

**Name - Abinash Gupta**  
**Roll No. - 120CS0157**  
**LAB 9**

**TCL Script**

```
# A 100-node example for ad-hoc simulation 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) 15 ;# max packet in ifq
set val(nn) 12 ;# number of
mobilenodes
set val(rp) AODV ;# routing protocol
set val(x) 800 ;# X dimension of
topography
set val(y) 800 ;# Y dimension of
topography
set val(stop) 100 ;# time of
simulation end

set ns [new Simulator]
set tracefd [open testAODV.tr w]
set windowVsTime2 [open win.tr w]
set namtrace [open my.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 ON \
        -routerTrace ON \
        -macTrace OFF \
        -movementTrace ON

    for {set i 0} {$i < $val(nn)} {incr i} {
        set node_($i) [$ns node]
        $node_($i) set X_ [expr 70 * $i]
        $node_($i) set Y_ [expr 70 * $i]
        $node_($i) set Z_ 0.0
    }

#   for {set i 0} {$i < $val(nn)} {incr i} {
#       $ns at [expr 15+round(rand()*60)] "$node_($i) setdest [expr
# 10+round(rand()*480)] [expr 10+round(rand()*380)] [expr
# 2+round(rand()*15)]"

#   }

# 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"
# $ns at 70.0 "$node_(2) setdest 480.0 300.0 5.0"
# $ns at 20.0 "$node_(3) setdest 200.0 200.0 5.0"
# $ns at 25.0 "$node_(4) setdest 50.0 50.0 10.0"

```

```
# $ns at 60.0 "$node_(5) setdest 150.0 70.0 2.0"
# $ns at 90.0 "$node_(6) setdest 380.0 150.0 8.0"
# $ns at 42.0 "$node_(7) setdest 200.0 100.0 15.0"
# $ns at 55.0 "$node_(8) setdest 50.0 275.0 5.0"
# $ns at 19.0 "$node_(9) setdest 250.0 250.0 7.0"
# $ns at 90.0 "$node_(10) setdest 150.0 150.0 20.0"
```

```
# Set a TCP connection between node_(2) and node_(8)
```

```
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_(11) $sink
$ns connect $tcp $sink
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ns at 0.1 "$ftp start"
```

```
set tcp1 [new Agent/TCP/Newreno]
$tcp1 set class_ 2
set sink1 [new Agent/TCPSink]
$ns attach-agent $node_(4) $tcp1
$ns attach-agent $node_(0) $sink1
$ns connect $tcp1 $sink1
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns at 0.75 "$ftp1 start"
```

```
set tcp2 [new Agent/TCP]
$tcp2 set class_ 3
set sink2 [new Agent/TCPSink]
$ns attach-agent $node_(4) $tcp2
$ns attach-agent $node_(11) $sink2
$ns connect $tcp2 $sink2
set cbr0 [new Application/Traffic/CBR]
$cbr0 attach-agent $tcp2
$ns at 0.5 "$cbr0 start"
```

```
# Printing 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 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 100 "puts \"end simulation\" ; $ns halt"
proc stop {} {
    global ns tracefd namtrace
    $ns flush-trace
    exec nam my.nam &
    close $tracefd
    close $namtrace
}

$ns run

```

## awk file

```

BEGIN {
    dropped=0;
    sentFid_1=0;
    sentFid_2=0;
}

{
    if ($1=="s"&&$4=="AGT")

```

```

{
    if($3=="_0_"){
        sentFid_1=sentFid_1+1;
    }
    if($3=="_4_"){
        sentFid_2=sentFid_2+1;
    }sent=sent+1;
}
if ($1 == "D") {
    dropped=dropped+1;
}

}

END {
    printf("Number of dropped packets is %d\n",dropped);
    printf("Number of packets sent in FID_1 = %d\n",sentFid_1);
    printf("Number of packets sent in FID_2 = %d\n",sentFid_2);
}

```

## OUTPUT

```

● avi@abinash-gupta:~/Documents/CN/LAB9$ awk -f awkfile.awk testA0DV.tr
Number of dropped packets is 374
Number of packets sent in FID_1 = 2616
Number of packets sent in FID_2 = 2827

```