ns-2 Tutorial

Running Simulations

Matthias Transier

transier@informatik.uni-mannheim.de

Universität Mannheim

Based on a tutorial by Marc Greis

Overview

- Creating a wireless scenario
  - Node configuration
  - Basic simulation setup
  - Movement definition
  - Trace file analysis

Scenario definition

- Area: 500m $\times$ 500m

Scenario definition

- Area: 500m $\times$ 500m
- 2 mobile nodes
  - one moving from left to right, the other vice versa

Scenario definition

- Area: 500m $\times$ 500m
- 2 mobile nodes
  - one moving from left to right, the other vice versa
- A TCP connection between them

Scenario definition

- Area: 500m $\times$ 500m
- 2 mobile nodes
  - one moving from left to right, the other vice versa
- A TCP connection between them
- Expectation: the nodes exchange data as long as they are in radio range of each other

Define options

- Specify components of the mobile node:

```
set val(chan)      Channel/WirelessChannel  ;# channel type
set val(prop)      Propagation/TwoRayGround ;# radio-propagation model
set val(ant)       Antenna/OmniAntenna      ;# Antenna type
set val(ll)        LL                       ;# Link layer type
set val(ifq)       Queue/DropTail/PriQueue  ;# Interface queue type
set val(ifqlen)    50                       ;# max packet in ifq
set val(netif)     Phy/WirelessPhy          ;# network interface type
set val(mac)       Mac/802_11               ;# MAC type
set val(rp)        DSDV                     ;# ad-hoc routing protocol
set val(nn)        2                        ;# number of mobilenodes
```

Basic setup

- Create simulator instance:

```
set ns_ [new Simulator]
```

- Open trace fi le and activate it:

```
set tracefd [open wireless.tr w]
$ns_ trace-all $tracefd
```

- Create topography and channel:

```
set topo [new Topography]
$topo load_flatgrid 500 500
set chan [new $val(chan)]
```

- Create the GOD object (General Operations Director):

```
create-god $val(nn)
```

The GOD object

- Stores information that an omniscient observer would have
  - number of nodes
  - connectivity information which else would have to be calculated on-the-fly

The GOD object

- Stores information that an omniscient observer would have
  - number of nodes
  - connectivity information which else would have to be calculated on-the-fly
- One single GOD object per simulation

The GOD object

- Stores information that an omniscient observer would have
  - number of nodes
  - connectivity information which else would have to be calculated on-the-fly
- One single GOD object per simulation
- Needed by the MAC layer

The GOD object

- Stores information that an omniscient observer would have
  - number of nodes
  - connectivity information which else would have to be calculated on-the-fly
- One single GOD object per simulation
- Needed by the MAC layer
- Gives the possibility to evaluate e.g. optimality of routes

Configuration of the mobile 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) \
                 -topoInstance $topo \
                 -channel $chan \
                 -agentTrace ON \
                 -routerTrace ON \
                 -macTrace OFF
```

Creating the nodes

- After setting the confi guration options, the nodes are created:

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

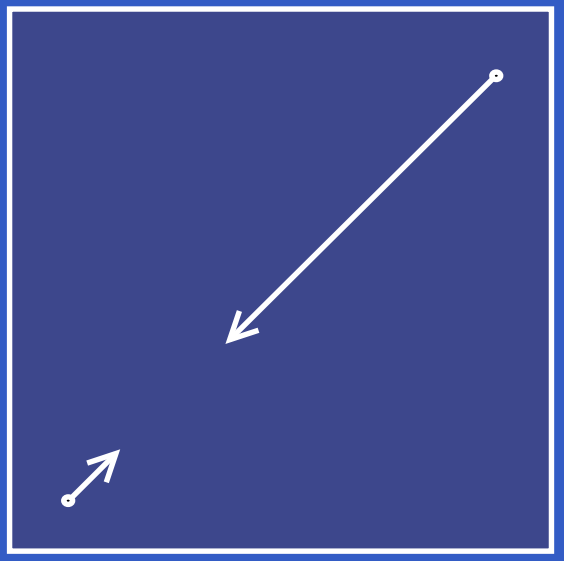
- The node movement will be explicitly provided in the following.

Movement specification

- Set start position:

```
$node_(0) set X_ 5.0
$node_(0) set Y_ 2.0
$node_(0) set Z_ 0.0

$node_(1) set X_ 390.0
$node_(1) set Y_ 385.0
$node_(1) set Z_ 0.0
```



- Node 1 starts to move towards node 0:

```
$ns_ at 50.0 "$node_(1) setdest 25.0 20.0 15.0"
$ns_ at 10.0 "$node_(0) setdest 20.0 18.0 1.0"
```

- Node 1 then starts to move away from node 0:

```
$ns_ at 100.0 "$node_(1) setdest 490.0 480.0 15.0"
```

Connection setup

- TCP connection from node 0 to node 1

```
set tcp [new Agent/TCP]
set sink [new Agent/TCPSink]
$ns_ attach-agent $node_(0) $tcp
$ns_ attach-agent $node_(1) $sink
$ns_ connect $tcp $sink
```

- Create data source

```
set ftp [new Application/FTP]
$ftp attach-agent $tcp
$ns_ at 10.0 "$ftp start"
```

Starting the scheduler

```
for {set i 0} {$i < $val(nn) } {incr i} {
    $ns_ at 150.0 "$node_($i) reset";
}

$ns_ at 150.0001 "stop"
$ns_ at 150.0002 "puts \"NS EXITING...\" ; $ns_ halt"
proc stop {} {
    global ns_ tracefd
    close $tracefd
}

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

Analysis of the trace file

Format:

```
event time node level  --- pktnr type pktsize [MAC info] ...
```

Data packet:

```
s 100.000000000 _0_ AGT  --- 21 tcp 40 [0 0 0 0] ------- [...]
r 100.000000000 _0_ RTR  --- 21 tcp 40 [0 0 0 0] ------- [...]
s 100.000000000 _0_ RTR  --- 21 tcp 60 [0 0 0 0] ------- [...]
r 100.000644018 _1_ AGT  --- 21 tcp 60 [13a 1 0 800] ------- [...]
```

Acknowledgment:

```
s 100.000644018 _1_ AGT  --- 22 ack 40 [0 0 0 0] ------- [...]
r 100.000644018 _1_ RTR  --- 22 ack 40 [0 0 0 0] ------- [...]
s 100.000644018 _1_ RTR  --- 22 ack 60 [0 0 0 0] ------- [...]
r 100.001552036 _0_ AGT  --- 22 ack 60 [13a 0 1 800] ------- [...]
```