Wiki of the Institute of Telematics

Visualisation of NS-2 propagation models

For better understanding of NS-2 propagation models we present easy scripts for visualisation the propagation behaviour (here + all scripts used below). For running the Tcl-Script you also need a special neighbouhood agent, that have to be compiled together with NS-2 (source). If you have general questions to the topic propagation please look at Understanding Propagation Models.

The downloadable archive contains two parts, the NS-2 simulation script and the Tex (pgf/TikZ) visualisation script. The first one is used to simulate the propagation behaviour. Therefor nodes are regularly deployed in a rectangle area (grid topology). The central node sends a fixed number of packets, whereas all surrounding nodes are counting the packets they received. The second script is used for the visualisation of the simulation results.

First a file containing the configuration of a propagation model has to be written. A file for the shadowing model would look as follows:

file shadowing.tcl

```
# customize simulation
set gridX   31;                 # number of nodes in a horizontal line
set gridY   31;                 # number of nodes in a vertical line
set numberOfPackets     2000;   # packets used for sending
set opt(width)          200;    # topography width
set opt(height)         200;    # topography height

# set propagation model
set opt(prop) Propagation/Shadowing;

# parameters for shadowing
Phy/WirelessPhy set Pt_ 0.28183815
Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0
Phy/WirelessPhy set freq_ 914e+6
Phy/WirelessPhy set L_ 1.0
Propagation/Shadowing set pathlossExp_ 2.0
Propagation/Shadowing set std_db_ 2.8
Propagation/Shadowing set dist0_ 1.0
Phy/WirelessPhy set RXThresh_ 3.3e-8
```

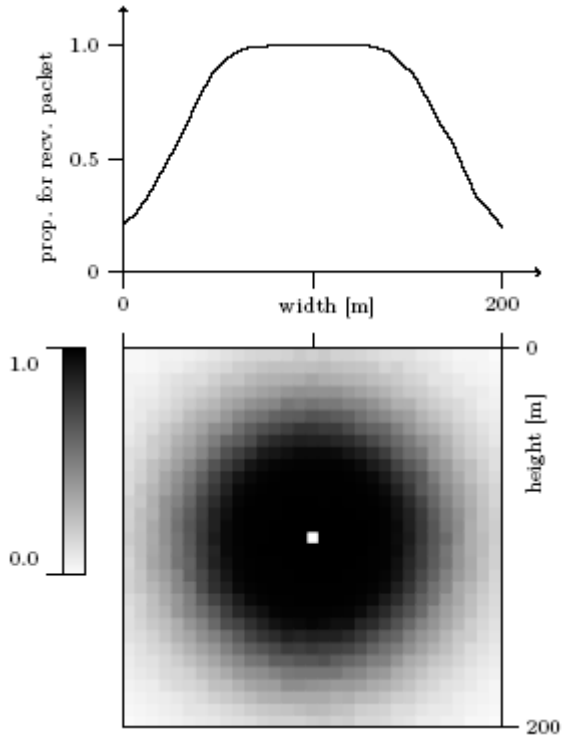
This file named as `shadowing.tcl` can be simulated by using the following command (in the folder sim_prop_model):

```
ns simulate.tcl shadowing.tcl
```

Finally the results can be visualized by using the following command (make sure a valid tex distribution with pgf/TikZ is installed):

```
pdflatex document.tex
```

The resulting diagram looks like this:



The upper graph shows the propability of receiving a packet by using the middle horizontal line of nodes (central node is skipped). The other graph is a 2D area plot (using all data). In black regions nodes can receive almost all packets. On the other hand the receiving propability in white regions is rather bad.

Shadowing Model

The results of the shadowing models are shown above. And analytic verification of the results can be found under Plotting receiving probabilities of shadowing model.

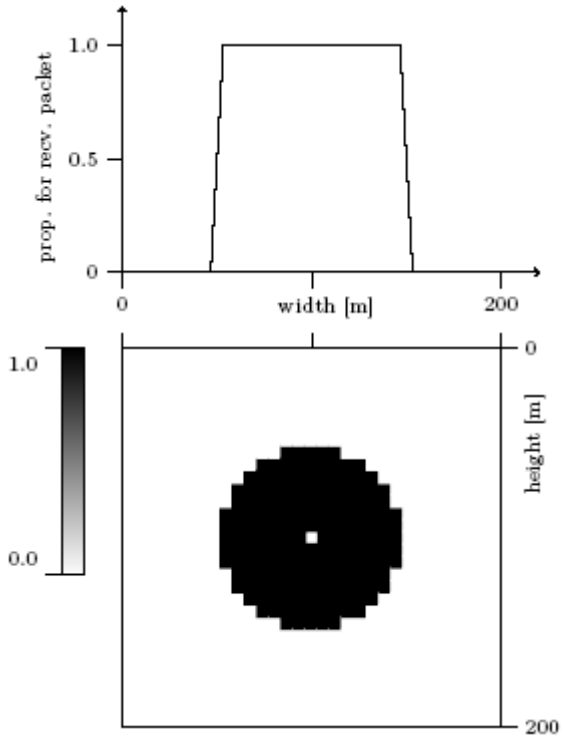
Two-ray ground reflection model

file two_ray_ground.tcl

```
# customize simulation
set gridX   31;                 # number of nodes in a horizontal line
set gridY   31;                 # number of nodes in a vertical line
set numberOfPackets     2000;   # packets used for sending
set opt(width)          200;    # topography width
set opt(height)         200;    # topography height

# set propagation model
set opt(prop) Propagation/TwoRayGround;

# parameters for shadowing
Phy/WirelessPhy set RXThresh_ 7.69113e-8; # 50 m
```



Note, that the NS-2 Free space model would have the same simulation result like the Two-ray ground reflection model.

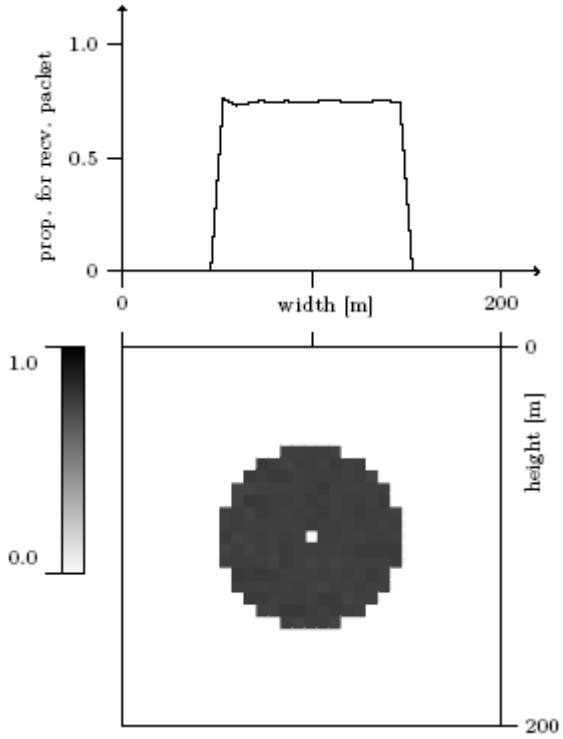
Random model

Behaviour of the propagation model developed in Writing a Propagation Model.

file random_prop.tcl

```
# customize simulation
set gridX       31;             # number of nodes in a horizontal line
set gridY       31;             # number of nodes in a vertical line
set numberOfPackets     2000;   # packets used for sending
set opt(width)          200;    # topography width
set opt(height)         200;    # topography height

# set propagation model
set opt(prop) Propagation/Random;
Propagation/Random set lossRate_ 0.25
Propagation/Random set range_ 50
```



Asymmetric Propagation Proxy

Here the behaviour of the Asymmetric Propagation Proxy we developed is shown. Please look under Asymmetric Propagation Proxy for further information.

Using the shadowing model (random definition of profile):

file approxy.tcl

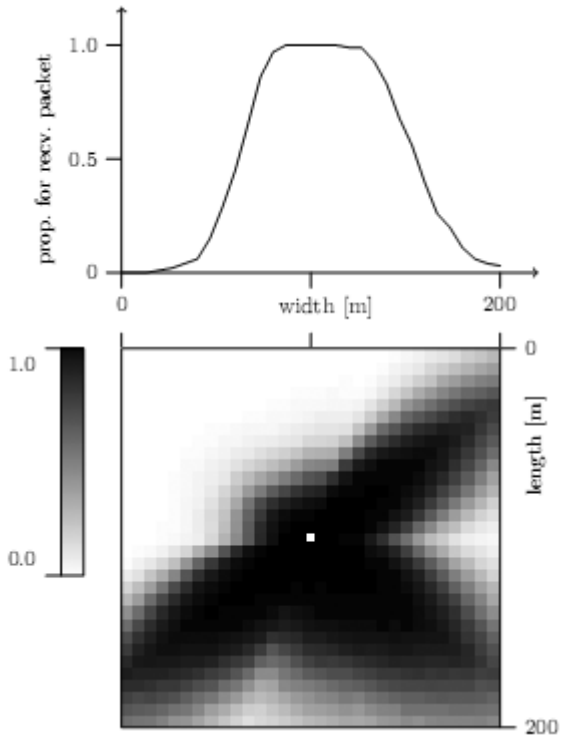
```
# customize simulation
set gridX   31;                 # number of nodes in a horizontal line
set gridY   31;                 # number of nodes in a vertical line
set numberOfPackets     2000;   # packets used for sending
set opt(width)          200;    # topography width
set opt(height)         200;    # topography height

Propagation/APProxy set minRandomGain_ -2.0;
Propagation/APProxy set maxRandomGain_ 2.0;
Propagation/APProxy set maxGain_ 10.0;
Propagation/APProxy set angleNumber_ 10;

# parameters for shadowing
Phy/WirelessPhy set Pt_ 0.28183815
Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0
Phy/WirelessPhy set freq_ 914e+6
Phy/WirelessPhy set L_ 1.0
Propagation/Shadowing set pathlossExp_ 2.0
Propagation/Shadowing set std_db_ 2.8
Propagation/Shadowing set dist0_ 1.0
Phy/WirelessPhy set RXThresh_ 3.3e-8

$defaultRNG seed 1000;

# set propagation model
set opt(propInstance) [new Propagation/APProxy]
$opt(propInstance) propagation [new Propagation/Shadowing]
```



Using the Two-ray ground reflection model (manual definition of profile):

file approxy2.tcl

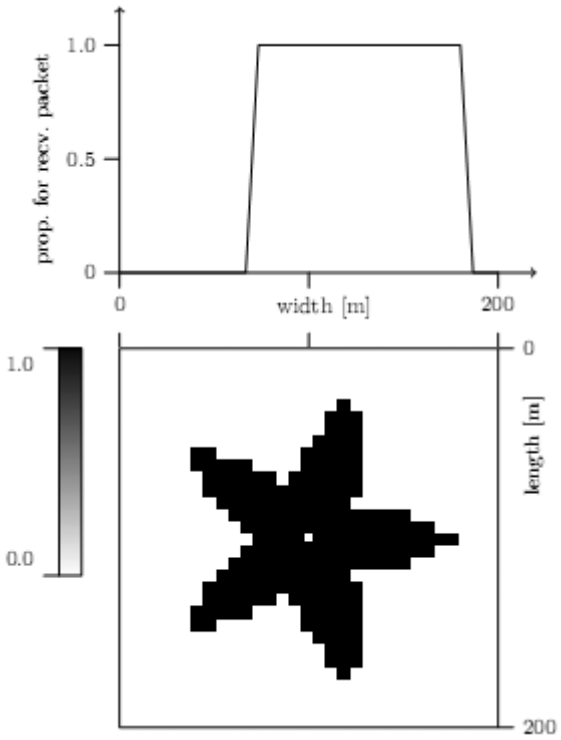
```
# customize simulation
set gridX   31;                 # number of nodes in a horizontal line
set gridY   31;                 # number of nodes in a vertical line
set numberOfPackets     2000;   # packets used for sending
set opt(width)          200;    # topography width
set opt(height)         200;    # topography height

Propagation/APProxy set minRandomGain_ -2.0;
Propagation/APProxy set maxRandomGain_ 2.0;
Propagation/APProxy set maxGain_ 10.0;
Propagation/APProxy set angleNumber_ 10;

Phy/WirelessPhy set RXThresh_ 7.69113e-8; # 50 m

# set propagation model
set opt(propInstance) [new Propagation/APProxy]
$opt(propInstance) propagation [new Propagation/TwoRayGround]

# [expr [expr $gridX*$gridY-1] /2] calculates the middle node
$opt(propInstance) profile [expr [expr $gridX*$gridY-1] /2]  1.0 -1.0  1.0 -1.0 \
                                                    1.0 -1.0  1.0 -1.0  1.0 -1.0
```



Frequency-shift keying (FSK) Propagation Model

FSK Propagation is an additional model for NS-2. It is an implementation of the propagation model presented inchannelmodellingSECON04.pdf [http://ceng.usc.edu/~bkrishna/research/papers/channelmodellingSECON04.pdf]. The model is used for simulating frequency-shift keying in low power wireless links (e.g. simulating MIKA motes). Unfortunately NS-2 has no sufficient interface for integrating such a model. Thus some variables like the frame size have to be set statically via Tcl. For further information about the binded variables please read the publication.

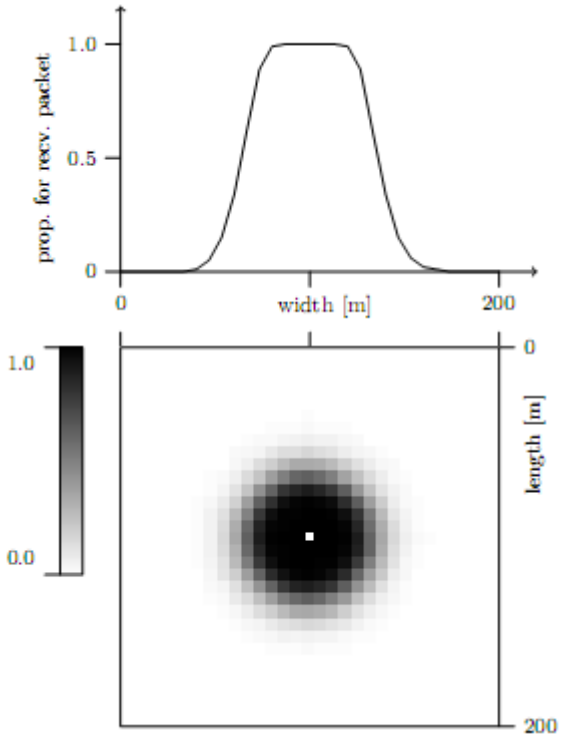
The behaviour is similar to the shadowing model:

file fsk_prop

```
# customize simulation
set gridX       31;             # number of nodes in a horizontal line
set gridY       31;             # number of nodes in a vertical line
set numberOfPackets     2000;   # packets used for sending
set opt(width)          200;    # topography width
set opt(height)         200;    # topography height

Propagation/fsk set frameLength_ 50.0
Propagation/fsk set pathLoss_ 3.8
Propagation/fsk set sigma_ 4.0
Propagation/fsk set dist0_ 1.0
Propagation/fsk set noiseBandwidth_ 30000
Propagation/fsk set dataRate_ 19200
Propagation/fsk set noiseFloor_ -105

# set propagation model
set opt(prop) Propagation/fsk;
```



It is possible to use the FSK model together with the Asymmetric Propagation Proxy, like shown below:

file asymmetricfsk_prop.tcl

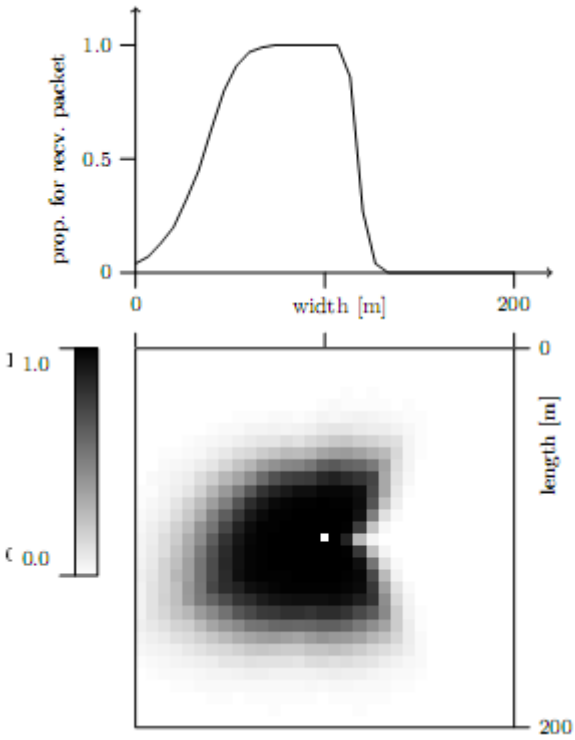
```
# customize simulation
set gridX       31;             # number of nodes in a horizontal line
set gridY       31;             # number of nodes in a vertical line
set numberOfPackets     2000;   # packets used for sending
set opt(width)          200;    # topography width
set opt(height)         200;    # topography height

Propagation/APProxy set minRandomGain_ -3.0;
Propagation/APProxy set maxRandomGain_ 3.0;
Propagation/APProxy set maxGain_ 10.0;
Propagation/APProxy set angleNumber_ 7;

Propagation/fsk set frameLength_ 50.0
Propagation/fsk set pathLoss_ 3.8
Propagation/fsk set sigma_ 4.0
Propagation/fsk set dist0_ 1.0
Propagation/fsk set noiseBandwidth_ 30000
Propagation/fsk set dataRate_ 19200
Propagation/fsk set noiseFloor_ -105


$defaultRNG seed 111;

# set propagation model
set opt(propInstance) [new Propagation/APProxy]
$opt(propInstance) propagation [new Propagation/fsk]
```


1.0
0.5
0
prop. for recv. packet
0
width [m]
200
1.0
0.0
0
length [m]
200