IPP HURRAY!

www.hurray.isep.ipp.pt

Technical Report

Mobility Simulator

Paulo Baltarejo Sousa
Luís Lino Ferreira

HURRAY-TR-060403

Version: 2

Date: 15-04-2006

Mobility Simulator

Paulo Baltarejo SOUSA, Luís Lino FERREIRA

IPP-HURRAY!

Polytechnic Institute of Porto (ISEP-IPP)

Rua Dr. António Bernardino de Almeida, 431

4200-072 Porto

Portugal

Tel.: +351.22.8340509, Fax: +351.22.8340509

E-mail: {pbsousa,llf }@dei.isep.ipp.pt

http://www.hurray.isep.ipp.pt

1

Abstract

This technical report describes the Mobility Simulator that implements a simulation model of the station mobility and the radio wave propagation.

Document history:

Revision	Description	Date	By
1		15-04-2006	LLF
2		26-06-2007	LLF

Index

List of Figures

List of Tables

Mobility Simulator

1.Introduction

Every radio technology has a limited physical coverage area within which radio communications can be performed in acceptable conditions. The area characteristics depend on several aspects such as the dimensions and layout of obstacles, the existence of electromagnetic interference and the radio technology (including its antenna type) in use. Therefore, to cover wider areas it is necessary to divide the application area into several radio cells, each one operating at its own radio channel.

While moving wireless mobile stations will always try to use the best radio cell signal to communicate, therefore these stations will belong to different radio cells, also called domains within the context of this dissertation. The domain, to which a wireless mobile station belongs, depends on its location over time and on the signal strength in that location. In a simplified model the signal strength depends on the distance to the wireless cell base station.

There are several models to estimate the radio signal strength [1]. Simple models estimate radio signal strength between a transmitter and a receiver based solely on the distance between them. More sophisticated models use environment information, such as buildings, mountains, wall materials and location of obstacles. These models require a very detailed representation of the objects in the environment and are computationally very demanding.

This chapter describes the Mobility Simulator (MSim) which simulates the mobility of wireless mobile stations and the radio signal strength on the current wireless mobile station location. With this information it is possible to generate a vector containing the wireless domains to which the wireless mobile station belongs. That information is used by the Repeater-Based Hybrid Wired/Wireless PROFIBUS Network Simulator (RHW2PNetSim)[2] and Bridge-Based Hybrid Wired/Wireless PROFIBUS Network Simulator (BHW2PNetSim) [3]., in order to determine the points in time at which the stations must move to another domain. The results from this simulator are feed into the other simulators on the content of the `_location_vector` parameter.

The MSim has been developed using an open source high performance 3D graphics toolkit called OpenSceneGraph [4, 5] and the C++ programming language.

This chapter is structured as follows. A brief description of the simulation environment used to develop the MSim is presented in Section 2. A detailed description of the adopted radio signal model is given in Section 3. Section 4 presents a description of the simulation model of the MSim. The architecture of the MSim and the simulator configuration are presented in Section 5 and Section 6, respectively. The output data files generated by this simulator are presented and described in Section 7.

2.OpenSceneGraph

The OpenSceneGraph [5] is an open source cross platform graphical toolkit for the development of high performance graphical applications such as visual simulation, flight simulation games, virtual reality, scientific visualization and modelling. Based around the concept of scene graph (detailed below), it provides an object oriented framework on top of OpenGL freeing the developer from implementing and optimizing low level graphics calls, and provides many additional utilities for rapid development of graphics applications.

Written entirely in standard C++ and OpenGL, it makes full use of the Standard Template Library and it runs on all Windows platforms, OSX, GNU/Linux, IRIX, Solaris and FreeBSD operating systems.

2.1. Scene Graph Concept

The scene graph concept is tree-like. It starts with a top-most root node which encompasses the whole virtual world, be it 2D or 3D. The world is then broken down into a hierarchy of nodes representing spatial groupings of objects, object positions, object animations, or definitions of logical relationships between objects. The leaves of the graph represent the physical objects themselves, the drawable geometry and their material properties.

A scene graph is not a complete game or simulation engine, although it may be one of the main components of such an engine. Its primary focus is the representation of 3D worlds, and efficient rendering. Physics models, collision detection and audio are left to other development libraries that a user may integrate with it, like Open Dynamics Engine [6].

2.2. OpenGL

OpenGL [7] is a software interface to the graphical hardware. This API consists of about 150 distinct commands that developers use to specify the objects and operations needed to produce interactive 2D and 3D applications.

OpenGL is designed as a streamlined, hardware-independent interface to be implemented on many different hardware platforms. In order to achieve these qualities, no commands for performing windowing tasks or obtaining user input are included in OpenGL.

OpenGL does not provide high-level commands for describing models of 2D and 3D objects. With OpenGL, models must be built from a small set of geometric primitives (points, lines, and polygons).

3. Wireless Communications: Radio Signal Propagation

In wireless communication the information is delivered to the transmitter and modulated into radio waves. The radio wave is radiated through the air, using a radio channel, to the receiving antenna. At the receiving antenna, the radio wave is demodulated and the transmitted information is extracted (Figure 1).

In a wireless communication based on Base Station (BS), the coverage area is divided into small coverage areas (called cells) in which wireless communications are all relayed by a BS. In such systems, all stations transmit on one channel (uplink) and listen on a second channel (downlink). The BS functions are to receive a frame from one station through its uplink channel and retransmit the frame on its downlink channel.

Several aspects have influence in the quality of wireless communications. The transmission path between transmitter and receiver can vary from a simple line-of-sight to one severely obstructed by objects, like buildings, mountains and other surrounding objects. These objects can cause *reflection*, *diffraction* and *scattering* of the radio waves.

Reflection of the radio waves occurs when the radio wave impinges upon an object which has very large dimensions compared to the wavelength. Diffraction occurs when a radio wave encounters obstructions and propagates around the edges, corners and behind the obstruction causing secondary radio waves to form behind the obstruction. Scattering, results from rough surfaces whose dimensions are of the order of the wavelength, which causes the reflected energy to scatter in all directions.
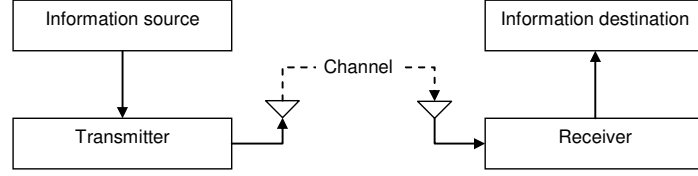
Figure 1 – Wireless communication model

Additionally, the objects could be perfect *dielectric*, perfect *conductor* or in-between. A dielectric object absorbs some of the radio wave energy, while the remainder is reflected back to the medium. A perfect dielectric object absorbs all radio wave energy without reflection. A perfect conductor reflects all radio wave energy. As a consequence, a radio wave breaks into several parts so that the received signal is a multiple delayed copy of the transmitted signal. This phenomenon is known as multipath propagation.

Radio wave quality depends on the type of antennas, the frequency and the bandwidth. On the other hand, the strength of a radio wave decreases as the distance between transmitter and receiver increases (fading) and a radio wave may be altered due to electromagnetic interference and noise.

Another aspect is, the Doppler spread. Doppler spread results when radio waves transmitted to or from a moving device undergo a shift in frequency if transmit-receive distance changes with time. The difference in frequency of the received signal and the transmitted signal is called the Doppler shift. In a multipath propagation the angles of arrival of the multipath components are different and each has a different Doppler shift.

Consequently, the wave propagation is very difficult to characterize, because it requires a very detailed representation of the objects in the environment, and is computationally very complex to treat.

In [8] several simulation models for radio wave propagation are presented. In [1] the author presents the main wireless communication principles and more specifically it also presents some models for radio wave propagation. According to this author the Log-normal Shadowing is a more general and widely-used model. In this models the power at the receiver ($P_r(d)$) can be calculated as follows:

$$P_r(d) = P_t + G_t - \overline{PL}(d) + G_r \qquad (1)$$

where P_t is the transmitted power, G_t is the transmitter antenna gain, G_r is the receiver antenna gain, d is the distance between transmitter and receiver in meters. $PL(d)$ is the average path loss at distance d between transmitter and receiver and is given by:

$$\overline{PL}(d) = PL_0(d_0) + 10n \log_{10}\left(\frac{d}{d_0}\right) + X_\sigma \qquad (2)$$

where n is the path loss exponent which indicates the rate at which the path loss increases with distance. X_σ is the shadowing term (the zero-mean Gaussian random variable in dB with standard deviation of σ). $PL_0(d_0)$ is the free-space path loss distance d_0 (d_0 is the close-in reference distance which is determined from measurements close to the transmitter) and is given by:

$$PL_0(d_0) = 20 \log_{10}\left(\frac{4\pi d_0}{\lambda}\right) \qquad (3)$$

where λ is the wavelength in meters and is related to the carrier frequency by:

$$\lambda = \frac{c}{f} \qquad (4)$$

where f is the carrier frequency in Hertz and c is the speed of the light ($3*10^8$m/s).

Table 1 and Table 2 list some typical path loss values exponents and the standard deviation on specific environments, respectively.

Table 1 – Path Loss Exponents for Different Environments. Source [9]

Environment		Path Loss Exponent, n(dB)
Outdoor	Free space	2
	Shadowed Urban area	2.7 to 5
In building	line-of-sight	1.6 to 1.8
	Obstructed	4 to 6

Table 2 – Standard Deviation for Different Environments. Source [9]

Environment	Standard deviation , σ (dB)
Outdoor	4 to 12
Office, hard partition	7
Office, soft partition	9.6
Factory, line-of-sight	3 to 6
Factory, obstructed	6.8

It is important to select a free space reference distance that is appropriate for the propagation environment. Usually, in large coverage cellular systems, a 1 km reference distance is commonly used, in microcellular systems, much smaller distances (such as 100 m or 1 m) are used. The reference distance should always be in the far field of the antenna so that near-field effects do not alter the reference path loss.

4.Simulation Model

This simulator models the radio signal strength using the Log-normal Shadowing model. Since, this model takes into account, not only, the distance between the transmitter and receiver, but also, the empirical characteristics of the environment.

The station mobility is calculated according to the wireless mobile station velocity and its path on the plant floor. In order to illustrate the simulation model, Figure 2 presents an example scenario.

The network comprises four domains, two wired domains (D^2 and D^4) and two wireless domains (D^1 and D^3). Three intermediate systems (IS1, IS2 and IS3) interconnect the wired and wireless domains. The network also comprises seven wired stations (S1, S2, S3, S4, S5, M1 and M2), three mobile wireless stations (M3, M4 and S6).

The wireless communications are relayed by two base stations (BSs), which are included in the ISs. Wireless mobile stations (M3, M4 and S6) move in a specific path, consequently the radio signal quality varies according to the propagation model. Further, in the mobility model of the wireless mobile station it is possible define stop points.

The handoff procedure is triggered in a periodic fashion and the BSs transmit a special purpose frame (the `Beacon` frame) during a pre-configured amount of time, which the wireless mobile stations use to assess the radio signal quality and change to the radio frequency of the BS with the strongest radio signal.
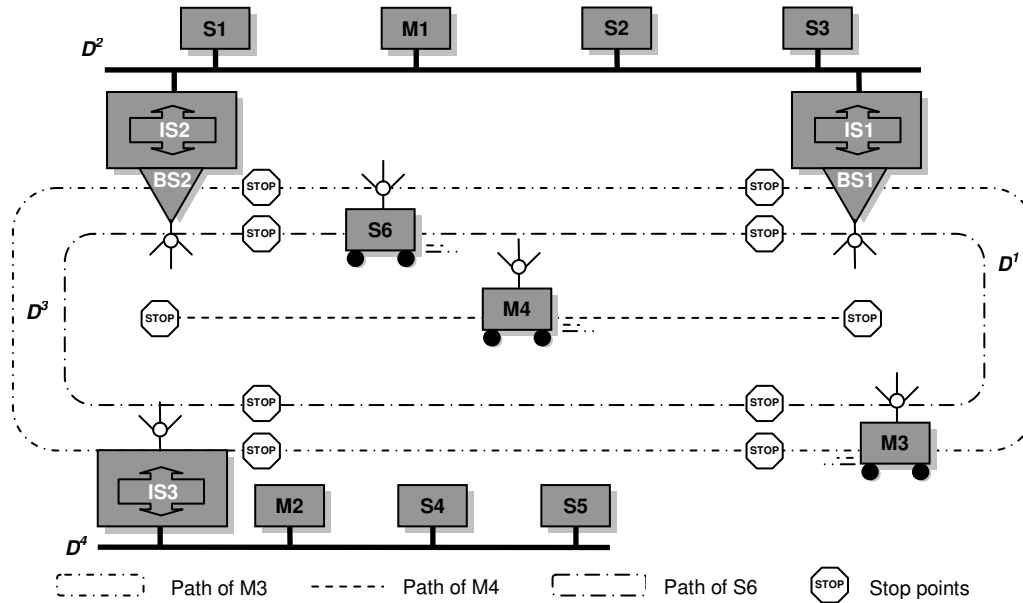
Figure 1 – Network scenario

5.Simulator Architecture

The following modules compose the MSim: `bs`, `agv`, `box`, `pc`, `antenna`, `camera`, `nhp` and `ground`. All these modules are implemented as C++ classes. There are two kinds of modules, the modules used for simulation and the modules used to compose the scenario. The first group includes the modules: `antenna` (which models a radio antenna), `bs` (which models a BS), `agv` (which models a wireless mobile station) and `nhp` (which models the handoff procedure). The second group includes: `box`, `pc` and `camera`. The `box` can be used to model a wall or a machine, the `pc` is used to model a personal computer or similar device. The `camera` is used on the `agv` module with esthetical purpose.

The main parameters (or attributes in the context of object-oriented programming) of the `antenna` module are: `frequency`; `Pt`; `Gt` and `Gr`. The `frequency` parameter contains the main radio frequency in use. `Pt`, `Gt` and `Gr` parameters are the transmitted power, the transmitter gain and the receiver gain of the antenna, respectively.

The `bs` module includes an instance of the `antenna` module. Each `bs` module instances is characterized by the parameters `name`, `domain_name`, `position` and `color`. The parameter `name` identifies a BS instance in the overall network, its value should be equal to the domain name (defined by `domain_name` parameter) in which it is operating. The `position` parameter specifies the `bs` module instance position and is used to calculate the distance between `bs` and `agv` module instances and the correspondent signal quality at the wireless mobile station location. The `color` parameter permits the assignment of a colour for easy visual identification of the wireless domain.

The `agv` module also includes an instance of the `antenna` module. The main parameters of the `agv` module are: `name`, `path`, `velocity` and `color`. The parameters `name` and `color` have the same purpose as in the `bs` module. The parameter `path` is used to assign a path to the `agv` module instance, using a list of points in Cartesian notation. Additionally, it is possible to define stop points by defining the stop duration together with path point coordinates.

To network handoff procedure is modelled by the `nhp` module. This module has the following parameters: `period`, `duration`, `exponent`, `std_dev` and `d0`. The `period` parameter is used to define the periodicity of the mobility procedure and the parameter `duration` the respective duration. The `exponent`, `std_dev` and `d0` are related to the Log-normal Shadowing model used in this simulator.

The values of path loss exponent, standard deviation and close-in reference distance are assigned to the parameters `exponent`, `std_dev` and `d0`, respectively.

Each simulation run has a predefined duration which is set in global simulation parameter called `sim_duration`.

The MSim is composed by two components, one is the front end and the other one is the simulator engine. The front end component can show two graphical 3D views of the simulation. It is useful to validate the simulation configuration. Figure 3 and Figure 4 present a screenshot of the simulator views.

In the view presented in Figure 3 the user is able to zoom and rotate of the simulation environment. The other view (Figure 4) shows a top level view of the simulator execution. This component allows checking the position and the radio frequency of `bs` instances, the path and the stop points of the `agv` instances. When these screenshot were taken, the `agv` instances M4 and S6 were operating at the same frequency as bs instance BS2 (at 2.3GHz) and `agv` instance M3 was operating at the same frequency of BS1 (at 2.4GHz). This means that M4 and S6 belong to domain D^3 and M3 belongs to domain D^3.
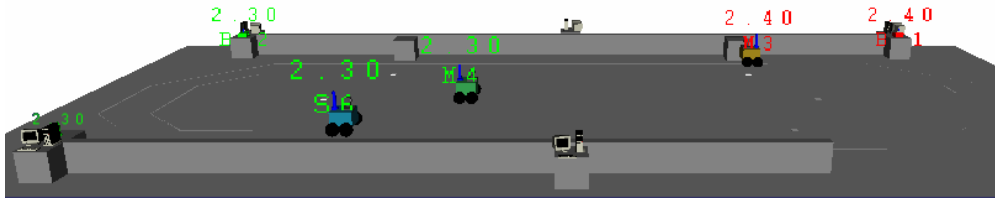
Figure 1 – View of the MSim front end component

Figure 2 – Top view of the MSim front end component

It is also possible to interact with this simulator using the keyboard. The 'Esc' key finishes the simulation run, The 'f' key allows increasing the simulation clock and in opposite the 's' key decreases the clock. The goal of this component is not to run a simulation, but to validate the simulation environment. The 'r' key allows running simulation runs in a background process (which was referred as the simulation engine module). Since this component does not have any graphics output, then the simulation runs can be performed in much less time.

6.Simulator Configuration

The simulation configuration is done through two text files. One of them specifies the parameter values of all components and the other specifies the seed number for the generation of random values used in the Log-normal Shadowing model. For each seed number a simulation run is done. Therefore, the number of simulation runs depends on the number of text lines.

Figure 5 presents part of configuration file. The duration of each simulation run (defined by the `sim.duration` parameter) is equal to two minutes. The network mobility procedure is triggered every 0.2 s (defined by the `nhp.period` parameter). The duration of each network handoff procedure is equal to 0.005 s (defined by the `nhp.duration` parameter).

The path loss exponent (defined by `nhp.exponent` parameter) is set to five and the standard deviation (defined by `nhp.std_dev` parameter) is set to 6. The close-in reference distance (defined by `nhp.d0` parameter) is set equal to one meter.

The simulated factory floor dimensions are 100 m per 200 m (defined by `ground.width` and `ground.length` parameters). The output directory for the simulation output files is called "output_files" (defined by the `output.dir` parameter).

```
sim.duration=2
nhp.period=0.2;
nhp.duration=0.005;
nhp.exponent=5
bnp.std_dev=6.8
bmp.d0=1
ground.width=100
ground.length=200
output.dir=../output_files;
```

Figure 1 – Configuration file related to global simulation parameters (excerpt)

Figure 6 presents part of a configuration file related to the `bs` module instance parameters. The system is composed by two `bs` instances (defined by the `sim.bs_num` parameter). One is called BS1 and the other is called BS2 (defined by the `bs[x].name` parameter). Wireless communications in domains D1 and D3 are done through BS1 and BS2, respectively (`bs[x].domain` parameter).

```
sim.bs_num=2;

bs[0].name=BS1;
bs[0].domain=D1;
bs[0].position=80.0,37,5.0,5.0;
bs[0].color=1.0,0.0,0.0,1.0;
bs[0].ant.freq=2.4;
bs[0].ant.pt=1;
bs[0].ant.gt=1;
bs[0].ant.gr=1;


bs[1]. name =BS2;
bs[1]. domain =D3;
bs[1].position=-80.0,37,5.0,5.0;
bs[1]. color =0.0,1.0,0.0,1.0;
bs[1]. ant.freq =2.3;
bs[1]. ant.pt =1;
bs[1]. ant.gt =1;
bs[1]. ant.gr =1;
```

Figure 2 – Configuration file related to the `bs` parameters (excerpt)

The definition of the `bs` module instance position (defined by `bs[x].position` parameter) is done by Cartesian coordinate (x,y,z) with origin at the centre of the ground. In this case, BS1 is located at position (50.0, 0.0, 5.0) and BS2 is located on the opposite side, at (-50, 0.0, 5.0). The colour of the `bs` module instances is defined by `bs[x].color` parameter and is set by four values (all in the range between 0.0 to 1.0). The first three are the RGB values and the fourth is the transparency.

Each `bs` module instance operates at different radio channel, BS1 operates on the 2.4GHz bandwidth and BS2 on the 2.3GHz bandwidth (defined by `bs[x].ant.freq` parameter). The transmitted power of each `antenna` module instance is defined by `bs[x].ant.pt` parameters, the

transmitter gain is defined by the `bs[x].ant.gt` parameters and the receiver gain is defined by `bs[x].ant.gr` parameters, which have been set equal to one in all `bs` module instances.

Figure 7 shows part of the configuration file related to an `agv` module. In this example there are two `agv` module instances (defined by `sim.agv_num` parameter). One is called M3 and the other S6 (defined by the `agv[x].name` parameter). Their velocity is set to 8 m/s and 6 m/s for M3 and S6, respectively.

The path and the stop points are defined by the `agv[x].path` parameter. This parameter is a string that is written using a pre-defined structure as follows. Each point of the path is defined using the coordinates x, y and z. The path is a set of points separated by colons. When a point is also a stop point, the stop time is defined after character '$'. For example, M3 starts at point (90, 10, 0), it follows in the direction of point (80, 20, 0), then goes to point (70, 30, 0). As soon as it arrives to the last point it follows to the next point in the path (40, 30, 0) where it stops for 0.125 seconds. After that, it follows to the next point (-40, 30, 0) and it again stops for 0.125 seconds. The path definition is cyclical, therefore when it reaches the last point it continues to the first.

The parameter `freq` of the `agv antenna` module instance changes during the simulation run, its value depends on the domain to which the `agv` module belongs to.

```
sim.agv_num=3;

agv[0].anme=M3;
agv[0].vel=8;
agv[0].path=90,10,0:80,20,0:70,30,0:40,30,0$0.125:-40,30,0$0.125:-70,30,0:-80,20,0:-90,10,0:-90,-10,0:-80,-20,0:-70,-30,0:-40,-
    30,0$0.125:40,-30,0$0.125:70,-30,0:80,-20,0:90,-10,0;
agv[0].color= 0.6,0.5,0.1,1.0;
agv[0].ant.freq=2.4;
agv[0].ant.pt=1;
agv[0].ant.gt=1;
agv[0].ant.gr=1;

agv[1].name=M4;
agv[1].vel=6;
agv[1].path= -50,0,0$0.250:0,0,0$0.250:50,0,0$0.25;
agv[1].color= 0.2,0.6,0.3,1.0;
agv[1].ant.freq=2.3;
agv[1].ant.pt=1;
agv[1].ant.gt=1;
agv[1].ant.gr=1;

agv[2].name=S6;
agv[2].vel=6;
agv[2].path=-80,-10,0:-70,-20,0:-40,-20,0$0.125:40,-20,0$0.125:70,-20,0:80,-10,0:80,10,0:70,20,0:40,20,0$0.125:-40,20,0$0.125:-
    70,20,0:-80,10,0;
agv[2].color= 0.1,0.5,0.6,1.0;
agv[2].ant.freq=2.3;
agv[2].ant.pt=1;
agv[2].ant.gt=1;
agv[2].ant.gr=1;
```

Figure 3 – Configuration file related to the `agv` parameters (excerpt)

7. Output Data Files

This simulator produces two kinds of output files. One contains information about the timings when the network handoff procedure was active, the domains to which the stations belong to and additional information (these files use the extension ".li"). The other type of file contains a text file with the information required to set the `_location_vector` parameter used in the RHW2PNetSim and BHW2PNetSim (these files use the extension ".sd").

Figure 8 presents part of the output file related to the `agv` module instance named M3 using the network configuration presented in Figure 2.

The first column (Time) refers the timestamp when the radio signal quality was evaluated. The second column (SHandProc) indicates the instant in time when the handoff procedure started and the third column (EHandProc) refers to the end of the handoff procedure. The radio frequency value, the BS name and domain name are in fourth (Freq), fifth (BS) and sixth (D) columns, respectively. The seventh column (Position(x,y,z)) contains data related to the `agv` position. The distance between the

`agv` instance and the `bs` instance with the best radio quality is presented in the eighth column (Dist). The signal power at the receiver location appears in the last column (Pr).

Time	SHandProc	EHandProc	Freq	BS	D	Position(x,y,z)	Dist	Pr
...								
0.200000	0.200000	0.205000	2.30	BS2	D3	-49.17,0.00,0.00	4 80	-104.40
0.400500	0.400000	0.405000	2.30	BS2	D3	-4 33,0.00,0.00	49.34	-120.04
0.600500	0.600000	0.605000	2.30	BS2	D3	-47.50,0.00,0.00	49.87	-117.47
0.800500	0.800000	0.805000	2.30	BS2	D3	-46.67,0.00,0.00	50.42	-127.38
1.000.000	1.000.000	1.005.000	2.30	BS2	D3	-45.84,0.00,0.00	50.97	-120.68
1.200.500	1.200.000	1.205.000	2.30	BS2	D3	-45.00,0.00,0.00	51.54	-120.72
1.400.500	1.400.000	1.405.000	2.30	BS2	D3	-44.17,0.00,0.00	52.11	-117.13
1.600.500	1.600.000	1.605.000	2.30	BS2	D3	-43.33,0.00,0.00	52.68	-12 74
1.800.500	1.800.000	1.805.000	2.30	BS2	D3	-42.50,0.00,0.00	53.27	-12 30
2.000.500	2.000.000	2.005.000	2.30	BS2	D3	-41.67,0.00,0.00	53.86	-115.78
2.200.500	2.200.000	2.205.000	2.30	BS2	D3	-40.84,0.00,0.00	54.45	-124.12
2.400.500	2.400.000	2.405.000	2.30	BS2	D3	-40.00,0.00,0.00	55.06	-123.01
2.600.500	2.600.000	2.605.000	2.30	BS2	D3	-39.17,0.00,0.00	55.66	-126.10
2.800.500	2.800.000	2.805.000	2.30	BS2	D3	-3 34,0.00,0.00	56.28	-11 07
0.200000	0.200000	0.205000	2.30	BS2	D3	-49.17,0.00,0.00	4 80	-104.40
0.400500	0.400000	0.405000	2.30	BS2	D3	-4 33,0.00,0.00	49.34	-120.04
0.600500	0.600000	0.605000	2.30	BS2	D3	-47.50,0.00,0.00	49.87	-117.47
0.800500	0.800000	0.805000	2.30	BS2	D3	-46.67,0.00,0.00	50.42	-127.38
...								

Figure 1 – Output file of wireless mobile station M3 (excerpt)

Figure 9 presents part of the output file related to the `agv` module instance M3, which contains the domain sequence to which a wireless mobile station belongs to during the simulation runs. In this file the information is organized as tuples separated by a colon. The first element indicates during how many handoff procedures the `agv` module instance stays in the domain referred in the second element. In this example, the `agv` module instance M3 stays in domain D1 for 43 network handoff procedures, and then it changes to the domain D3 where it stays for another 10 handoff procedures.

The information contained in this file can be assigned to the `_location_vector` parameter (see Section 6.2.3 for details) which determines to which domain a `Master` or a `Slave` module instance belongs to.

```
43,D1:10,D3:29,D1:1,D3:12,D1:2,D3:3,D1:3,D3:3,D1:
```

Figure 2 – Output location file of wireless mobile station M3 (excerpt)

References

[1]   T. S. Rappaport, "Wireless Communications. Principles and Practice": Prentice Hall, 1996.
[2]   P. Sousa and L. Ferreira, "Repeater-Based Hybrid Wired/Wireless PROFIBUS Network Simulator," Polytechnic Institute of Porto., Porto, Technical-Report Hurray-tr-060402, April 2006.
[3]   P. Sousa and L. Ferreira, "Bridge-Based Hybrid Wired/Wireless PROFIBUS Network Simulator," Polytechnic Institute of Porto, Porto, Technical-Report Hurray-tr-050403, April 2005.
[4]   "A Short Introduction to the Basic Principles of the Open Scene Graph". http:// www.openscenegraph.org/.
[5]   "OpenSceneGraph". http:// www.openscenegraph.org.
[6]   "Open Dynamics Engine V0.5 User Guide". http:// www.ode.org.
[7]   D. Shreiner, et al., "OpenGL(R) Programming Guide: The Official Guide to Learning OpenGL(R), Version 2", 5 ed: Addison-Wesley Professional, 2005.
[8]   W. H. Tranter, et al., "Principles of Communication Systems Simulation with Wireless Applications": Prentice Hall, 2003.
[9]   "SimPy Manual". http:// simpy.sourceforge.net/SimPyDocs/Manual.html.