Ns Simulation of IEEE 802.11

SC546 Project (Fall 2002)

SOO IL KIM (kimsooil@bu.edu)
JISUN YOON (faithink@bu.edu)
Project descriptions

- **Goals**
  - Understand the IEEE 802.11
  - Do wireless LAN simulations using Ns

- **Focus**
  - Ad hoc networking
  - Collision avoidance (RTS/CTS handshake)
Wired vs. Wireless

- **Wireless communication**
  - No wired links: radio, infrared, laser
  - Ad hoc network

- **Problems in Wireless Network (IEEE 802.11)**
  - No multi-hop awareness
  - Hidden/Exposed
  - Unfairness
  - Packet drop is occurred often by errors in transmission layer

- **(Compare) Problems in Wired network**
  - Major cause of dropped packets: Congestion in Routers
Hidden/Exposed node

Data transmission from A to B

Hidden node = D (possibly Deaf node)
  - Cause packet collision

Exposed node = C
  - Prohibited from transmitting

From “The deaf node problem in Ad hoc wireless LANs” by S. Ray, D. Starobinski, and J.B. Carrunthers

SC546 (Fall 2002) Ns Simulation of IEEE 802.11
802.11 Operations (#1)

1. [Diagram showing S, R, and B with A and C]

2. [Diagram showing S, R, A, B, and C with RTS]

3. [Diagram showing S, R, A, B, and C with CTS]

4. [Diagram showing S, R, A, B, and C with overlapping circles]

SC546 (Fall 2002)  Ns Simulation of IEEE 802.11
Receive RTS: Defer until CTS should have been sent
Receive CTS: Defer until Data should have been sent
If you don’t receive CTS or ACK, back off and try it all over again

(from http://www-ece.rice.edu/~ashu/reneclass/lectures/elec437lecture2.pdf)
A discrete event simulator targeted at networking research

The collaboration of USC/ISI, LBL, UCB, and Xerox PARC

Two main components: Ns, Nam

Validation is needed
Support wired/wireless models

- Traffic models and applications
  - Web, FTP, telnet, constant-bit rate, stochastic
- Transport protocols
  - Unicast: TCP (Reno, Vegas, etc.), UDP
  - Multicast: SRM
- Routing and queueing
  - Wired routing, ad hoc routing and directed diffusion
  - Queueing protocols: RED, drop-tail, etc.
- Physical media
  - Wired (point-to-point, LANs), wireless (multiple propagation models), satellite

Tracing, visualization using Nam
Ns Programming

- Create the event scheduler
- Turn tracing
- Create network
- Setup routing
- Insert errors
- Create transport connection
- Create traffic
- Transmit application-level data

Using Ns
(from http://www.isi.edu/nsnam/ns/ns-tutorial/)
Environments/Configurations

- 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(nn) 4 ;# number of mobilenodes
- set val(rp) AODV ;# routing protocol
- set val(x) 800
- set val(y) 800
Simulation #1

**Scenario**
- Two fixed nodes
- moving within 600m x 600m flat topology
- DSR ad hoc routing
- TCP and CBR traffic
- Receiver move in and out of range

**Results**
- Time vs. packets arrived
Simulation #2

- Scenario
  - Two fixed pairs (4 nodes)
  - moving within 800m x 800m flat topology
  - AODV ad hoc routing
  - TCP and CBR traffic
  - 2 nodes in each pair communicate each other (hidden node)

- Results
  - Time vs. packets arrived
Simulation #3

- **Scenario**
  - Six fixed nodes
  - Change Routing algorithm
  - 4 Ad hoc routing: DSR/ DSDV/ AODV/ TORA
  - *The left-most node sends data to the right-most node*

- **Results**
  - Time vs. packets arrived
Further studies

- Check the effectiveness of RTS/CTS handshake
- Consider a lot of nodes in a small space
- More experiments using other traffic model (e.g. burst)
- Source-level (C++) modification for deeper understanding
Useful links

- Monarch project
  - [http://www.monarch.cs.rice.edu](http://www.monarch.cs.rice.edu)
  - (more links will be added on the web)
That’s all

- Thanks.