|  |
| --- |
| # Define options |
|  | set opt(chan) Channel/WirelessChannel ;# channel type |
|  | set opt(prop) Propagation/TwoRayGround ;# radio-propagation |
|  | model |
|  | set opt(netif) Phy/WirelessPhy ;# network interface type |
|  | set opt(mac) Mac/802\_11 ;# MAC type |
|  | set opt(ifq) Queue/DropTail/PriQueue ;# interface queue type |
|  | set opt(ll) LL ;# link layer type |
|  | set opt(ant) Antenna/OmniAntenna ;# antenna model |
|  | set opt(ifqlen) 1000 ;# max packet in ifq |
|  | set opt(nn) 4 ;# number of mobilenodes |
|  | set opt(adhocRouting) DSDV ;# routing protocol |
|  | set opt(x) 100 ;# x coordinate of topology |
|  | set opt(y) 100 ;# y coordinate of topology |
|  | set opt(stop) 10 ;# time to stop simulation |
|  | set num\_wired\_nodes 4 |
|  | set num\_bs\_nodes 2 ;# this is not really used here. |
|  | set size 500 |
|  | # create simulator instance |
|  | set ns\_ [new Simulator] |
|  | # set up for hierarchical routing |
|  | $ns\_ node-config -addressType hierarchical |
|  | AddrParams set domain\_num\_ 3 ;# number of domains |
|  | lappend cluster\_num 1 1 1 ;# number of clusters in each domain |
|  | AddrParams set cluster\_num\_ $cluster\_num |
|  | lappend eilastlevel 7 3 3 ;# number of nodes in each cluster |
|  | AddrParams set nodes\_num\_ $eilastlevel ;# of each domain |
|  | set tracefd [open out.tr w] |
|  | set namtrace [open out.nam w] |
|  | $ns\_ trace-all $tracefd |
|  | $ns\_ namtrace-all $namtrace |
|  | $ns\_ namtrace-all- wireless $namtrace $opt(x) $opt(y) |
|  |  |
|  | # Create topography object |
|  | set topo [new Topography] |
|  | # define topology |
|  | $topo load\_flatgrid $opt(x) $opt(y) |
|  | # create God |
|  | create-god $opt(nn) |
|  | #-=- =-=- NAM=-=- =-=- -=- =-=- =-=- =-=- =-=- -- =-=- -- =-=- =-=- = |
|  | #set ns\_ [new Simulator] |
|  | set f0 [open wir7.tr w] |
|  | $ns\_ trace-all $f0 |
|  | set namtrace [open out.nam w] |
|  | $ns\_ namtrace-all- wireless $namtrace $opt(x) $opt(y) |
|  | #=-=- -=- =-=- =-=- =-=- =-=- =-=- =-=- =-=- =-=- =-=- =-=- =-=- =-=- |
|  | #create wired nodes |
|  | set W1 [$ns\_ node 0.0.0] |
|  | set W2 [$ns\_ node 0.0.1] |
|  | set W3 [$ns\_ node 0.0.2] |
|  | set W4 [$ns\_ node 0.0.3] |
|  | set W5 [$ns\_ node 0.0.4] |
|  | set W8 [$ns\_ node 0.0.5] |
|  | set W9 [$ns\_ node 0.0.6] |
|  | # Configure for Basestation Node |
|  | $ns\_ node-config -adhocRouting $opt(adhocRouting) \ |
|  | -llType $opt(ll) \ |
|  | -macType $opt(mac) \ |
|  | -ifqType $opt(ifq) \ |
|  | -ifqLen $opt(ifqlen) \ |
|  | -antType $opt(ant) \ |
|  | -propType $opt(prop) \ |
|  | -phyType $opt(netif) \ |
|  | -channelType $opt(chan) \ |
|  | -topoInstance $topo \ |
|  | -wiredRouting ON \ |
|  | -agentTrace ON \ |
|  | -routerTrace ON \ |
|  | -macTrace OFF |
|  |  |
|  | # Position (fixed) for base-station nodes (HA &amp; FA). |
|  | set BS1 [$ns\_ node 1.0.0] |
|  | set BS2 [$ns\_ node 2.0.0] |
|  |  |
|  | # create a mobilenode that would be moving between HA and FA. |
|  | # note address of MH indicates its in the same domain as HA. |
|  | $ns\_ node-config -wiredRouting OFF |
|  | set R1 [$ns\_ node 1.0.2] |
|  | set R2 [$ns\_ node 1.0.3] |
|  | set R3 [$ns\_ node 2.0.2] |
|  | set R4 [$ns\_ node 2.0.3] |
|  |  |
|  | $R1 base-station [AddrParams addr2id [$BS1 node-addr]] |
|  | $R2 base-station [AddrParams addr2id [$BS1 node-addr]] |
|  | $R3 base-station [AddrParams addr2id [$BS2 node-addr]] |
|  | $R4 base-station [AddrParams addr2id [$BS2 node-addr]] |
|  | # position of the nodes |
|  | $R1 set X\_ 120.000000000000 |
|  | $R1 set Y\_ 80.000000000000 |
|  | $R1 set Z\_ 0.000000000000 |
|  | $R2 set X\_ 160.000000000000 |
|  | $R2 set Y\_ 40.000000000000 |
|  | $R2 set Z\_ 0.000000000000 |
|  | $R3 set X\_ 160.000000000000 |
|  | $R3 set Y\_ 0.000000000000 |
|  | $R3 set Z\_ 0.000000000000 |
|  | $R4 set X\_ 160.000000000000 |
|  | $R4 set Y\_ -40.000000000000 |
|  | $R4 set Z\_ 0.000000000000 |
|  |  |
|  | # create links between wired and BaseStation nodes |
|  | $ns\_ duplex-link $W1 $W3 2Mb 20ms DropTail |
|  | $ns\_ duplex-link $W2 $W4 2Mb 20ms DropTail |
|  | $ns\_ duplex-link $W8 $W3 2Mb 20ms DropTail |
|  | $ns\_ duplex-link $W9 $W4 2Mb 20ms DropTail |
|  | $ns\_ duplex-link $W3 $W5 5Mb 20ms DropTail |
|  | $ns\_ duplex-link $W4 $W5 5Mb 20ms DropTail |
|  |  |
|  | $ns\_ duplex-link $W5 $BS1 5Mb 20ms DropTail |
|  | $ns\_ duplex-link $W5 $BS2 5Mb 20ms DropTail |
|  | # set the layout of links in NAM |
|  | $ns\_ duplex-link- op $W1 $W3 orient right |
|  | $ns\_ duplex-link- op $W8 $W3 orient right-down |
|  | $ns\_ duplex-link- op $W2 $W4 orient right-up |
|  | $ns\_ duplex-link- op $W9 $W4 orient right |
|  | $ns\_ duplex-link- op $W3 $W5 orient right-down |
|  | $ns\_ duplex-link- op $W4 $W5 orient right-up |
|  |  |
|  | $ns\_ duplex-link- op $W5 $BS1 orient right-up |
|  | $ns\_ duplex-link- op $W5 $BS2 orient right-down |
|  |  |
|  | $ns\_ at 0.0 &quot;$W1 label W1&quot; |
|  | $ns\_ at 0.0 &quot;$W2 label W2&quot; |
|  | $ns\_ at 0.0 &quot;$W8 label W3&quot; |
|  | $ns\_ at 0.0 &quot;$W9 label W4&quot; |
|  | $ns\_ at 0.0 &quot;$W3 label R1&quot; |
|  | $ns\_ at 0.0 &quot;$W4 label R2&quot; |
|  | $ns\_ at 0.0 &quot;$W5 label R3&quot; |
|  | $ns\_ at 0.0 &quot;$BS1 label BS1&quot; |
|  | $ns\_ at 0.0 &quot;$BS2 label BS2&quot; |
|  | $ns\_ at 0.0 &quot;$R1 label R1&quot; |
|  | $ns\_ at 0.0 &quot;$R2 label R2&quot; |
|  | $ns\_ at 0.0 &quot;$R3 label R3&quot; |
|  | $ns\_ at 0.0 &quot;$R4 label R4&quot; |
|  | $ns\_ at 0.0 &quot;$R1 add-mark m1 green circle&quot; |
|  | $ns\_ at 0.0 &quot;$R2 add-mark m1 red circle&quot; |
|  | $ns\_ at 0.0 &quot;$R3 add-mark m1 blue circle&quot; |
|  | $ns\_ at 0.0 &quot;$R4 add-mark m1 purple circle&quot; |
|  | # setup TCP connections |
|  | set tcp1 [new Agent/TCP/Newreno] |
|  | $tcp1 set packetSize\_ $size |
|  | $ns\_ attach-agent $W1 $tcp1 |
|  | set sink1 [new Agent/TCPSink] |
|  | $ns\_ attach-agent $R1 $sink1 |
|  | $ns\_ connect $tcp1 $sink1 |
|  | set ftp1 [new Application/FTP] |
|  | $ftp1 attach-agent $tcp1 |
|  | $ns\_ at 1.0 &quot;$ftp1 start&quot; |
|  | $ns\_ at 1.0 &quot;$ns\_ trace-annotate \&quot;W1 Sends packets to R1 via Home |
|  | Agent(BS1). \&quot;&quot; |
|  | set tcp2 [new Agent/TCP/Newreno] |
|  | $tcp2 set packetSize\_ $size |
|  | $ns\_ attach-agent $W8 $tcp2 |
|  | set sink2 [new Agent/TCPSink] |
|  | $ns\_ attach-agent $R2 $sink2 |
|  | $ns\_ connect $tcp2 $sink2 |
|  | set ftp2 [new Application/FTP] |
|  | $ftp2 attach-agent $tcp2 |
|  | $ns\_ at 2.0 &quot;$ftp2 start&quot; |
|  | $ns\_ at 2.0 &quot;$ns\_ trace-annotate \&quot;W3 Sends packets to R2 via Home |
|  | Agent(BS1). \&quot;&quot; |
|  | set tcp3 [new Agent/TCP/Newreno] |
|  | $tcp3 set packetSize\_ $size |
|  | $ns\_ attach-agent $W2 $tcp3 |
|  | set sink3 [new Agent/TCPSink] |
|  | $ns\_ attach-agent $R3 $sink3 |
|  | $ns\_ connect $tcp3 $sink3 |
|  | set ftp3 [new Application/FTP] |
|  | $ftp3 attach-agent $tcp3 |
|  | $ns\_ at 3.0 &quot;$ftp3 start&quot; |
|  | $ns\_ at 3.0 &quot;$ns\_ trace-annotate \&quot;W2 Sends packets to R3 via Home |
|  | Agent(BS2). \&quot;&quot; |
|  | set tcp4 [new Agent/TCP/Newreno] |
|  | $tcp4 set packetSize\_ $size |
|  | $ns\_ attach-agent $W9 $tcp4 |
|  | set sink4 [new Agent/TCPSink] |
|  | $ns\_ attach-agent $R4 $sink4 |
|  | $ns\_ connect $tcp4 $sink4 |
|  | set ftp4 [new Application/FTP] |
|  | $ftp4 attach-agent $tcp4 |
|  | $ns\_ at 4.0 &quot;$ftp4 start&quot; |
|  | $ns\_ at 4.0 &quot;$ns\_ trace-annotate \&quot;W4 Sends packets to R4 via Home |
|  | Agent(BS2). \&quot;&quot; |
|  |  |
|  | # Define initial node position in nam |
|  | $ns\_ initial\_node\_pos $R1 10 |
|  | $ns\_ initial\_node\_pos $R2 10 |
|  | $ns\_ initial\_node\_pos $R3 10 |
|  | $ns\_ initial\_node\_pos $R4 10 |
|  |  |
|  | # Tell all nodes when the siulation ends |
|  | $ns\_ at $opt(stop).0 &quot;$R1 reset&quot;; |
|  | $ns\_ at $opt(stop).0 &quot;$R2 reset&quot;; |
|  | $ns\_ at $opt(stop).0 &quot;$R3 reset&quot;; |
|  | $ns\_ at $opt(stop).0 &quot;$R4 reset&quot;; |
|  | $ns\_ at $opt(stop).0002 &quot;puts \&quot;NS EXITING...\&quot; ; $ns\_ halt&quot; |
|  | $ns\_ at $opt(stop).0001 &quot;stop&quot; |
|  | proc stop {} { |
|  | global ns\_ tracefd namtrace opt f0 |
|  | global ns\_ tcp1 tcp2 tcp3 tcp4 |
|  | global sink |
|  | $ns\_ flush-trace |
|  | close $tracefd |
|  | close $f0 |
|  | close $namtrace |
|  |  |
|  | # close $tracefd |
|  | # close $namtrace |
|  | exec nam out.nam &amp; |
|  | exit 0 |
|  | } |
|  | puts &quot;Starting Simulation...&quot; |
|  | $ns\_ run |
|  | # |