/\*\*\* Assignment NS2. \*\*\*/

/\*\*\*\*\*Part A:---- tcp\_udp\_ftp

#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 2Mb 10ms DropTail

$ns duplex-link $n1 $n2 2Mb 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\_ 1

$ns attach-agent $n0 $tcp

set sink [new Agent/TCPSink]

$ns attach-agent $n3 $sink

$ns connect $tcp $sink

#Setup a FTP over TCP connection

set ftp [new Application/FTP]

$ftp attach-agent $tcp

#Setup a UDP connection

set udp [new Agent/UDP]

$udp set class\_ 2

$ns attach-agent $n1 $udp

set null [new Agent/Null]

$ns attach-agent $n3 $null

$ns connect $udp $null

#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

#Schedule events for the CBR and FTP agents

$ns at 0.1 "$cbr start"

$ns at 1.0 "$ftp start"

$ns at 4.0 "$ftp stop"

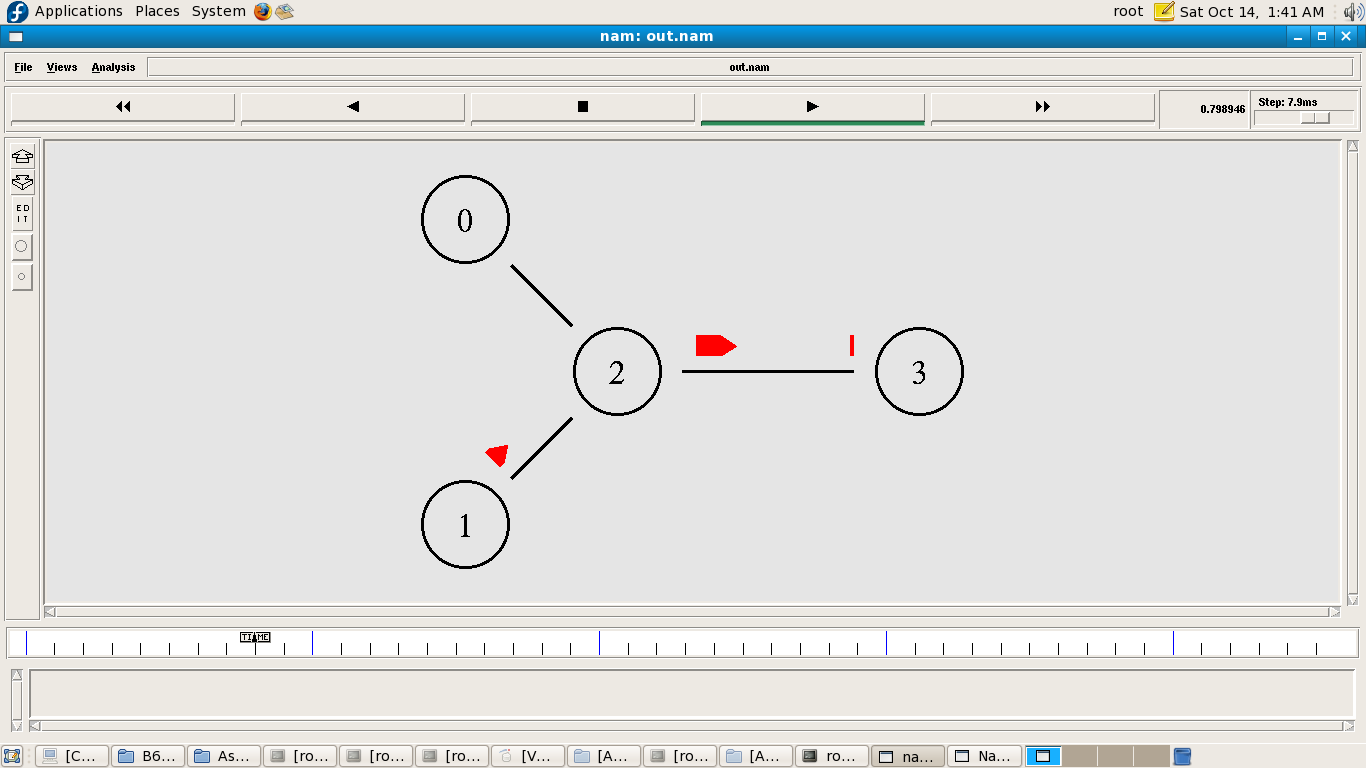
$ns at 4.5 "$cbr stop"

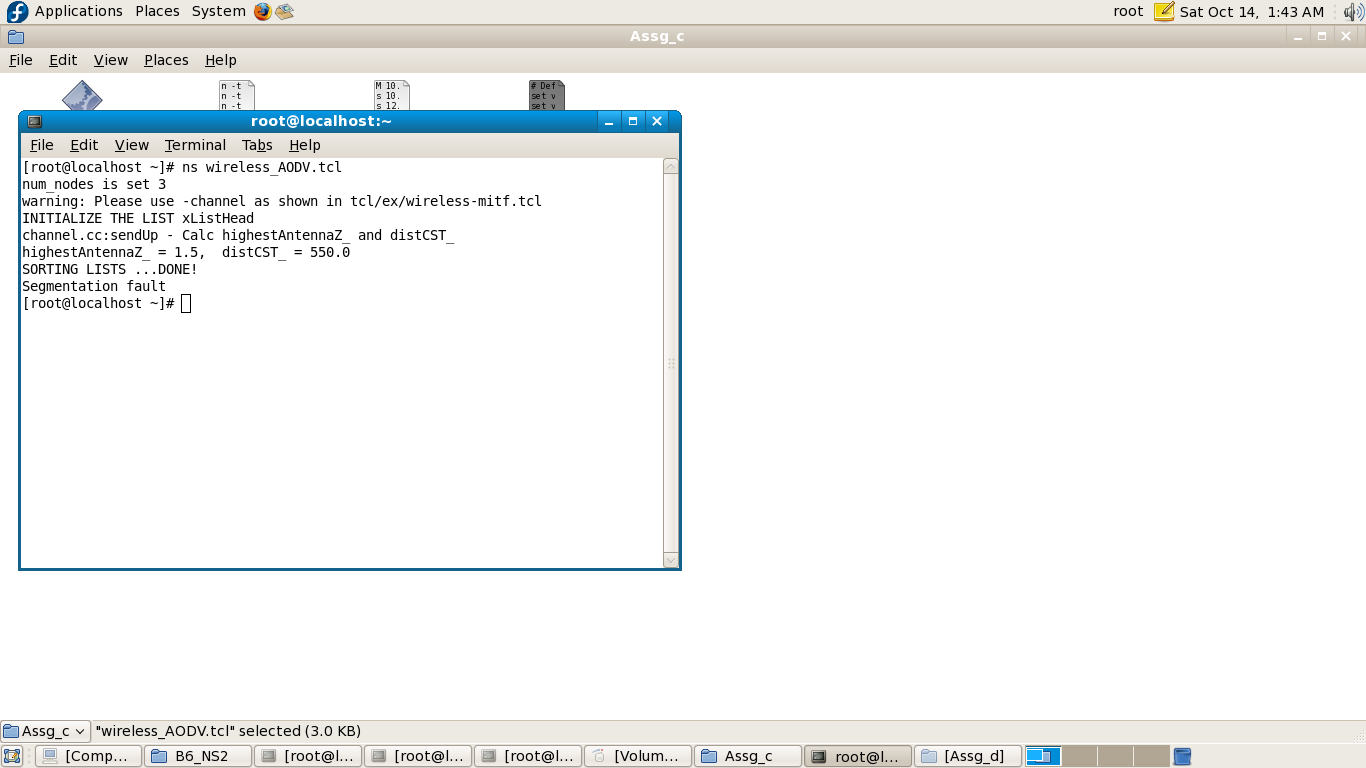
#Call the finish procedure after 5 seconds of simulation time

$ns at 5.0 "finish"

#Run the simulation

$ns run





/\*\*\* Assignment NS2. \*\*\*/

/\*\*\*\*\*Part C:---- wireless\_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) 110 ;# time of simulation end

set ns [new Simulator]

set tracefd [open wireless.tr w]

set namtrace [open wireless.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)

#

# Create nn mobilenodes [$val(nn)] and attach them to the channel.

#

# 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 OFF \

-routerTrace OFF \

-macTrace ON \

-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 tcp [new Agent/TCP]

$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"

# 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) 50

}

# Telling nodes when the simulation 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 110.01 "puts \"end simulation\" ; $ns halt"

proc stop {} {

global ns tracefd namtrace

$ns flush-trace

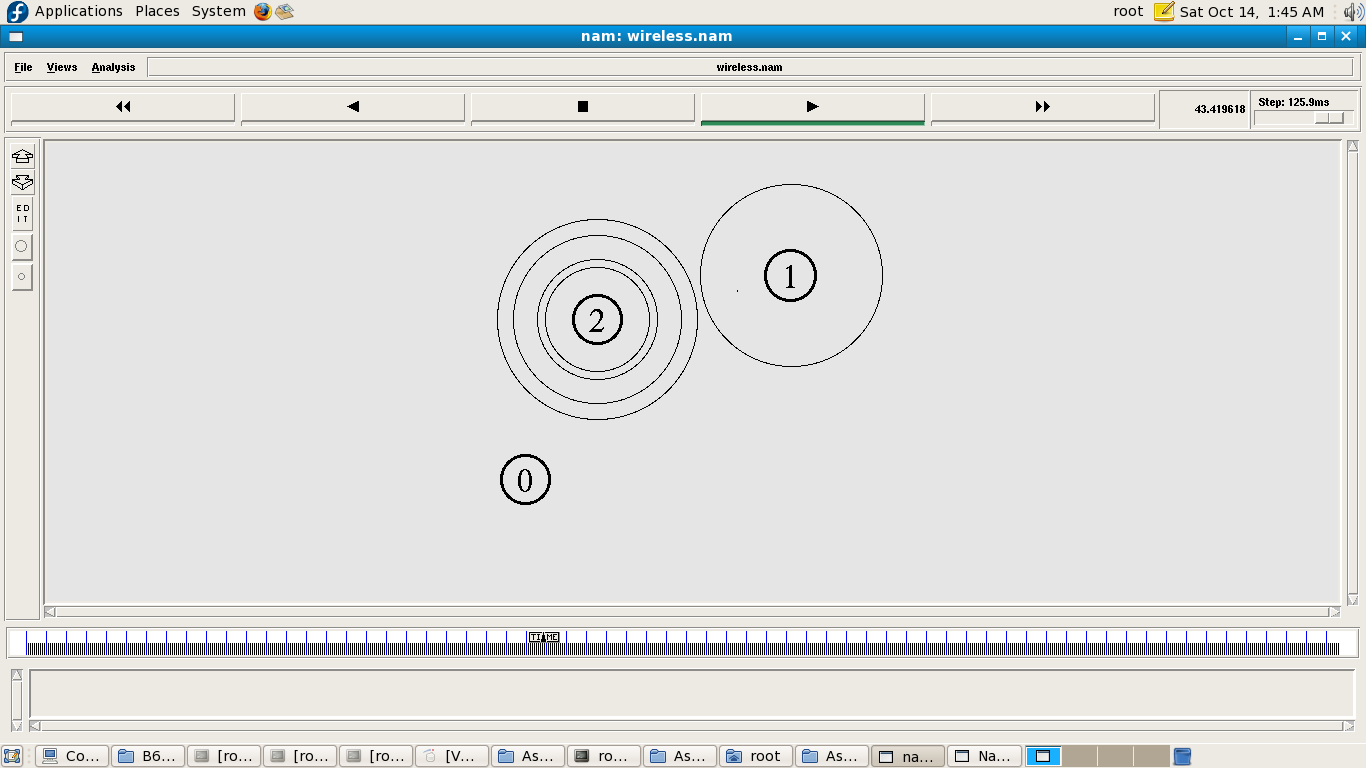
close $tracefd

close $namtrace

exec nam wireless.nam &

}

$ns run



/\*\*\* Assignment NS2. \*\*\*/

/\*\*\*\*\*Part D:---- wired

set ns [new Simulator]

#Define different colors for data flows (for NAM)

$ns color 1 Blue

$ns color 2 Red

#Open the Trace files

set file1 [open out.tr w]

$ns trace-all $file1

#Open the NAM trace file

set file2 [open out.nam w]

$ns namtrace-all $file2

#Define a 'finish' procedure

proc finish {} {

global ns file1 file2

$ns flush-trace

close $file1

close $file2

exec nam out.nam &

exit 0

}

#Create six 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]

$ns at 0.1 "$n1 label \"CBR\""

$ns at 1.0 "$n0 label \"FTP\""

#Create links between the nodes

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

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

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

$ns duplex-link $n3 $n4 0.5Mb 40ms DropTail

$ns duplex-link $n3 $n5 0.5Mb 30ms DropTail

#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 $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

#Set Queue Size of link (n2-n3) to 10

$ns queue-limit $n2 $n3 40

#Setup a 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 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

#Setup a UDP connection

set udp [new Agent/UDP]

$ns attach-agent $n1 $udp

set null [new Agent/Null]

$ns attach-agent $n5 $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\_ 0.01mb

$cbr set random\_ false

$ns at 0.1 "$cbr start"

$ns at 1.0 "$ftp start"

$ns at 124.0 "$ftp stop"

$ns at 624.5 "$cbr stop"

# Trace Congestion Window and RTT

set file [open cwnd\_rtt.tr w]

$tcp attach $file

$tcp trace cwnd\_

$tcp trace rtt\_

$ns at 625.0 "finish"

$ns run

