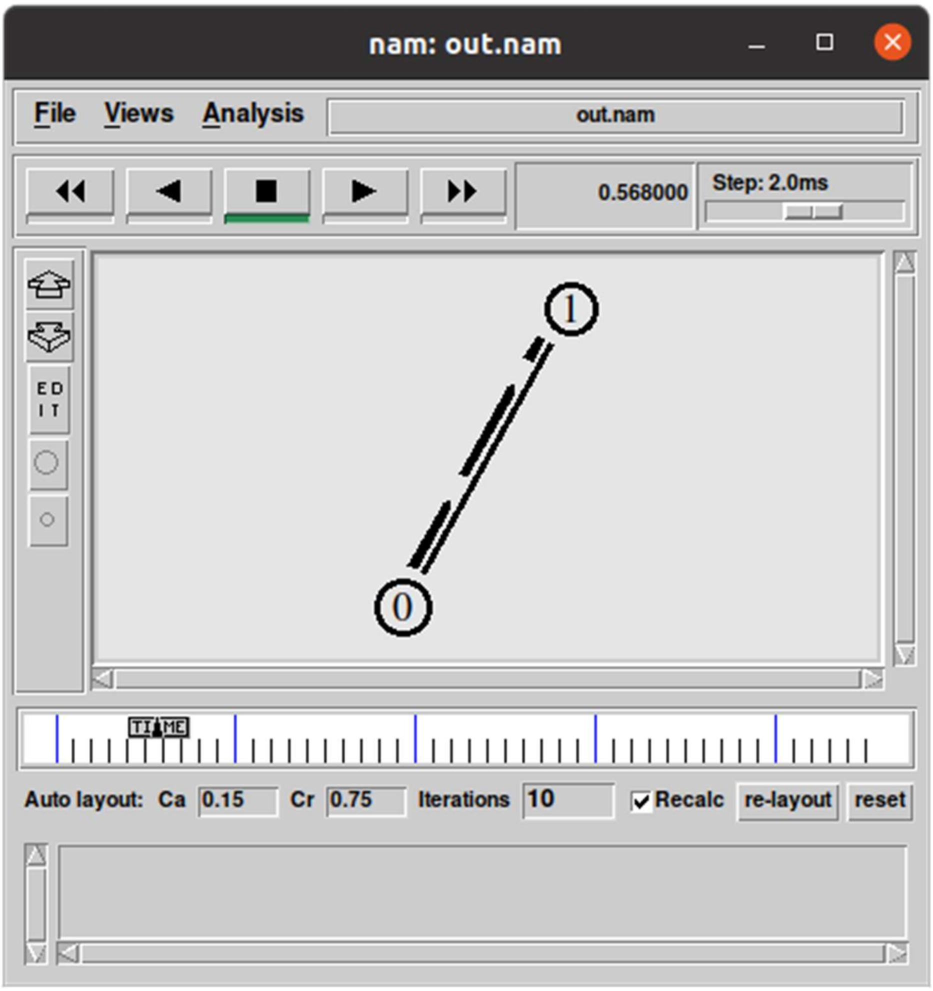
SimpleNode.tcl

```
#Create a simulator object
set ns [new Simulator]

#open the nam trace file
set nf [open out.nam w]
$ns namtrace-all $nf

#Define a 'finish' procedure
```

```
proc finish {} {
        global ns nf
        $ns flush-trace

        #close the trace file
        close $nf

        #execute nam on the trace file
        exec nam out.nam &

        exit 0
}

#Create two nodes
set n0 [$ns node]
set n1 [$ns node]


#Create a duplex link between the nodes
$ns duplex-link $n0 $n1 1Mb 10ms DropTail

#create a udp agent and attach it to node n0
set udp0 [new Agent/UDP]
$ns attach-agent $n0 $udp0

#Create a CBR traffic source and attach it to udp0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0

#create a Null agent(a traffic sink) and attach it to node n1
set null0 [new Agent/Null]
$ns attach-agent $n1 $null0

#Connect the traffic source to the sink
$ns connect $udp0 $null0

#Schedule events for CBR traffic
$ns at 0.5 "$cbr0 start"
$ns at 4.5 "$cbr0 stop"

#Call the finish procedure after 5 seconds of simulation time
```

```
$ns at 5.0 "finish"

#Run the simulation
$ns run
```

Output


nam: out.nam
File
Views
Analysis
out.nam
0.568000
Step: 2.0ms
ED IT
1
0
TIME
Auto layout: Ca 0.15 Cr 0.75 Iterations 10 ☑ Recalc re-layout reset

4. Create Different Topologies (Star Topology and Ring Topology)

starTopology.tcl

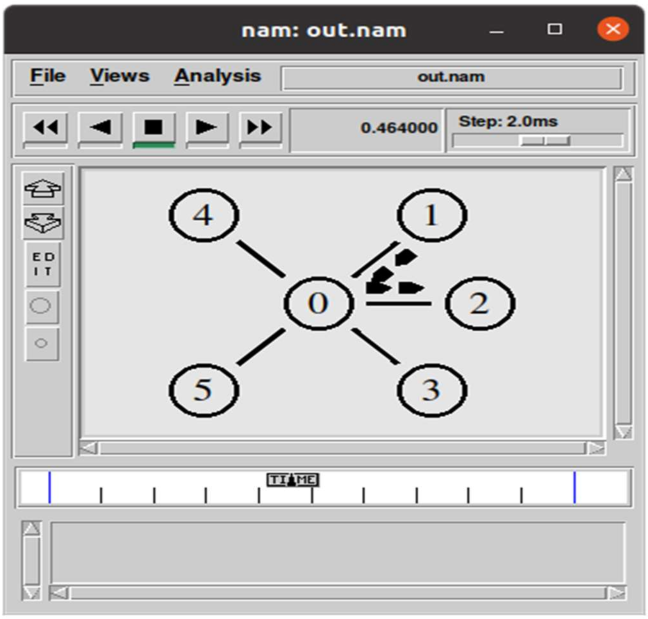
```
#Create a simulator object
set ns [new Simulator]

#Define different colors for data flows (for NAM)
$ns color 1 Blue
$ns color 2 Red

#open the nam trace file
set nf [open out.nam w]
$ns namtrace-all $nf
```

```
#Define a 'finish' procedure
proc finish {} {
        global ns nf
        $ns flush-trace

        #close the trace file
        close $nf

        #execute nam on the trace file
        exec nam out.nam &

        exit 0
}

#Create two 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]
#set n6 [$ns node]

#Create a duplex link between the nodes
$ns duplex-link $n0 $n1 1Mb 10ms DropTail
$ns duplex-link $n0 $n2 1Mb 10ms DropTail
$ns duplex-link $n0 $n3 1Mb 10ms DropTail
$ns duplex-link $n0 $n4 1Mb 10ms DropTail
$ns duplex-link $n0 $n5 1Mb 10ms DropTail
#$ns duplex-link $n0 $n6 1Mb 10ms DropTail

#Give node position (for NAM)
$ns duplex-link-op $n0 $n1 orient right-up
$ns duplex-link-op $n0 $n2 orient right
$ns duplex-link-op $n0 $n3 orient right-down
$ns duplex-link-op $n0 $n4 orient left-up
$ns duplex-link-op $n0 $n5 orient left-down
#$ns duplex-link-op $n0 $n6 orient left

#create a udp agent and attach it to node
# 1 to 0
set udp0 [new Agent/UDP]
```

```
$ns attach-agent $n1 $udp0

# 0 to 2
set udp1 [new Agent/UDP]
$ns attach-agent $n0 $udp1

# 0 to 3
set udp2 [new Agent/UDP]
$ns attach-agent $n0 $udp2

#Create a CBR traffic source and attach it to udp
# 1 to 0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0

# 0 to 2
set cbr1 [new Application/Traffic/CBR]
$cbr1 set packetSize_ 500
$cbr1 set interval_ 0.005
$cbr1 attach-agent $udp1

# 0 to 3
set cbr2 [new Application/Traffic/CBR]
$cbr2 set packetSize_ 500
$cbr2 set interval_ 0.005
$cbr2 attach-agent $udp2

#create a Null agent(a traffic sink) and attach it to node
# 1 to 0
set null0 [new Agent/Null]
$ns attach-agent $n0 $null0

# 0 to 2
set null1 [new Agent/Null]
$ns attach-agent $n2 $null1

# 0 to 3
set null2 [new Agent/Null]
$ns attach-agent $n3 $null2
#Connect the traffic source to the sink
# 1 to 0
```

```
$ns connect $udp0 $null0
# 0 to 2
$ns connect $udp1 $null1

# 0 to 3
$ns connect $udp2 $null2

#Schedule events for CBR traffic
# 1 to 0
$ns at 0.1 "$cbr0 start"
$ns at 0.5 "$cbr0 stop"
# 0 to 2
$ns at 0.1 "$cbr1 start"
$ns at 0.5 "$cbr1 stop"
# 1 to 0
$ns at 0.6 "$cbr0 start"
$ns at 1.0 "$cbr0 stop"

# 0 to 3
$ns at 0.6 "$cbr2 start"
$ns at 1.0 "$cbr2 stop"

#Call the finish procedure after 5 seconds of simulation time
$ns at 1.0 "finish"

#Run the simulation
$ns run
```

Output

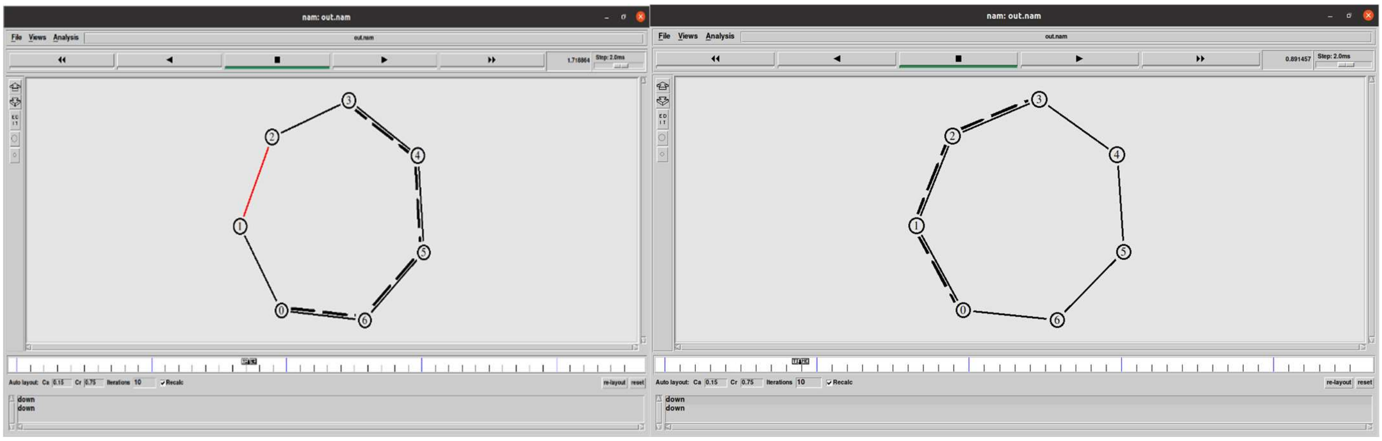


ringTopology.tcl

```
#Create a simulator object
set ns [new Simulator]

#Tell the simulator to use dynamic routing
$ns rtproto DV

#Open the nam trace file
set nf [open out.nam w]
$ns namtrace-all $nf



#Define a 'finish' procedure
proc finish {} {
        global ns nf
        $ns flush-trace
            #Close the trace file
        close $nf
            #Execute nam on the trace file
        exec nam out.nam &
        exit 0
}

#Create seven nodes
for {set i 0} {$i < 7} {incr i} {
        set n($i) [$ns node]
}



#Create links between the nodes
for {set i 0} {$i < 7} {incr i} {
        $ns duplex-link $n($i) $n([expr ($i+1)%7]) 1Mb 10ms DropTail
}

#Create a UDP agent and attach it to node n(0)
set udp0 [new Agent/UDP]
$ns attach-agent $n(0) $udp0

# Create a CBR traffic source and attach it to udp0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0
```

```
#Create a Null agent (a traffic sink) and attach it to node n(3)
set null0 [new Agent/Null]
$ns attach-agent $n(3) $null0

#Connect the traffic source with the traffic sink
$ns connect $udp0 $null0

#Schedule events for the CBR agent and the network dynamics
$ns at 0.5 "$cbr0 start"
$ns rtmodel-at 1.0 down $n(1) $n(2)
$ns rtmodel-at 2.0 up $n(1) $n(2)
$ns at 4.5 "$cbr0 stop"
#Call the finish procedure after 5 seconds of simulation time
$ns at 5.0 "finish"

#Run the simulation
$ns run
```

Output



```
down
down
```

```
down
down
```

5. Create Topology with UDP Agent

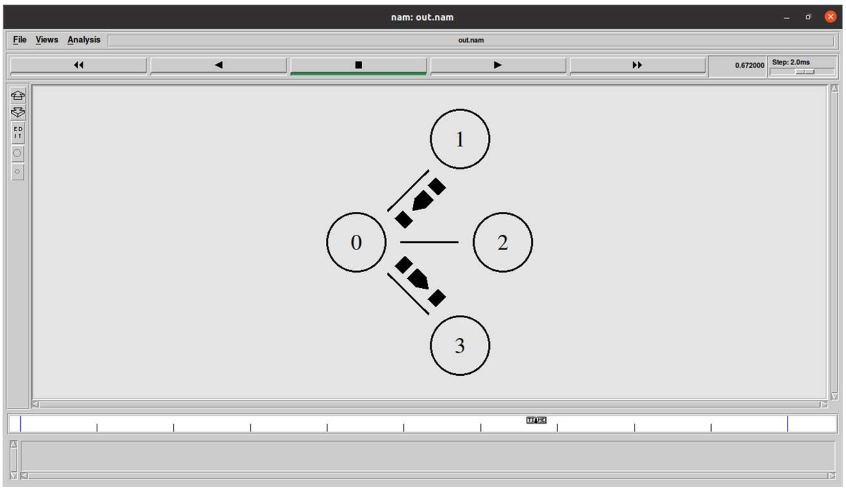
udpagent.tcl

```
#Create a simulator object
set ns [new Simulator]

#open the nam trace file
set nf [open out.nam w]
$ns namtrace-all $nf

#Define a 'finish' procedure
proc finish {} {
global ns nf
$ns flush-trace
#close the trace file
close $nf

#execute nam on the trace file
exec nam out.nam &

exit 0
}

#Create two nodes
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]

#Create a duplex link between the nodes
$ns duplex-link $n0 $n1 1Mb 10ms DropTail
$ns duplex-link $n0 $n2 1Mb 10ms DropTail
$ns duplex-link $n0 $n3 1Mb 10ms DropTail

#Give node position (for NAM)
$ns duplex-link-op $n0 $n1 orient right-up
$ns duplex-link-op $n0 $n2 orient right
$ns duplex-link-op $n0 $n3 orient right-down

#create a udp agent and attach it to node
# 1 to 0
set udp0 [new Agent/UDP]
$ns attach-agent $n1 $udp0

# 0 to 2
set udp1 [new Agent/UDP]
$ns attach-agent $n0 $udp1

# 0 to 3
set udp2 [new Agent/UDP]
$ns attach-agent $n0 $udp2

#Create a CBR traffic source and attach it to udp
# 1 to 0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
```

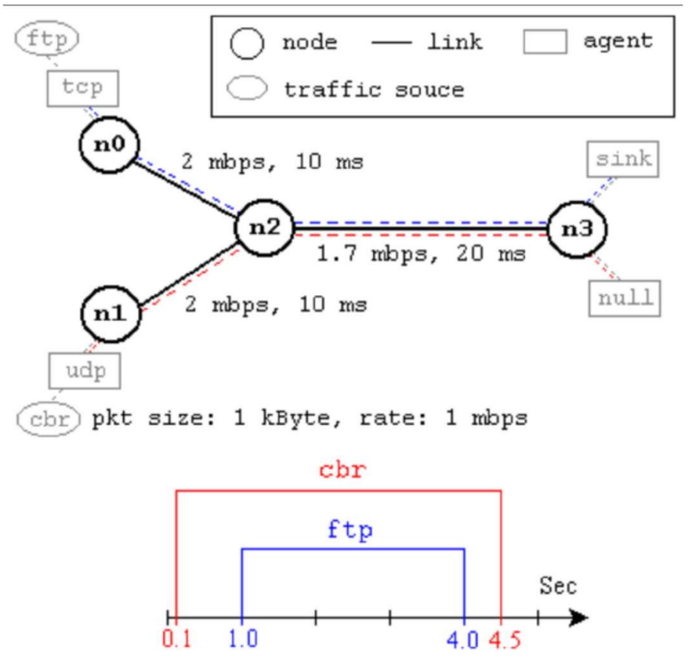
```
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0

# 0 to 2
set cbr1 [new Application/Traffic/CBR]
$cbr1 set packetSize_ 500
$cbr1 set interval_ 0.005
$cbr1 attach-agent $udp1

# 0 to 3
set cbr2 [new Application/Traffic/CBR]
$cbr2 set packetSize_ 500
$cbr2 set interval_ 0.005
$cbr2 attach-agent $udp2

#create a Null agent(a traffic sink) and attach it to node
# 1 to 0
set null0 [new Agent/Null]
$ns attach-agent $n0 $null0

# 0 to 2
set null1 [new Agent/Null]
$ns attach-agent $n2 $null1

# 0 to 3
set null2 [new Agent/Null]
$ns attach-agent $n3 $null2

#Connect the traffic source to the sink
# 1 to 0
$ns connect $udp0 $null0
# 0 to 2
$ns connect $udp1 $null1

# 0 to 3
$ns connect $udp2 $null2

#Schedule events for CBR traffic
# 1 to 0
$ns at 0.1 "$cbr0 start"
$ns at 0.5 "$cbr0 stop"
# 0 to 2
$ns at 0.1 "$cbr1 start"
```

```
$ns at 0.5 "$cbr1 stop"
# 1 to 0
$ns at 0.6 "$cbr0 start"
$ns at 1.0 "$cbr0 stop"

# 0 to 3
$ns at 0.6 "$cbr2 start"
$ns at 1.0 "$cbr2 stop"

#Call the finish procedure after 5 seconds of simulation time
$ns at 1.0 "finish"

#Run the simulation
$ns run
```

Output



6. Create Topology as below



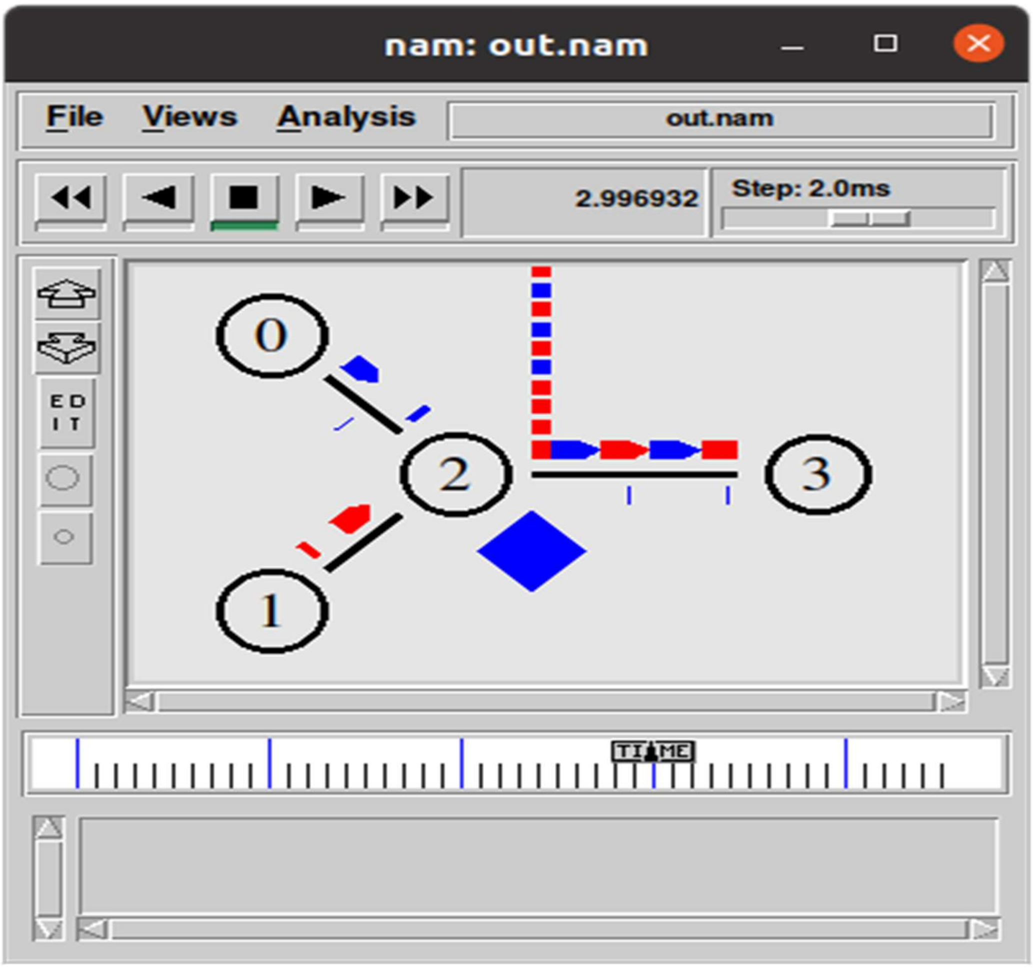
19_07_2022.tcl

```
#Create a simulator object
set ns [new Simulator]

#Define different colors for data flows (for NAM)
$ns color 1 Blue
$ns color 2 Red

#Open the NAM trace file
set nf [open out.nam w]
$ns namtrace-all $nf

#Define a 'finish' procedure
proc finish {} {
   global ns nf
   $ns flush-trace
   #Close the NAM trace file
   close $nf
   #Execute NAM on the trace file
   exec nam out.nam &
   exit 0
}

#Create four nodes
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]

#Create links between the nodes
$ns duplex-link $n0 $n2 1.8Mb 10ms DropTail
$ns duplex-link $n1 $n2 1.8Mb 10ms DropTail
$ns duplex-link $n2 $n3 1.7Mb 20ms DropTail

#Set Queue Size of link (n2-n3) to 10
$ns queue-limit $n2 $n3 10

#Give node position (for NAM)
$ns duplex-link-op $n0 $n2 orient right-down
$ns duplex-link-op $n1 $n2 orient right-up
$ns duplex-link-op $n2 $n3 orient right

#Monitor the queue for link (n2-n3). (for NAM)
```

```
$ns duplex-link-op $n2 $n3 queuePos 0.5

#Setup a TCP connection
set tcp [new Agent/TCP]
$tcp set class_ 2
$ns attach-agent $n0 $tcp
set sink [new Agent/TCPSink]
$ns attach-agent $n3 $sink
$ns connect $tcp $sink
$tcp set fid_ 1

#Setup a FTP over TCP connection
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ftp set type_ FTP

#Setup a UDP connection
set udp [new Agent/UDP]
$ns attach-agent $n1 $udp
set null [new Agent/Null]
$ns attach-agent $n3 $null
$ns connect $udp $null
$udp set fid_ 2

#Setup a CBR over UDP connection
set cbr [new Application/Traffic/CBR]
$cbr attach-agent $udp
$cbr set type_ CBR
$cbr set packet_size_ 1000
$cbr set rate_ 1mb
$cbr set random_ false

#Schedule events for the CBR and FTP agents
$ns at 0.1 "$cbr start"
$ns at 4.5 "$cbr stop"
$ns at 1.0 "$ftp start"
$ns at 4.0 "$ftp stop"



#Detach tcp and sink agents (not really necessary)
$ns at 4.5 "$ns detach-agent $n0 $tcp ; $ns detach-agent $n3 $sink"

#Call the finish procedure after 5 seconds of simulation time
```

```
$ns at 5.0 "finish"

#Print CBR packet size and interval
puts "CBR packet size = [$cbr set packet_size_]"
puts "CBR interval = [$cbr set interval_]"

#Run the simulation
$ns run
```

Output



BABU MADHAV INSTITUTE OF INFORMATION TECHNOLOGY, UTU
Integrated M.Sc.(IT) / B.Sc.(IT)

060010715 - Wireless Networks
Experiment List: 02 (Wireless Practical)

/* Enrollment No. : 201906100110032
Name : Choksi Abhishek Premalkumar
Class : M.Sc(IT) – 7th
Date : 04-11-2022
***/**

1. Create Simple Two Node and Establish Traffic.

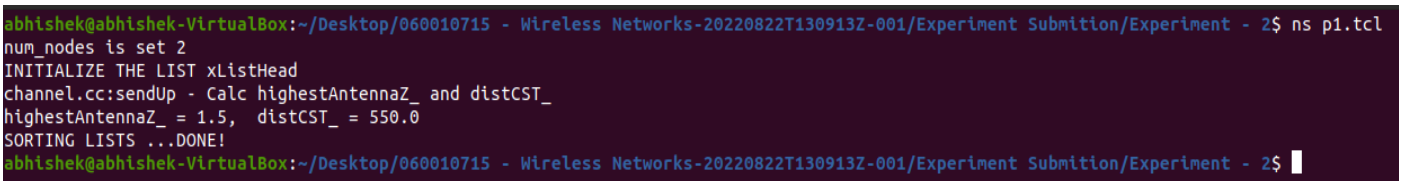
p1.tcl

```
# P1 Create Simple Two Node and Establish Traffic
#===================================
#    Simulation parameters setup
#===================================
set val(chan)   Channel/WirelessChannel    ;# channel type
set val(prop)   Propagation/TwoRayGround   ;# radio-propagation model
set val(netif)  Phy/WirelessPhy            ;# network interface type
set val(mac)    Mac/802_11                 ;# MAC type
set val(ifq)    Queue/DropTail/PriQueue    ;# interface queue type
set val(ll)     LL                         ;# link layer type
set val(ant)    Antenna/OmniAntenna        ;# antenna model
set val(ifqlen) 50                         ;# max packet in ifq
set val(nn)     2                          ;# number of mobilenodes
set val(rp)     DSDV                       ;# routing protocol
set val(x)      825                        ;# X dimension of topography
set val(y)      515                        ;# Y dimension of topography
set val(stop)   5.0                        ;# time of simulation end

#===================================
#       Initialization
#===================================
#Create a ns simulator
set ns [new Simulator]

#Setup topography object
set topo       [new Topography]
$topo load_flatgrid $val(x) $val(y)
create-god $val(nn)
```

```
#Open the NS trace file
set tracefile [open out.tr w]
$ns trace-all $tracefile

#Open the NAM trace file
set namfile [open out.nam w]
$ns namtrace-all $namfile
$ns namtrace-all-wireless $namfile $val(x) $val(y)
set chan [new $val(chan)];#Create wireless channel

#===================================
#     Mobile node parameter setup
#===================================
$ns node-config -adhocRouting  $val(rp) \
                -llType        $val(ll) \
                -macType       $val(mac) \
                -ifqType       $val(ifq) \
                -ifqLen        $val(ifqlen) \
                -antType       $val(ant) \
                -propType      $val(prop) \
                -phyType       $val(netif) \
                -channel       $chan \
                -topoInstance  $topo \
                -agentTrace    ON \
                -routerTrace   ON \
                -macTrace      ON \
                -movementTrace ON

#===================================
#        Nodes Definition
#===================================
#Create 2 nodes
set n0 [$ns node]
$n0 set X_ 369
$n0 set Y_ 406
$n0 set Z_ 0.0
$ns initial_node_pos $n0 20
set n1 [$ns node]
$n1 set X_ 725
$n1 set Y_ 415
$n1 set Z_ 0.0
$ns initial_node_pos $n1 20
```

```
#===================================
#     Agents Definition
#===================================
set udp0 [new Agent/UDP]
$ns attach-agent $n0 $udp0

#===================================
#     Applications Definition
#===================================
#Setup a CBR Application over UDP connection
set cbr0 [new Application/Traffic/CBR]
$cbr0 attach-agent $udp0
$cbr0 set packetSize_ 1000
$cbr0 set rate_ 1.0Mb
$cbr0 set random_ null

$ns at 0.5 "$cbr0 start"
$ns at 4.5 "$cbr0 stop"
#===================================
#     Termination
#===================================
#Define a 'finish' procedure
proc finish {} {
   global ns tracefile namfile
   $ns flush-trace
   close $tracefile
   close $namfile
   exec nam out.nam &
   exit 0
}
for {set i 0} {$i < $val(nn) } { incr i } {
   $ns at $val(stop) "\$n$i reset"
}
$ns at $val(stop) "$ns nam-end-wireless $val(stop)"
$ns at $val(stop) "finish"
$ns at $val(stop) "puts \"done\" ; $ns halt"
$ns run
```

Output



```
abhishek@abhishek-VirtualBox:~/Desktop/060010715 - Wireless Networks-20220822T130913Z-001/Experiment Submition/Experiment - 2$ ns p1.tcl
num_nodes is set 2
INITIALIZE THE LIST xListHead
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 550.0
SORTING LISTS ...DONE!
abhishek@abhishek-VirtualBox:~/Desktop/060010715 - Wireless Networks-20220822T130913Z-001/Experiment Submition/Experiment - 2$
```

2. Create Piconet Topology

p2.tcl

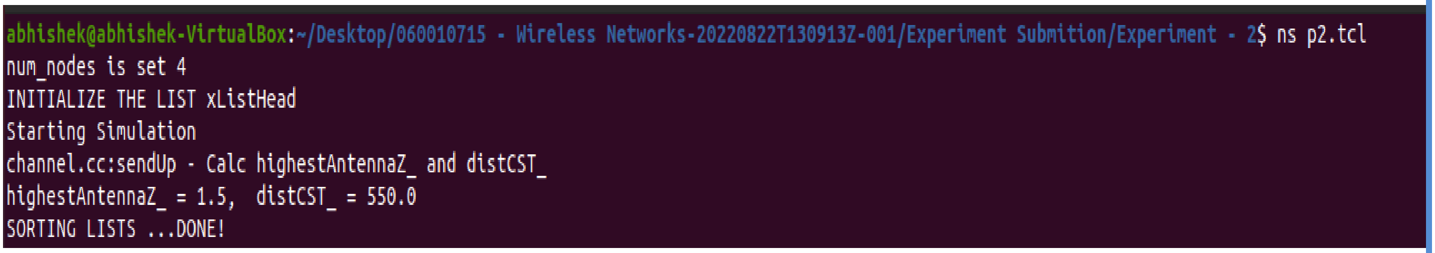
```
# P2 Piconet Topology
#===================================
#    Simulation parameters setup
#===================================
set val(chan)   Channel/WirelessChannel    ;# channel type
set val(prop)   Propagation/TwoRayGround   ;# radio-propagation model
set val(netif)  Phy/WirelessPhy            ;# network interface type
set val(mac)    Mac/802_11                 ;# MAC type
set val(ifq)    Queue/DropTail/PriQueue    ;# interface queue type
set val(ll)     LL                         ;# link layer type
set val(ant)    Antenna/OmniAntenna        ;# antenna model
set val(ifqlen) 50                         ;# max packet in ifq
set val(nn)     4                          ;# number of mobilenodes
set val(rp)     DSDV                       ;# routing protocol
set val(x)      825                        ;# X dimension of topography
set val(y)      515                        ;# Y dimension of topography
set val(stop)   5.0                        ;# time of simulation end

#===================================
#       Initialization
#===================================
#Create a ns simulator
set ns [new Simulator]

#Setup topography object
set topo       [new Topography]
$topo load_flatgrid $val(x) $val(y)
create-god $val(nn)

#Open the NS trace file
set tracefile [open out.tr w]
$ns trace-all $tracefile

#Open the NAM trace file
set namfile [open out.nam w]
$ns namtrace-all $namfile
$ns namtrace-all-wireless $namfile $val(x) $val(y)
set chan [new $val(chan)];#Create wireless channel
```

```
#===================================
#    Mobile node parameter setup
#===================================
$ns node-config -adhocRouting  $val(rp) \
            -llType       $val(ll) \
            -macType      $val(mac) \
            -ifqType      $val(ifq) \
            -ifqLen       $val(ifqlen) \
            -antType      $val(ant) \
            -propType     $val(prop) \
            -phyType      $val(netif) \
            -channel      $chan \
            -topoInstance  $topo \
            -agentTrace    ON \
            -routerTrace   ON \
            -macTrace      ON \
            -movementTrace ON

#===================================
#       Nodes Definition
#===================================
#Create 4 nodes
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]

#initial coordinator of the nodes
$n0 set X_ 369.0
$n0 set Y_ 300.0
$n0 set Z_ 0.0
$ns initial_node_pos $n0 40

$n1 set X_ 369.0
$n1 set Y_ 400.0
$n1 set Z_ 0.0
$ns initial_node_pos $n1 40

$n2 set X_ 280.0
$n2 set Y_ 200.0
$n2 set Z_ 0.0
$ns initial_node_pos $n2 40
```

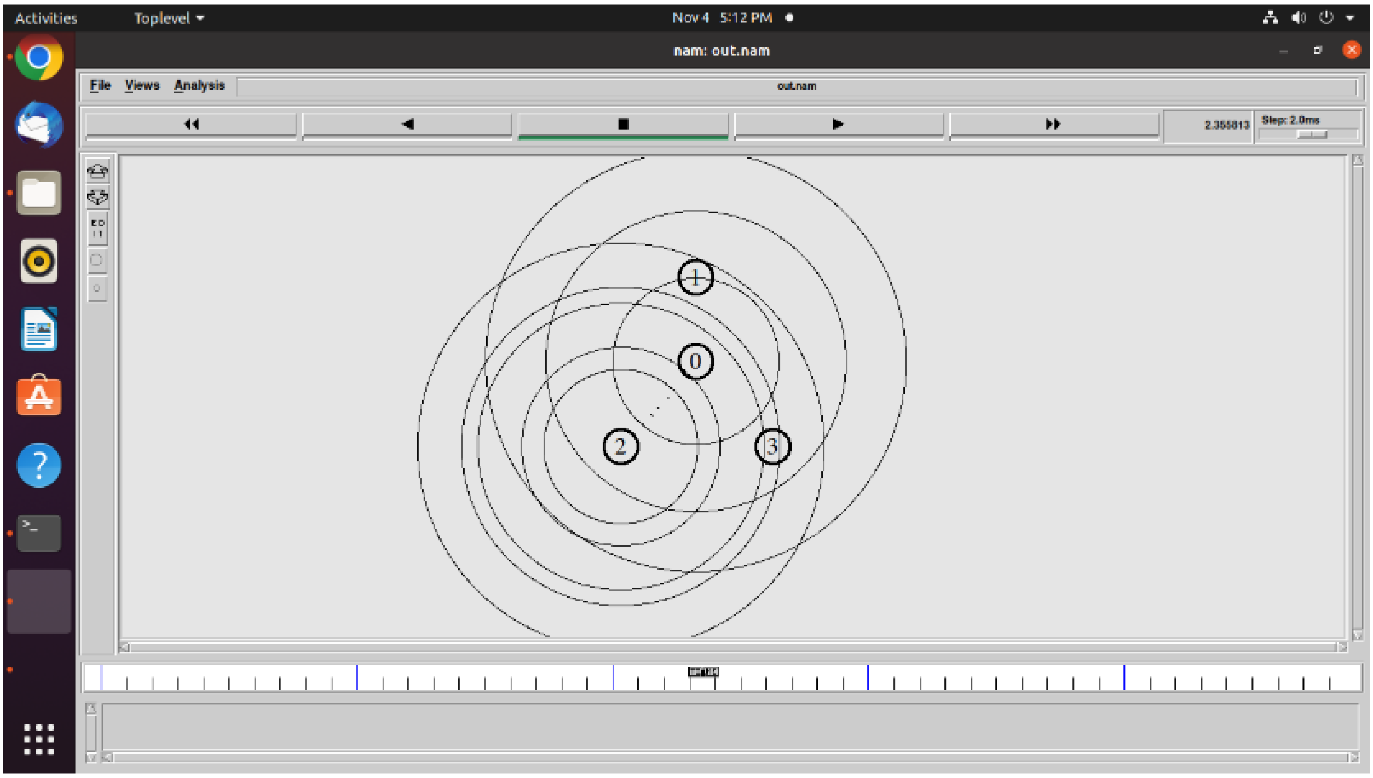
```
$n3 set X_ 460.0
$n3 set Y_ 200.0
$n3 set Z_ 0.0
$ns initial_node_pos $n3 40

#===================================
#     Agents Definition
#===================================
#===================================
#     Applications Definition
#===================================
#Setup a CBR Application over UDP connection
set tcp [new Agent/TCP]
set sink [new Agent/TCPSink]
$ns attach-agent $n0 $tcp
$ns attach-agent $n1 $sink
$ns connect $tcp $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ns at 1.0 "$ftp start"
$ns at 2.0 "$ftp stop"

set udp [new Agent/UDP]
set null [new Agent/Null]
$ns attach-agent $n0 $udp
$ns attach-agent $n2 $null
$ns connect $udp $null
set cbr [new Application/Traffic/CBR]
$cbr attach-agent $udp
$ns at 2.3 "$cbr start"
$ns at 3.3 "$cbr stop"

set tcp [new Agent/TCP]
set sink [new Agent/TCPSink]
$ns attach-agent $n0 $tcp
$ns attach-agent $n3 $sink
$ns connect $tcp $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp

$ns at 3.6 "$ftp start"
$ns at 4.6 "$ftp stop"
```

```
$ns at 5.0 "finish"
#===================================
#       Termination
#===================================
#Define a 'finish' procedure
proc finish {} {
   global ns tracefile namfile
   $ns flush-trace
   close $tracefile
   close $namfile
   exec nam out.nam &
   exit 0
}
$ns run
```

Output



```
abhishek@abhishek-VirtualBox:~/Desktop/060010715 - Wireless Networks-20220822T130913Z-001/Experiment Submition/Experiment - 2$ ns p2.tcl
num_nodes is set 4
INITIALIZE THE LIST xListHead
Starting Simulation
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 550.0
SORTING LISTS ...DONE!
```



3. Create Topology using UDP and TCP agents in a single network.

p3.tcl

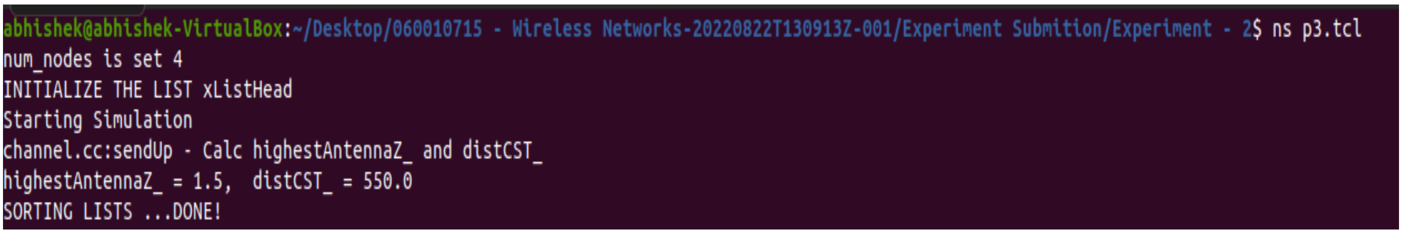
```
# P3 Create Topology using UDP And TCP in Single Network
#===================================
#    Simulation parameters setup
#===================================
set val(chan)   Channel/WirelessChannel    ;# channel type
set val(prop)   Propagation/TwoRayGround   ;# radio-propagation model
set val(netif)  Phy/WirelessPhy            ;# network interface type
set val(mac)    Mac/802_11                 ;# MAC type
set val(ifq)    Queue/DropTail/PriQueue    ;# interface queue type
set val(ll)     LL                         ;# link layer type
set val(ant)    Antenna/OmniAntenna        ;# antenna model
set val(ifqlen) 50                         ;# max packet in ifq
set val(nn)     4                          ;# number of mobilenodes
set val(rp)     DSDV                       ;# routing protocol
set val(x)      825                        ;# X dimension of topography
set val(y)      515                        ;# Y dimension of topography
set val(stop)   5.0                        ;# time of simulation end

#===================================
#        Initialization
#===================================
#Create a ns simulator
set ns [new Simulator]

#Setup topography object
set topo       [new Topography]
$topo load_flatgrid $val(x) $val(y)
create-god $val(nn)

#Open the NS trace file
set tracefile [open out.tr w]
$ns trace-all $tracefile

#Open the NAM trace file
set namfile [open out.nam w]
$ns namtrace-all $namfile
$ns namtrace-all-wireless $namfile $val(x) $val(y)
set chan [new $val(chan)];#Create wireless channel
```

```
#===================================
#    Mobile node parameter setup
#===================================
$ns node-config -adhocRouting  $val(rp) \
            -llType       $val(ll) \
            -macType      $val(mac) \
            -ifqType      $val(ifq) \
            -ifqLen       $val(ifqlen) \
            -antType      $val(ant) \
            -propType     $val(prop) \
            -phyType      $val(netif) \
            -channel      $chan \
            -topoInstance  $topo \
            -agentTrace    ON \
            -routerTrace   ON \
            -macTrace      ON \
            -movementTrace ON

#===================================
#      Nodes Definition
#===================================
#Create 2 nodes
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]

$n0 random-motion 0
$n1 random-motion 0
$n2 random-motion 0
$n3 random-motion 0

$ns initial_node_pos $n0 20
$ns initial_node_pos $n1 20
$ns initial_node_pos $n2 20
$ns initial_node_pos $n3 20

#initial coordinator of the nodes
$n0 set X_ 10.0
$n0 set Y_ 20.0
$n0 set Z_ 0.0
```

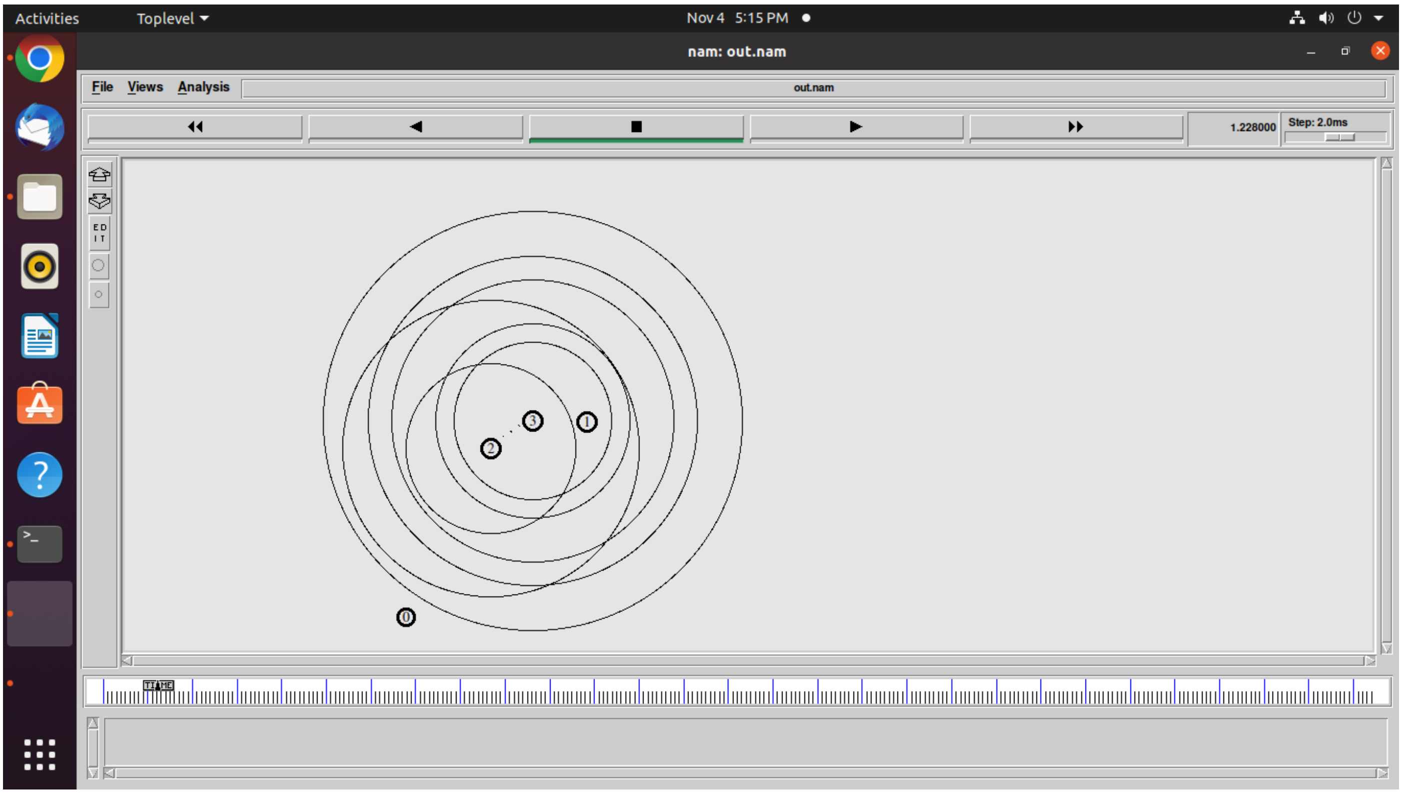
```
$n1 set X_ 210.0
$n1 set Y_ 230.0
$n1 set Z_ 0.0

$n2 set X_ 100.0
$n2 set Y_ 200.0
$n2 set Z_ 0.0

$n3 set X_ 150.0
$n3 set Y_ 230.0
$n3 set Z_ 0.0

$ns at 1.0 "$n1 setdest 490.0 340.0 25.0"
$ns at 1.0 "$n2 setdest 300.0 130.0 5.0"
$ns at 1.0 "$n3 setdest 190.0 440.0 15.0"

#===================================
#       Agents Definition
#===================================
#===================================
#       Applications Definition
#===================================
#Setup a CBR Application over UDP connection
set tcp [new Agent/TCP]
set sink [new Agent/TCPSink]
$ns attach-agent $n0 $tcp
$ns attach-agent $n1 $sink
$ns connect $tcp $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp

$ns at 1.0 "$ftp start"
$ns at 2.5 "$ftp stop"

set udp [new Agent/UDP]
set null [new Agent/Null]
$ns attach-agent $n2 $udp
$ns attach-agent $n3 $null
$ns connect $udp $null
set cbr [new Application/Traffic/CBR]
$cbr attach-agent $udp
```

```
$ns at 1.0 "$cbr start"
$ns at 3.0 "$cbr stop"
$ns at 30.0 "finish"

#===================================
#       Termination
#===================================
#Define a 'finish' procedure
proc finish {} {
    global ns tracefile namfile
    $ns flush-trace
    close $tracefile
    close $namfile
    exec nam out.nam &
    exit 0
}
$ns run
```

Output



```
abhishek@abhishek-VirtualBox:~/Desktop/060010715 - Wireless Networks-20220822T130913Z-001/Experiment Submition/Experiment - 2$ ns p3.tcl
num_nodes is set 4
INITIALIZE THE LIST xListHead
Starting Simulation
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 550.0
SORTING LISTS ...DONE!
```



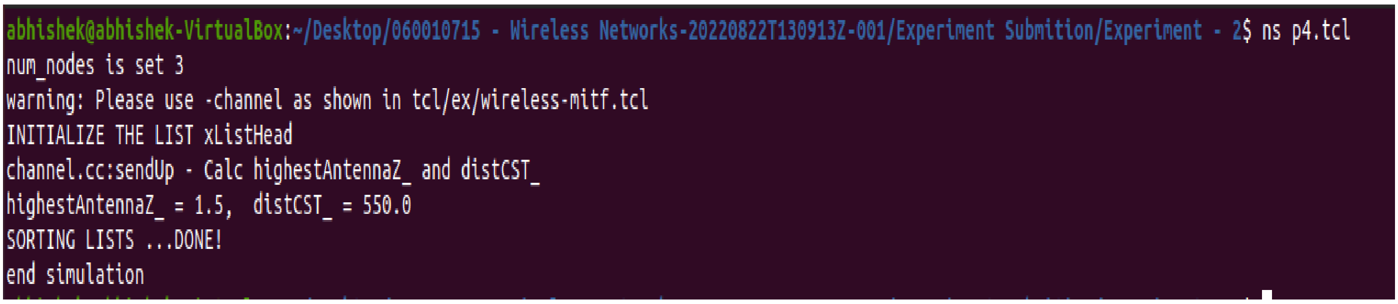
4. Create 3-Node Example for Ad-hoc Simulator with AODV.

p4.tcl

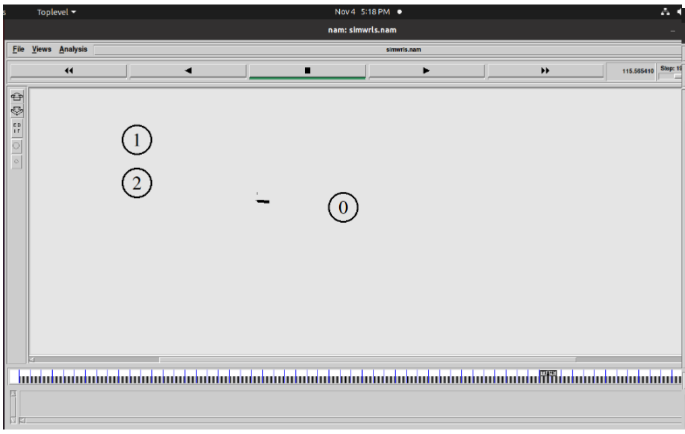
```
# P4 3-Node Example For ad-hoc simulator with AODV.

#Define options
set val(chan)   Channel/WirelessChannel    ;# channel type
set val(prop)   Propagation/TwoRayGround   ;# radio-propagation model
set val(netif)  Phy/WirelessPhy            ;# network interface type
set val(mac)    Mac/802_11                 ;# MAC type
set val(ifq)    Queue/DropTail/PriQueue    ;# interface queue type
set val(ll)     LL                         ;# link layer type
set val(ant)    Antenna/OmniAntenna        ;# antenna model
set val(ifqlen) 50                         ;# max packet in ifq
set val(nn)     3                          ;# number of mobilenodes
set val(rp)     AODV                       ;# routing protocol
set val(x)      500                        ;# X dimension of topography
set val(y)      400                        ;# Y dimension of topography
set val(stop)   150                        ;# time of simulation end

set ns   [new Simulator]
set tracefd      [open simple.tr w]
set windowVsTime2 [open win.tr w]
set namtrace [open simwrls.nam w]

$ns trace-all $tracefd
$ns namtrace-all-wireless $namtrace $val(x) $val(y)

#set up topography object
set topo       [new Topography]
$topo load_flatgrid $val(x) $val(y)

create-god $val(nn)

#configure the nodes
$ns node-config -adhocRouting  $val(rp) \
            -llType       $val(ll) \
            -macType      $val(mac) \
            -ifqType      $val(ifq) \
            -ifqLen       $val(ifqlen) \
            -antType      $val(ant) \
            -propType     $val(prop) \
            -phyType      $val(netif) \
        -channelType   $val(chan) \
        -topoInstance  $topo \
        -agentTrace    ON \
        -routerTrace   ON \
        -macTrace      OFF \
        -movementTrace ON

for {set i 0} {$i < $val(nn) } { incr i } {
        set node_($i) [$ns node]
}

# Provide initial location of mobilenodes
$node_(0) set x_ 5.0

$node_(0) set Y_ 5.0
$node_(0) set Z_ 0.0

$node_(1) set x_ 490.0
$node_(1) set Y_ 285.0
$node_(1) set Z_ 0.0

$node_(2) set x_ 150.0
$node_(2) set Y_ 240.0
$node_(2) set Z_ 0.0

#Generation of movements
$ns at 10.0 "$node_(0) setdest 250.0 250.0 3.0"
$ns at 15.0 "$node_(1) setdest 45.0 285.0 5.0"
$ns at 110.0 "$node_(0) setdest 480.0 300.0 5.0"

#set a TCP connection between node_(0) and node_(1)
set tcp [new Agent/TCP/Newreno]
$tcp set class_ 2
set sink [new Agent/TCPSink]
$ns attach-agent $node_(0) $tcp
$ns attach-agent $node_(1) $sink
$ns connect $tcp $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ns at 10.0 "$ftp start"

#Printin the window size
proc plotWindow {tcpSource file} {
```

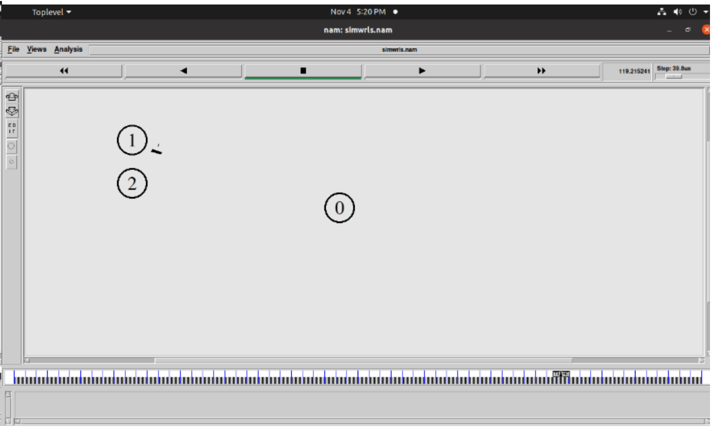
```
global ns
set time 0.01
set now [$ns now]
set cwnd [$tcpSource set cwnd_]
puts $file "$now $cwnd"
$ns at [expr $now+$time] "plotWindow $tcpSource $file" }
$ns at 10.1 "plotWindow $tcp $windowVsTime2"

#Define node initial position in nam
for {set i 0} {$i < $val(nn)} {incr i} {
#30 defines the node size for nam
$ns initial_node_pos $node_($i) 30
}

#Telling nodes when the simulator ends
for {set i 0} {$i < $val(nn) } {incr i} {
        $ns at $val(stop) "$node_($i) reset";
}

#ending nam and the simulation
$ns at $val(stop) "$ns nam-end-wireless $val(stop)"
$ns at $val(stop) "stop"
$ns at 150.01 "puts \"end simulation\"; $ns halt"

proc stop {} {
        global ns tracefd namtrace
        $ns flush-trace
        close $tracefd
        close $namtrace
}

$ns run
```

Output



```
abhishek@abhishek-VirtualBox:~/Desktop/060010715 - Wireless Networks-20220822T130913Z-001/Experiment Submition/Experiment - 2$ ns p4.tcl
num_nodes is set 3
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 550.0
SORTING LISTS ...DONE!
end simulation
```





5. Create 3-Node Example for Ad-hoc Simulator with DSDV.

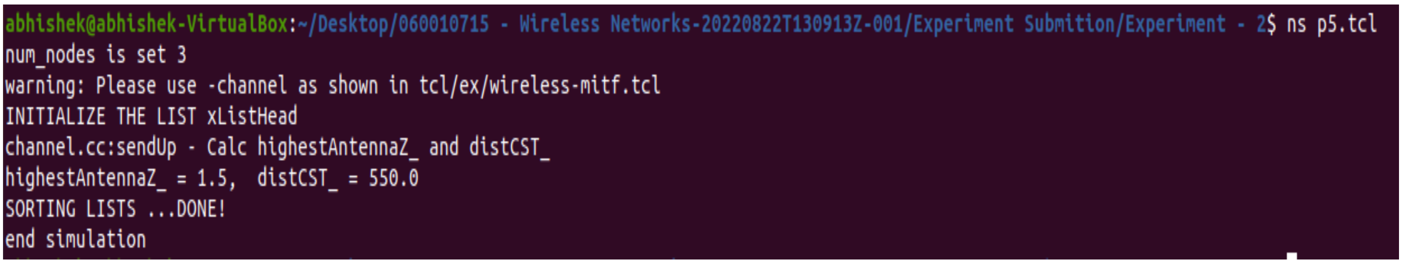
p5.tcl

```
# P5 3-Node Example for Ad-hoc Simulator with DSDV
# Define options
set val(chan)   Channel/WirelessChannel    ;# channel type
set val(prop)   Propagation/TwoRayGround   ;# radio-propagation model
set val(netif)  Phy/WirelessPhy           ;# network interface type
set val(mac)    Mac/802_11                ;# MAC type
set val(ifq)    Queue/DropTail/PriQueue    ;# interface queue type
set val(ll)     LL                        ;# link layer type
set val(ant)    Antenna/OmniAntenna        ;# antenna model
set val(ifqlen) 50                        ;# max packet in ifq
set val(nn)     3                         ;# number of mobilenodes
set val(rp)     DSDV                      ;# routing protocol
set val(x)      500                       ;# X dimension of topography
set val(y)      400                       ;# Y dimension of topography
set val(stop)   150                       ;# time of simulation end

set ns   [new Simulator]
set tracefd       [open simple.tr w]
set windowVsTime2 [open win.tr w]
set namtrace [open simwrls.nam w]

$ns trace-all $tracefd
$ns namtrace-all-wireless $namtrace $val(x) $val(y)

#set up topography object
set topo       [new Topography]
$topo load_flatgrid $val(x) $val(y)
```

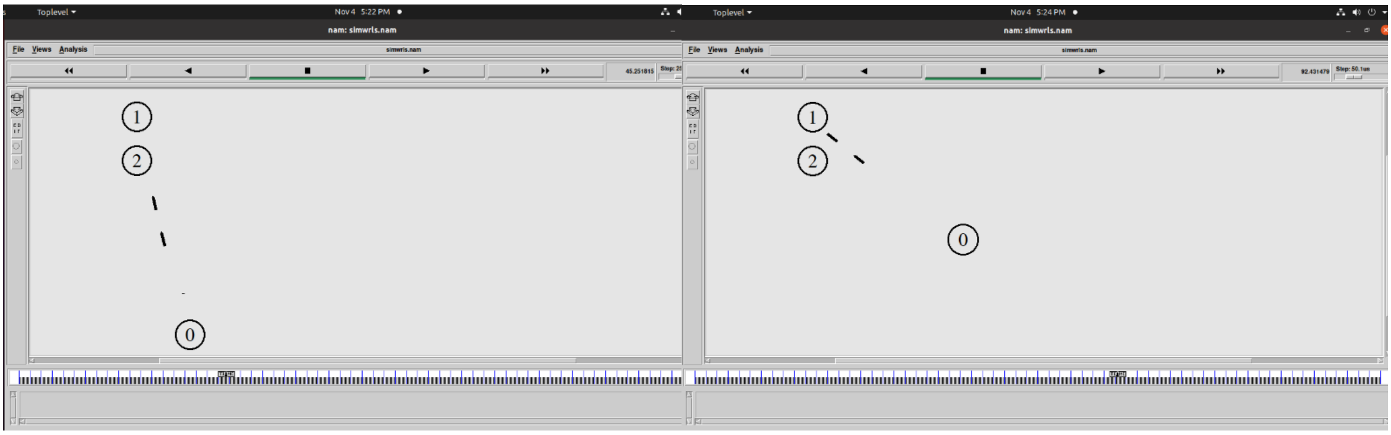
```
create-god $val(nn)

# configure the nodes
$ns node-config -adhocRouting  $val(rp) \
            -llType       $val(ll) \
            -macType      $val(mac) \
            -ifqType      $val(ifq) \
            -ifqLen       $val(ifqlen) \
            -antType      $val(ant) \
            -propType     $val(prop) \
            -phyType      $val(netif) \
            -channelType  $val(chan) \
            -topoInstance $topo \
            -agentTrace   ON \
            -routerTrace  ON \
            -macTrace     OFF \
            -movementTrace ON

for {set i 0} {$i < $val(nn) } { incr i } {
          set node_($i) [$ns node]
}

# Provide initial location of mobilenodes
$node_(0) set x_ 5.0
$node_(0) set Y_ 5.0
$node_(0) set Z_ 0.0

$node_(1) set x_ 490.0
$node_(1) set Y_ 285.0
$node_(1) set Z_ 0.0

$node_(2) set x_ 150.0
$node_(2) set Y_ 240.0
$node_(2) set Z_ 0.0

#Generation of movements
$ns at 10.0 "$node_(0) setdest 250.0 250.0 3.0"
$ns at 15.0 "$node_(1) setdest 45.0 285.0 5.0"
$ns at 110.0 "$node_(0) setdest 480.0 300.0 5.0"

#set a TCP connection between node_(0) and node_(1)
set tcp [new Agent/TCP/Newreno]
$tcp set class_ 2
```

```
set sink [new Agent/TCPSink]
$ns attach-agent $node_(0) $tcp
$ns attach-agent $node_(1) $sink
$ns connect $tcp $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ns at 10.0 "$ftp start"

#Printin the window size
proc plotWindow {tcpSource file} {
global ns
set time 0.01
set now [$ns now]
set cwnd [$tcpSource set cwnd_]
puts $file "$now $cwnd"
$ns at [expr $now+$time] "plotWindow $tcpSource $file" }
$ns at 10.1 "plotWindow $tcp $windowVsTime2"

#Define node initial position in nam
for {set i 0} {$i < $val(nn)} {incr i} {
#30 defines the node size for nam
$ns initial_node_pos $node_($i) 30
}

#Telling nodes when the simulator ends
for {set i 0} {$i < $val(nn) } {incr i} {
        $ns at $val(stop) "$node_($i) reset";
}

#ending nam and the simulation
$ns at $val(stop) "$ns nam-end-wireless $val(stop)"
$ns at $val(stop) "stop"
$ns at 150.01 "puts \"end simulation\"; $ns halt"

proc stop {} {
        global ns tracefd namtrace
        $ns flush-trace
        close $tracefd
        close $namtrace
}

$ns run
```

Output



```
abhishek@abhishek-VirtualBox:~/Desktop/060010715 - Wireless Networks-20220822T130913Z-001/Experiment Submition/Experiment - 2$ ns p5.tcl
num_nodes is set 3
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 550.0
SORTING LISTS ...DONE!
end simulation
```


Toplevel
Nov 4 5:22 PM
nam: simwrls.nam
File Views Analysis
simwrls.nam
45.251815
Step: 2
1
2
0
Toplevel
Nov 4 5:24 PM
nam: simwrls.nam
File Views Analysis
simwrls.nam
92.431479
Step: 50.1us
1
2
0