

1

```
# This script is created by NSG2 beta1
# <http://wushoupong.googlepages.com/nsg>

#=====
#      Simulation parameters setup
#=====
set val(stop)    4.9                                ;# time of simulation end

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

#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

#=====
#      Nodes Definition
#=====
#Create 3 nodes
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]

#=====
#      Links Definition
#=====
#Createlinks between nodes
$ns duplex-link $n0 $n1 100.0Mb 10ms DropTail
$ns queue-limit $n0 $n1 50
$ns duplex-link $n1 $n2 0.25Mb 10ms DropTail
$ns queue-limit $n1 $n2 5

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

```

#=====
#           Agents Definition
#=====
#Setup a TCP connection
set tcp0 [new Agent/TCP]
$ns attach-agent $n0 $tcp0
set sink1 [new Agent/TCPSink]
$ns attach-agent $n2 $sink1
$ns connect $tcp0 $sink1
$tcp0 set packetSize_ 1500

#=====
#           Applications Definition
#=====
#Setup a FTP Application over TCP connection
set ftp0 [new Application/FTP]
$ftp0 attach-agent $tcp0
$ns at 0.1 "$ftp0 start"
$ns at 5.0 "$ftp0 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
}
$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

```

2

```

# This script is created by NSG2 beta1
# <http://wushoupong.googlepages.com/nsg>

```

```

#=====
#      Simulation parameters setup
#=====
set val(stop) 7.0 ;# time of simulation end

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

#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

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

#=====
#      Links Definition
#=====
#Createlinks between nodes
$ns duplex-link $n0 $n1 100.0Mb 10ms DropTail
$ns queue-limit $n0 $n1 50
$ns duplex-link $n2 $n1 100.0Mb 10ms DropTail
$ns queue-limit $n2 $n1 50
$ns duplex-link $n1 $n3 10.0Mb 1ms DropTail
$ns queue-limit $n1 $n3 5

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

#=====
#      Agents Definition

```

```

#=====
#Setup a TCP connection
set tcp0 [new Agent/TCP]
$ns attach-agent $n0 $tcp0
set sink2 [new Agent/TCPSink]
$ns attach-agent $n3 $sink2
$ns connect $tcp0 $sink2
$tcp0 set packetSize_ 1500

#Setup a UDP connection
set udp3 [new Agent/UDP]
$ns attach-agent $n2 $udp3
set null4 [new Agent/Null]
$ns attach-agent $n3 $null4
$ns connect $udp3 $null4
$udp3 set packetSize_ 1500

#=====
#           Applications Definition
#=====
#Setup a FTP Application over TCP connection
set ftp0 [new Application/FTP]
$ftp0 attach-agent $tcp0
$ns at 0.1 "$ftp0 start"
$ns at 5.0 "$ftp0 stop"

#Setup a CBR Application over UDP connection
set cbr1 [new Application/Traffic/CBR]
$cbr1 attach-agent $udp3
$cbr1 set packetSize_ 1000
$cbr1 set rate_ 1.0Mb
$cbr1 set random_ null
$ns at 1.0 "$cbr1 start"
$ns at -7.0 "$cbr1 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
}
$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

```

3

```

# This script is created by NSG2 beta1
# <http://wushoupong.googlepages.com/nsg>

#=====
#      Simulation parameters setup
#=====
set val(stop)    10.0                                ;# time of simulation end

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

#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

#=====
#      Nodes Definition
#=====
#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]

#=====

```

```

#           Links Definition
#=====
#Createlinks between nodes
$ns duplex-link $n0 $n2 1.0Mb 10ms DropTail
$ns duplex-link $n1 $n2 1.0Mb 10ms DropTail
$ns duplex-link $n2 $n3 0.5Mb 10ms DropTail
$ns duplex-link $n3 $n4 2.0Mb 10ms DropTail
$ns duplex-link $n3 $n5 2.0Mb 10ms DropTail

#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
$ns duplex-link-op $n3 $n4 orient right-up
$ns duplex-link-op $n3 $n5 orient right-down

#=====
#           Agents Definition
#=====
set ping0 [new Agent/Ping]
$ns attach-agent $n0 $ping0

set ping1 [new Agent/Ping]
$ns attach-agent $n1 $ping1

set ping2 [new Agent/Ping]
$ns attach-agent $n2 $ping2

set ping3 [new Agent/Ping]
$ns attach-agent $n3 $ping3

set ping4 [new Agent/Ping]
$ns attach-agent $n4 $ping4

set ping5 [new Agent/Ping]
$ns attach-agent $n5 $ping5

$ns connect $ping0 $ping4
$ns connect $ping1 $ping5
#=====
#           Applications Definition
#=====

#=====
#           Termination

```

```

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

proc sendPingPacket {} {
    global ns ping0 ping1
    set intervalTime 0.001
    set now [$ns now]
    $ns at [expr $now + $intervalTime] "$ping0 send"
    $ns at [expr $now + $intervalTime] "$ping1 send"
    $ns at [expr $now + $intervalTime] "sendPingPacket"
}

Agent/Ping instproc recv {from rtt} {
    global seq
    $self instvar node_
    puts " [$node_ id] and $from and rtt is $rtt";
}

$ns at 0.01 "sendPingPacket"
$ns at 10 "finish"
$ns run

```

4.tcl

```

# This script is created by NSG2 beta1
# <http://wushoupong.googlepages.com/nsq>

#=====
#      Simulation parameters setup
#=====
set val(stop)    10.0                                ;# time of simulation end

#=====
#      Initialization
#=====
#Create a ns simulator

```

```

set ns [new Simulator]

#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

#=====
#           Nodes Definition
#=====
#Create 10 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 [$ns node]
set n7 [$ns node]
set n8 [$ns node]
set n9 [$ns node]

$ns make-lan "$n0 $n1 $n2 $n3 $n4" 1Mb 10ms LL Queue/DropTail Mac/802_3
$ns make-lan "$n5 $n6 $n7 $n8 $n9" 1Mb 10ms LL Queue/DropTail Mac/802_3

#=====
#           Links Definition
#=====
#Createlinks between nodes
$ns duplex-link $n2 $n6 2.5Mb 10ms DropTail
$ns queue-limit $n2 $n6 50

#Give node position (for NAM)
$ns duplex-link-op $n6 $n2 orient left-down

#=====
#           Agents Definition
#=====
#Setup a TCP connection
set tcp0 [new Agent/TCP]
$ns attach-agent $n0 $tcp0
set sink2 [new Agent/TCPSink]
$ns attach-agent $n3 $sink2

```



```

$ns connect $tcp0 $sink2
$tcp0 set packetSize_ 1500

#Setup a UDP connection
set udp3 [new Agent/UDP]
$ns attach-agent $n2 $udp3
set null4 [new Agent/Null]
$ns attach-agent $n5 $null4
$ns connect $udp3 $null4
$udp3 set packetSize_ 1500

#=====
#           Applications Definition
#=====
#Setup a FTP Application over TCP connection
set telnet [new Application/Telnet]
$telnet attach-agent $tcp0
$ns at 0.5 "$telnet start"
$ns at 9.0 "$telnet stop"

#Setup a CBR Application over UDP connection
set cbr1 [new Application/Traffic/CBR]
$cbr1 attach-agent $udp3
$cbr1 set packetSize_ 500
$cbr1 set rate_ 1.0Mb
$cbr1 set random_ null
$ns at 0.1 "$cbr1 start"
$ns at 9.0 "$cbr1 stop"

$tcp0 set class_ 1
$udp3 set class_ 2

set err [new ErrorModel]
$ns lossmodel $err $n2 $n6
$err set rate_ 0.01
$ns set dataRate_ 10Mb

#=====
#           Termination
#=====
#Define a 'finish' procedure
proc finish {} {
    global ns tracefile namfile
    $ns flush-trace
    close $tracefile

```

```

        close $namfile
        exec nam out.nam &
        exec awk -f 4.awk out.tr &
        exit 0
    }

    $ns at 10.0 "finish"
    $ns run

```

4.awk

```

#!/usr/bin/awk -f
BEGIN{
    cbr=0;
    total=0;
    ftp=0;
    thr=0;
}
{
    if(($1=="+" )&&($3=="2" )&&($4=="6" )&&($5=="cbr" )){
        cbr++;
    }
    if(($1=="+" )&&($3=="2" )&&($4=="6" )&&($5=="tcp" )){
        ftp++;
    }
    total=cbr+ftp;
}
END{
    thr=((total*500*8)/(8*1000000));
    print"The Throughput is %d\n",thr;
}

```

5

```

#!/usr/bin/awk -f
BEGIN{
    cbr=0;
    total=0;
    ftp=0;
    thr=0;
}

```

```

{
    if(($1=="+" )&&($3=="2" )&&($4=="6" )&&($5=="cbr" )){
        cbr++;
    }
    if(($1=="+" )&&($3=="2" )&&($4=="6" )&&($5=="tcp" )){
        ftp++;
    }
    total=cbr+ftp;
}
END{
    thr=((total*500*8)/(8*1000000));
    print"The Throughput is %d\n",thr;
}

```

6

```

#Define the simulation options
set val(chan) Channel/WirelessChannel

set val(prop) Propagation/TwoRayGround

set val(ant) Antenna/OmniAntenna

set val(ll) LL

set val(ifq) Queue/DropTail/PriQueue

set val(ifqlen) 50

set val(netif) Phy/WirelessPhy

set val(mac) Mac/802_11

set val(rp) AODV

set val(nn) [lindex $argv 0]

set opt(x) 750

set opt(y) 750

set val(stop) 100

```

```

set ns [new Simulator]
set trfd [open Wireless.tr w]
set namfd [open Wireless.nam w]
$ns trace-all $trfd
$ns namtrace-all-wireless $namfd $opt(x) $opt(y)
#Create a Topography Object
set topo [new Topography]
$topo load_flatgrid $opt(x) $opt(y)
set god_ [create-god $val(nn)]
#Configure the nodes
$ns node-config -adhocRouting $val(rp) \
-llType $val(ll) \
-macType $val(mac) \
-ifqType $val(ifq) \
-channelType $val(chan) \
-propType $val(prop) \
-antType $val(ant) \
-ifqLen $val(ifqlen) \
-phyType $val(netif) \
-topoInstance $topo \
-agentTrace ON \
-routerTrace ON \
-macTrace OFF \
-movementTrace OFF
for {set i 0} {$i < $val(nn)} {incr i} {
set n($i) [$ns node]
}
#Randomly placing the nodes
for {set i 0} {$i < $val(nn)} {incr i} {
set XX [expr rand()*750]
set YY [expr rand()*750]
$n($i) set X_ $XX
$n($i) set Y_ $YY
}
$ns at 0.0 "destination"
for {set i 0} {$i < $val(nn)} {incr i} {
$ns initial_node_pos $n($i) 30
}
proc destination {} {
global ns val n
set now [$ns now]
set time 5.0
for {set i 0} {$i < $val(nn)} {incr i} {
set XX [expr rand()*750]
set YY [expr rand()*750]
$ns at [expr $now + $time] "$n($i) setdest $XX $YY 20.0"
}

```

```

}
$ns at [expr $now + $time] "destination"
}
set tcp1 [new Agent/TCP]
$ns attach-agent $n(1) $tcp1
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
set sink1 [new Agent/TCPSink]
$ns attach-agent $n(3) $sink1
$ns connect $tcp1 $sink1
$ns at 5.0 "$ftp1 start"

$ns at $val(stop) "$ns nam-end-wireless $val(stop)"
$ns at $val(stop) "stop"
proc stop {} {
global ns trfd namfd
close $trfd
close $namfd
exec nam Wireless.nam &
exec awk -f 6a.awk Wireless.tr &
exit 0
}
$ns run

```

6.awk

```

BEGIN {
    PacketRcvd=0;
    Throughput=0.0;

}
{
    if(($1=="r")&&($3=="_3_")&&($4=="AGT")&&($7=="tcp")&&($8>35))
    {
        PacketRcvd++;

    }
}
END {
    Throughput=((PacketRcvd*1000*8)/(95.0*1000000));
    printf "The Throughput is :%f\n",Throughput;
}

```