Mobile Communication
Summer 2008

Matthias Frank, Peter Martini;
Christoph Barz, Christoph Fuchs, Lukas Pustina, Christian de Waal

Practical Assignment Sheet

Release date: April 22th, 2008. Presentation anticipated at the 2nd half of June 2008.

General information on the practical exercises:

- The practical exercises are to be solved and presented in groups of two students.
- The presentation will take place in the 2nd half of June – further information will be given in the exercises and on the mailing lists.
- For the presentation, produce printouts of your plots and prepare to explain, interpret and answer questions regarding the plots.
- During the presentation, your programs must run error-free. Furthermore, you should be able to answer questions regarding design and implementation issues.

General information on ns-2:

The practical exercise consists of a simulation and evaluation part with the network simulator ns-2 and an implementation part in Java. Documentation for ns-2 can be found on the ns-2 web site at

`http://www.isi.edu/nsnam/`

To get started with ns-2, we strongly recommend Marc Greis' ns-2 tutorial:

`http://www.isi.edu/nsnam/ns/tutorial/`

Especially section IX, "running wireless simulations in ns" should be of interest to you.

Information on running the network simulator 2 (ns-2) on our CIP-pool computers:

Location of ns-2 binaries:

From the Linux computers in our CIP-pool (e.g. zeus or heilbutt), you have access to a modified version of ns-2, which is installed in the directory

`/home/work/Rechnernetze/SS08/ns-allinone-2.28/ns-2.28`

Trace files:

To avoid difficulties with large trace files, you can

1. write raw trace output to the `/tmp` directory and process it there and
2. compress the output using the following trace file descriptor assignment in your TCL script:

```
set trace [open "| gzip > trout.tr.gz"]
```

Practical Assignment 1: (Wireless links)

In the lecture, the properties of a wireless channel have been introduced. Data transmission is only successful if the signal strength at the receiver is sufficiently high. A (bidirectional) link between two stations exists, if communication is possible in both directions.

Within scenarios deploying mobile nodes, wireless links are frequently lost when two stations move too far apart from each other. But even in scenarios without mobility, the links between stations are instable. In this task you shall analyse the effects of "small scale fading" on wireless links. Small scale fading denotes the fluctuations in signal strength which occur with only small or even without any movements of sender and receiver.

Your task is to simulate a wireless network with 80 nodes which are positioned randomly and uniformly distributed across an area of 500 m x 500 m. You can find a howto on the creation of wireless scenarios as well as an example script in chapter IX of the tutorial of Marc Greis. Do not use the deprecated method for the creation of a wireless channel as is shown in the howto, but insert the line

```
set chan_1 [new $val(chan)]
```

in front of the `node-config` command and use

```
-channel $chan_1
```

instead of

```
-channelType $val(chan)
```

Also set the transmitting power of all stations to 40 mW by adding the command

```
Phy/WirelessPhy set Pt_ .04
```

in your TCL-Script (in front of `node-config`).

In order to deploy the "Ricean small scale fading" in your simulation, use the module `Propagation/SimpleRice` which is included in the provided version of the simulator.

To monitor when wireless links between stations are established and at which times they break, use another module of this simulator version which provides a combination of link detection and power control. For link detection, each station transmits so-called "beacons" in static time intervals. If two stations mutually receive a sufficient number of their respective beacons, they assume the existence of a bidirectional link between them (further on, this will be called "link-up event"). In detail, both stations must have received at least the fraction of `link_up_threshold` beacons of the last `up_beacons` beacons sent; if this value falls below `link_down_threshold` for one of the stations the link is considered as no longer existent (further on called "link-down event").

For further details on the usage and configuration of the power control module, refer to the file

```
/home/work/Rechnernetze/SS08/ns-allinone-2.28/ns-2.28/Powercontrol/ctc.txt
```

In your analysis, you shall consider a link as existent when both stations announce it (in the logfile) and as broken when one of either station announces the link to be down. Unidirectional links are not considered.

The following tasks are to be solved:

a) Initially, execute three simulation runs with a duration of 400 s each, where up_beacons=15 and

link_up_threshold = 1, link_down_threshold = 0.8
link_up_threshold = 0.9, link_down_threshold = 0.7
link_up_threshold = 0.8, link_down_threshold = 0.6

Create plots for these simulation runs reflecting the development of the following network properties over time:

- Overall number of existing links (as measure for the "connectivity" of the topology)
- Number of link-up and link-down events per time unit (as measure for the "dynamics" of the topology)

Furthermore, create histograms of the neighbour counts (number of bidirectional links) of the stations at the time of the end of the simulation.
How can you explain the differences among the results?

b) To counteract the loss of links caused by movement or by fading effects, the transmission power of stations can be varied. The mechanism employed here tries to keep the number of neighbours of each station within a given interval [min_nb;max_nb].

Now conduct one of the simulations from a) with activated power control. Configure the module with values for min_nb and max_nb that you consider to be appropriate and produce the abovementioned plots for this simulation as well. Visualize also the distribution of transmission power at the time of the end of the simulation.

Available transmission powers are 10 mW, 16 mW, 25 mW, 40 mW, 63 mW, 100 mW.

How do the network properties change compared to the scenarios in which the stations are transmitting with constant power?

Practical Assignment 2: (Visualizing the scenarios)

In order to analyse the topology changes in a user friendly way, it is your task to develop a Java based visualization. The application should display the node positions and the links between them. The links should be updated according to the topology evolution over time.

Use the logfiles created during the simulation. If you need further information, e.g. node positions, adapt your simulation accordingly.

The GUI shall be implemented in Java with JDK 6 only. No additional libraries are permitted.

The following features of the GUI are mandatory:

- Configurable size of area (e.g. 500 m x 500 m)
- Scalable progression of time
- Display of nodes at their proper positions within the area and existing links in the course of time
- Marking of nodes which exceed the value of `max_nb` or fall below `min_nb` (e.g. using different colours)

It is required that your sources can be compiled using *ant*. Therefore, you will have to provide an appropriate build.xml file in the top-level directory of your final submission.
Information about using *ant* can be found on the following sites:

http://ant.apache.org/
http://wiki.apache.org/ant/FrontPage

Certainly you are encouraged to implement further features as you wish, e.g. display of the number of neighbour nodes, marking depending on level of transmission power, etc. These optional features should be activated only on demand, in order to keep the visualization concise.

A final note on solving the practical exercise:

- The described modules are <u>not</u> included in the version of ns-2 which can be downloaded from www.isi.edu. Therefore you will have to use the simulator version provided by us to solve the exercise.

Good luck!