```
# wireless1.tcl
# A simple example for wireless simulation

# ========================================================================
# Define options
# ========================================================================

set val(chan)       Channel/WirelessChannel
set val(prop)       Propagation/TwoRayGround
set val(netif)      Phy/WirelessPhy
set val(mac)        Mac/802_11
set val(ifq)        Queue/DropTail/PriQueue
set val(ll)         LL
set val(ant)        Antenna/OmniAntenna
set val(x)              670   ;# X dimension of the topography
set val(y)              670   ;# Y dimension of the topography
set val(ifqlen)         50            ;# max packet in ifq
set val(seed)           0.0
set val(adhocRouting)   DSR
set val(nn)             3             ;# how many nodes are simulated
set val(cp)             "/home/ssn/ns-allinone-2.35/ns-2.35/tcl/mobility/scene/cbr-3-test"
set val(sc)             "/home/ssn/ns-allinone-2.35/ns-2.35/tcl/mobility/scene/scen-3-test"
set val(stop)           400.0           ;# simulation time

# ========================================================================
# Main Program
# ========================================================================

#
# Initialize Global Variables
#

# create simulator instance

set ns_         [new Simulator]

# setup topography object

set topo        [new Topography]

# create trace object for ns and nam

set tracefd     [open wl.tr w]
set namtrace    [open wl.nam w]

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

# define topology
$topo load_flatgrid $val(x) $val(y)

#
# Create God
#
set god_ [create-god $val(nn)]

#
# define how node should be created
#

#global node setting

$ns_ node-config -adhocRouting $val(adhocRouting) \
                 -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 OFF \
                -macTrace OFF

#
#  Create the specified number of nodes [$val(nn)] and "attach" them
#  to the channel.

for {set i 0} {$i < $val(nn) } {incr i} {
        set node_($i) [$ns_ node]
        $node_($i) random-motion 0              ;# disable random motion
}


#
# Define node movement model
#
puts "Loading connection pattern..."
source $val(cp)

#
# Define traffic model
#
puts "Loading scenario file..."
source $val(sc)

# Define node initial position in nam

for {set i 0} {$i < $val(nn)} {incr i} {

    # 20 defines the node size in nam, must adjust it according to your scenario
    # The function must be called after mobility model is defined

    $ns_ initial_node_pos $node_($i) 20
}


#
# Tell nodes when the simulation ends
#
for {set i 0} {$i < $val(nn) } {incr i} {
    $ns_ at $val(stop).0 "$node_($i) reset";
}

$ns_ at  $val(stop).0002 "puts \"NS EXITING...\" ; $ns_ halt"

puts $tracefd "M 0.0 nn $val(nn) x $val(x) y $val(y) rp $val(adhocRouting)"
puts $tracefd "M 0.0 sc $val(sc) cp $val(cp) seed $val(seed)"
puts $tracefd "M 0.0 prop $val(prop) ant $val(ant)"

puts "Starting Simulation..."
$ns_ run
```