OMNeT++ Community Summit, 2015

INET 3.0 Wireless Tutorial

IBM Research - Zurich, Switzerland – September 3 - 4, 2015

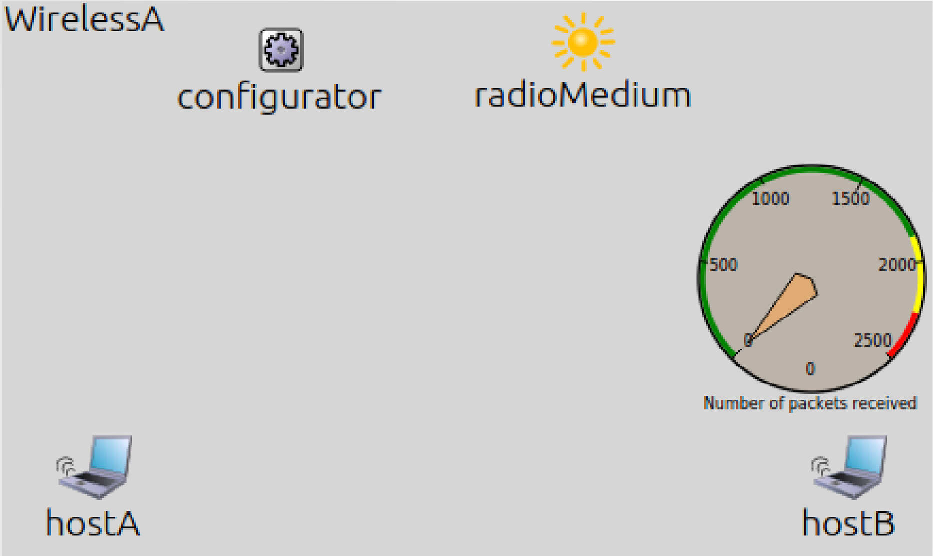
Rudolf Hornig

INET 3.0 Wireless Tutorial

Outline:

The tutorial consists of 13 steps with increasingly more realistic wireless models

Step1a: The Network



```
@figure[thruputInstrument](
    type=gauge;
    pos=370,90; size=120,120;
    maxValue=2500; tickSize=500;
    colorStrip=green 0.75 yellow 0.9 red;
    label=Number of packets received;
    moduleName=hostB.udpApp[0];
    signalName=rcvdPk);
```

```
network WirelessA
{
    @display("bgb=500,500");
    @figure[thruputInstrument](type=gauge;...);
    string hostType = default("WirelessHost");
    string mediumType =
              default("IdealRadioMedium");
    submodules:
        configurator: IPv4NetworkConfigurator;
        radioMedium: <mediumType> like
                                IRadioMedium;
        hostA: <hostType> like INetworkNode;
        hostB: <hostType> like INetworkNode;
}
```

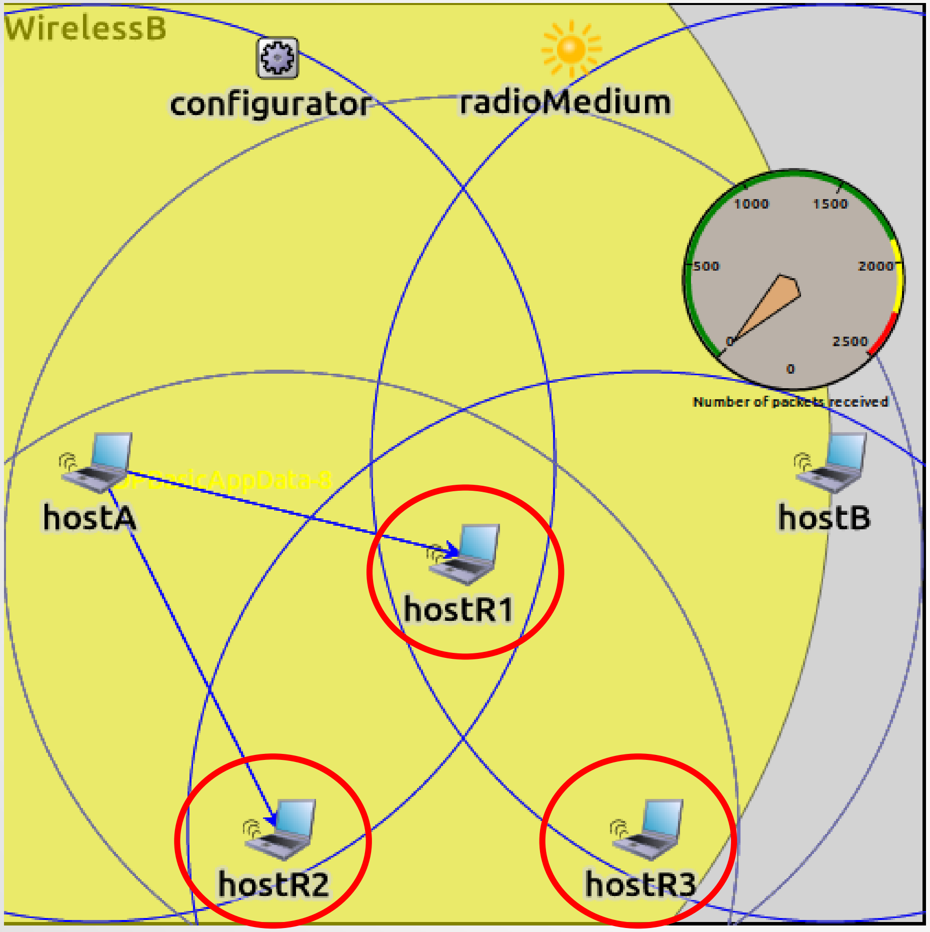
Step 1b: Set Up the Communication

```
sim-time-limit = 25s
*.hostA.numUdpApps = 1
*.hostA.udpApp[0].typename = "UDPBasicApp"
*.hostA.udpApp[0].destAddresses = "hostB"
*.hostA.udpApp[0].destPort = 5000
*.hostA.udpApp[0].messageLength = 1000B
*.hostA.udpApp[0].sendInterval = exponential(10ms)
# We expect around 2500 packets to be transmitted during the whole simulation.
# In reality we have 2453 packets transmitted in total.

•*.hostB.numUdpApps = 1
•*.hostB.udpApp[0].typename = "UDPSink"
•*.hostB.udpApp[0].localPort = 5000

•*.host*.wlan[*].typename = "IdealWirelessNic" # unit disc radio
•**.bitrate = 1Mbps
•*.host*.wlan[*].radio.transmitter.maxCommunicationRange = 500m
•*.host*.wlan[*].radio.receiver.ignoreInterference = true
```

Step 2: Enhancing the Animation

```
# Expanding circles to show transmission in every 100ns
*.radioMedium.mediumVisualizer.displayCommunication = true
*.radioMedium.mediumVisualizer.updateCanvasInterval = 100ns

# Show fading arrows where communication happen
*.radioMedium.mediumVisualizer.leaveCommunicationTrail = true
```

Step 3: Adding New Nodes



Note that no traffic reaches hostB!

NED:

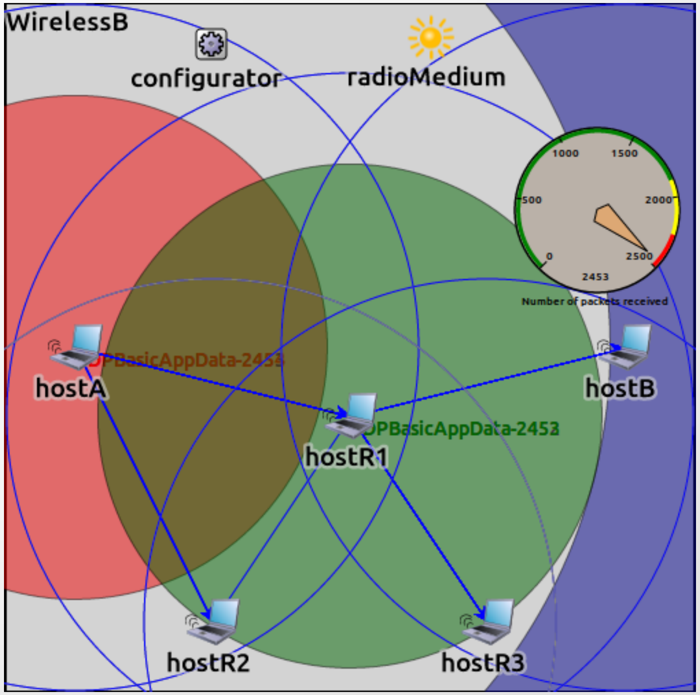
```
network WirelessB extends WirelessA
{
    submodules:
        hostR1: <hostType> like INetworkNode {
            @display("p=250,300");
        }
        hostR2: <hostType> like INetworkNode {
            @display("p=150,450");
        }
        hostR3: <hostType> like INetworkNode {
            @display("p=350,450");
        }
}
```

Ini:

```
*.host*.wlan[*].radio.transmitter.maxCommunicationRange = 250m
```

Step 4: Set Up Static Routing

```
# Enable static routing
*.host*.forwarding = true

# Configure routes and IP addresses statically (based on errorRate)
# routes are configured only for in-range nodes
*.configurator.config = xml("
  <config>
    <interface hosts='**' address='10.0.0.x' netmask='255.255.255.0'/>
    <autoroute metric='errorRate'/>
  </config>")
```



hostB received all packets (2453) = 100%!

Step 5: Enable Interference

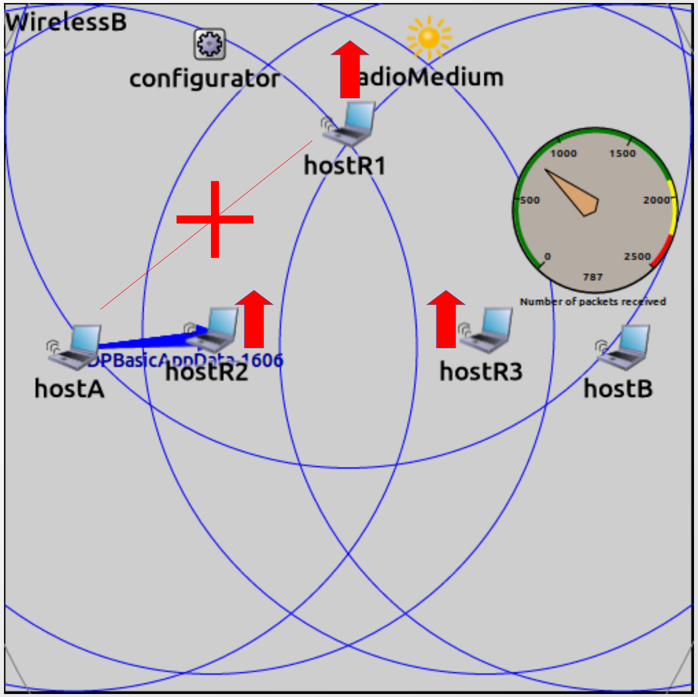
```
*.host*.wlan[*].radio.receiver.ignoreInterference = false

*.host*.wlan[*].radio.transmitter.maxInterferenceRange = 500m

# Unfortunately, almost no packet gets through (only around 40),
# because of collisions between the hostA and hostR1 packets
```

Step 6. Using CSMA

```
# We want to simulate collision avoidance, too
*.host*.wlan[*].typename = "WirelessNic"
*.host*.wlan[*].radioType = "IdealRadio"
*.host*.wlan[*].macType = "CSMA"
*.host*.wlan[*].mac.useMACAcks = true
# Using CSMA 1172 packet were transmitted (48%)
```

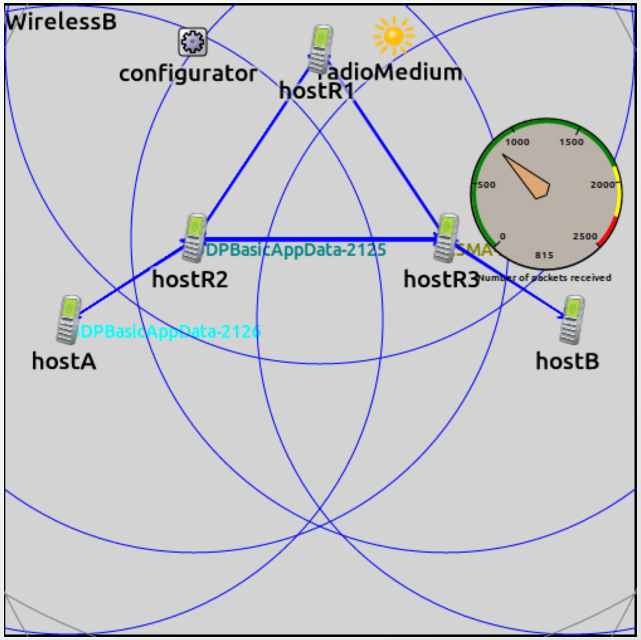
Step 7: Move the Nodes

```
# When moving the hostR* nodes northward the communication
# breaks down as soon as hostR1 gets out of range
*.hostR*.mobilityType = "LinearMobility"

*.hostR*.mobility.speed = 12mps

*.hostR*.mobility.angle = 270deg

# only 787 packets get through
# which is 32%
```



Step 8: Configure Adhoc Routing

```
# Turn off the static configurator (assign only IP
# addresses) and use AODV routing instead

*.configurator.addStaticRoutes = false

*.configurator.addDefaultRoutes = false

*.configurator.addSubnetRoutes = false

*.hostType = "AODVRouter"

# Route is reconfigured once
# hostR1 gets out of range
# so 890 packets are transmitted
```

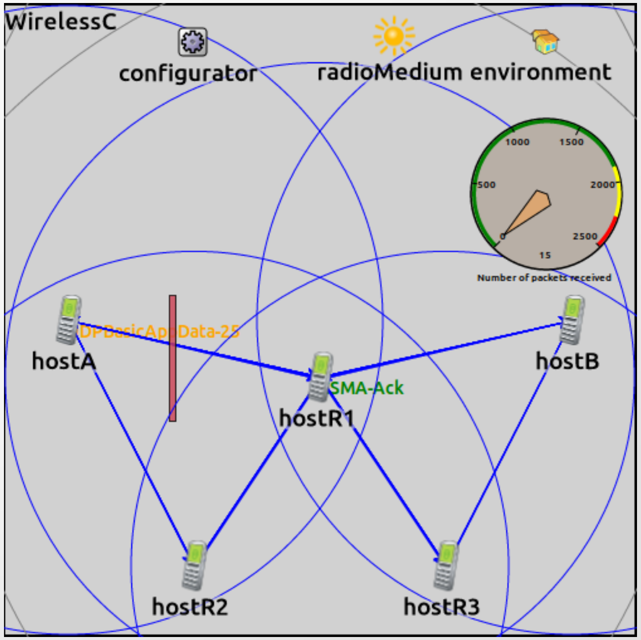


Step 9: Install a Battery

```
**.energyConsumerType = "RadioStateBasedEnergyConsumer"

*.host*.energyStorageType = "IdealEnergyStorage"

# You can watch the host*.energyStorage.energyBalance variable
# to see how the node's energy consumed over time.
```

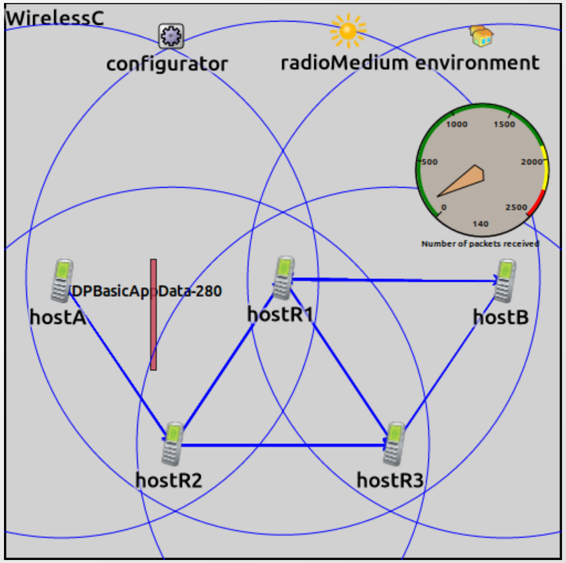
Step 10: Using Obstacles

```
*.environment.config = xmldoc("wall.xml")
*.radioMedium.obstacleLossType = "TracingObstacleLoss"

wall.xml:
<environment>
  <object position="min 130 230 0" orientation="0 0 0"
          shape="cuboid 5 100 3" material="concrete"
          fill-color="203 65 84" opacity="0.8"/>
</environment>

# Unfortuantely the wall is not blocking
# the transmission because of the
# simplified radio model we are using.
# (still trasmitting 890 packets)
```



Step 11: Enhanced Transmission Modeling

```
*.mediumType = "APSKScalarRadioMedium"
*.radioMedium.backgroundNoise.power = -110dBm
*.host*.wlan[*].radioType = "APSKScalarRadio"
*.host*.wlan[*].radio.carrierFrequency = 2GHz
*.host*.wlan[*].radio.bandwidth = 2MHz
*.host*.wlan[*].radio.transmitter.power = 1.2mW
*.host*.wlan[*].radio.transmitter.headerBitLength = 100b
*.host*.wlan[*].radio.receiver.sensitivity = -85dBm
*.host*.wlan[*].radio.receiver.energyDetection = -85dBm
*.host*.wlan[*].radio.receiver.snirThreshold = 4dB

# Radio signals are now blocked by
# the installed wall (482 packets)
```

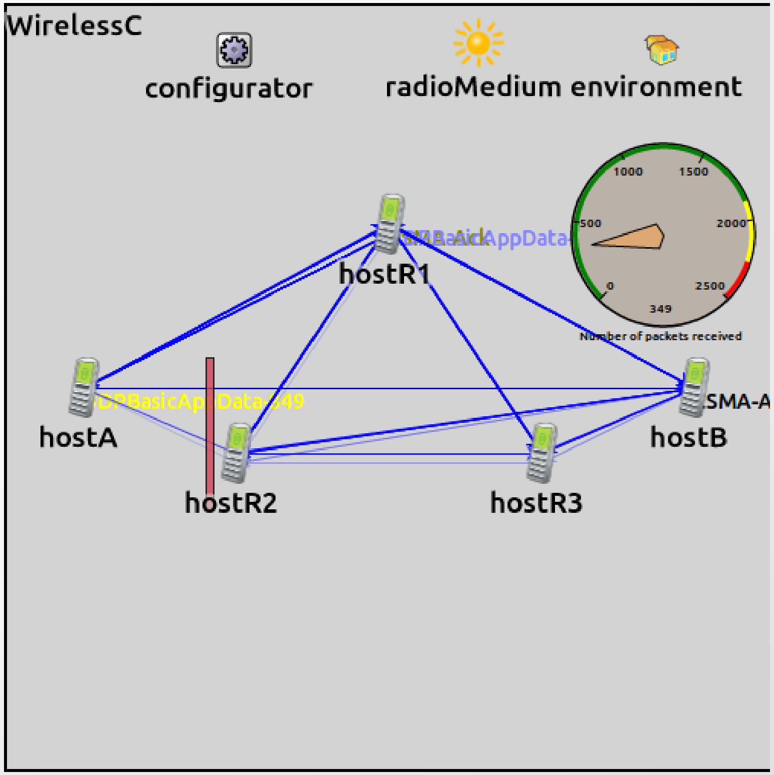


Step 12: Adding Pathloss Model

```
# We configure a more accurate pathloss model

*.radioMedium.pathLossType = "TwoRayGroundReflection"

*.radioMedium.pathLoss.transmitterAntennaHeight = 1.5m

*.radioMedium.pathLoss.receiverAntennaHeight = 1.5m

# As a result of the more accurate model, the number of
# transmitted packets dropped to 242 (10%)
```

Step 13: Configuring Antennas

```
# Increase the antenna gain

*.host*.wlan[*].radio.antennaType = "ConstantGainAntenna"

*.host*.wlan[*].radio.antenna.gain = 12dB

# The increased gain allows
# some packets to get through
# even the wall (879 packets).
```



More steps (planned)

- Using directionaly antenna
- Route changes on battery depletion
- Adding radio noise on a nearby frequency
- Support cross-talk with dimensional radio model
- Introduce short high energy bursts in the channel
- Bit precise radio model
- Add forward error correction, scrambling, interleaving
- Optimization: MAC and range fiters

Thank you for your Attention!

- Ideas for additional steps are welcomed
- More tutorials on different topics are welcomed
- Anyone wants to do a YouTube toturial?