BonnMotion - A Mobility Scenario Generation and Analysis Tool

Nils Aschenbruck, Raphael Ernst, Elmar Gerhards-Padilla, Matthias Schwamborn
University of Bonn - Institute of Computer Science 4
Roemerstr. 164, 53117 Bonn, Germany
{aschenbruck,ernst,padilla,schwamborn}@cs.uni-bonn.de

ABSTRACT

Simulation and emulation are techniques frequently used for performance evaluation of wireless multi-hop networks. If the wireless devices are mobile, the movement patterns of these objects are found to have significant impact on the simulation and emulation results. This is quite obvious as the movements influence the topology of the network.

In this paper we describe and present BonnMotion. BonnMotion is an open-source Java software which creates and analyzes mobility scenarios. It has been developed at the University of Bonn, Germany, where it serves as a tool for the investigation of mobile multi-hop network scenario characteristics. The scenarios can also be exported for the network simulators ns-2, GloMoSim/QualNet, COOJA, and MiXiM.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless Communication*; I.6.5 [**Simulation and Modeling**]: Model Development

General Terms

Performance, Reliability

Keywords

Mobility Modeling, Performance Evaluation, Motion Generator

1. INTRODUCTION

For the performance evaluation of algorithms, protocols as well as communication systems, simulations and emulations are frequently used. Compared to a testbed implementation, simulation and emulation show advantages concerning scalability, reproducibility, and cost-efficiency. Modeling the movements of the nodes is found to have significant impact on the results of simulative and emulative performance evaluation. If the wireless devices are mobile, this is quite obvious as the movements influence the topology of the network. Due to this reason, many different mobility models were proposed during the last decade.

From the point of a user who wants to conduct simulative performance analysis, models are great, but tools that provide traces are even better. In this paper we describe and present BonnMotion. BonnMotion is an open-source Java software which creates and analyzes mobility scenarios. It has been developed at the University of Bonn, Germany, where it serves as a tool for the investigation of mobile multi-hop network scenario characteristics. The tool is available at: `http://bonnmotion.net.cs.uni-bonn.de/`. The scenarios can also be exported for the network simulators ns-2, GloMoSim/QualNet, COOJA, and MiXiM. The goal of this paper is to introduce the tool BonnMotion and provide some sample analyses.

The remaining parts of this paper are structured as follows: First of all, we survey the related work (section 2). Then, we introduce the tool BonnMotion and describe its architecture (section 3). Next, we describe the models provided and explain the usage of the different models within BonnMotion (section 4). In section 5 we provide an overview of the different export formats and simulators supported by BonnMotion. Thereafter, we introduce BonnMotion's capabilities to analyze scenarios generated (section 6). In this section we also show sample results for analyses concerning different metrics. Finally, we conclude the paper and point out topics for future work (section 7).

2. RELATED WORK

Broadly, mobility models are classified in three categories: microscopic, mesoscopic, and macroscopic mobility models. A microscopic model describes the movements of the individual nodes. Typically, location, velocity, and acceleration of the individual nodes are modeled over time. A macroscopic model abstracts the individual movements and just models the parameters relevant to the system being evaluated. A typical example for this is the impact of the movement on a specific region (e.g. cell). By doing so, abstract location and time-dependent metrics such as cell-change-rate or handover-traffic are considered. Mesoscopic models aggregate the movements of the different nodes.

A macroscopic model is appropriate if the impact of the movements on the communication system is sufficiently modeled by abstract metrics like cell-change-rate. A microscopic model is needed if the movements of the individual nodes have a decisive impact on the communication sys-

SIMUTools 2010 March 15–19, Torremolinos, Malaga, Spain.

Digital Object Identifier: 10.4108/ICST.SIMUTOOLS2010.8684
http://dx.doi.org/10.4108/ICST.SIMUTOOLS2010.8684

tem. Recently, communication systems that at least contain multi-hop components (e.g. Mobile Ad-hoc NETworks (MANETs) or Mesh-Networks) are studied excessively. For the performance evaluation of these systems, microscopic models are needed. In this paper we focus on microscopic mobility models.

In the past there have been several general surveys [3, 5, 12, 44, 48] as well as some specific ones for vehicular [20, 22] and tactical [1] modeling.

Overall, various mobility models have been proposed so far. Few of these yield specific scenarios while the others are more generic. The generic ones are easier to use and often allow theoretical analysis. Based on the dependencies, the models are distinguished, analogous to [3], by the following three categories:

- temporal dependencies: Actual movement of a node is influenced by its movement in the past.
- spatial dependencies: Movement of a node is influenced by the surrounding nodes (e.g. group mobility).
- geographic restrictions: The area in which the node is allowed to move is restricted.

There are models such as Random-Waypoint or Random-Walk that do not exhibit any of these dependencies. While other models which are more realistic or scenario-specific may realize different kind of dependencies.

Instead of classifying the models after their dependencies, the applications or scenarios are of more interest for the users and may be used as a criterion for classification. This classification may seem to be the more intuitive one. But a model created for one scenario may be re-parameterized for another one. Table 1 provides a comprehensive survey of the existing synthetic mobility models.

We classified the models according to their dependencies as well as scenarios described in the respective papers. As scenarios we consider: Campus, Conference, Pop Concert / Fair, City / Urban, Vehicular, Public Safety, Battlefield, as well as Daily Movement.

As can be seen in table 1, for nearly all scenarios and applications different models have been proposed. However, for most of these models there are neither implementations nor synthetic traces public available. Thus, from a user's point of view there are interesting models, but they are not usable. Furthermore, there is a demand for easy-to-use tools that enable a user to analyze and compare characteristics of synthetic traces generated. There are other tools and codebases available: Toilers-Code-Base [12], Important [4], MobiSim [41], CanuMobiSim [52, 53], and SUMO [31]. However, in our opinion our tool BonnMotion is easier to use and extend as well as more exhaustive concerning models and analysis options. Thus, in the next section we introduce BonnMotion which can be used for generation as well as analysis of mobility scenarios.

3. BONNMOTION

As mentioned before, BonnMotion is an open-source Java software. To use this software, you need to have a JDK or JRE installed. During the installation, a few other shell scripts / batch files are created. "bm" starts the BonnMotion application. Starting it without command line parameters prints a detailed help message.

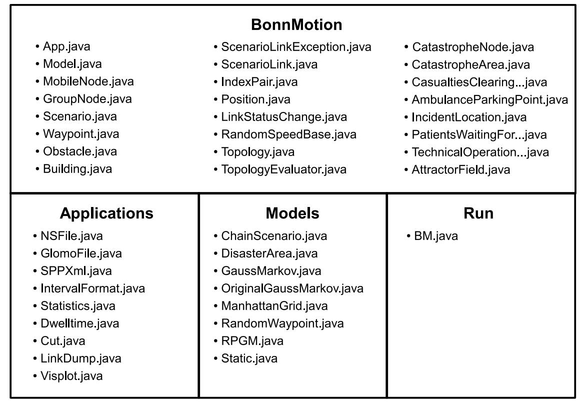


Figure 1: Architecture of BonnMotion

Fig. 1 shows the general architecture of BonnMotion. There are four packages:

- BonnMotion general basics
- Models
- Applications
- Run (start module)

The *general basics* contain components that are used by different models. There are generic classes such as *Model*, *Position*, *RandomSpeedBase*, and *MobileNode* as well as more specific ones like *AttractorField*. Each model has to implement the class *Model*. All models belong to the *Models* package. The *Applications* package contains little programs that work with the generated movement traces. There are exporters for different simulators as well as statistic applications. The *start module* is a wrapper that starts the BonnMotion application.

Currently, there are several mobility models available, which are introduced in the next section. There are two possibilities to feed input parameters into the scenario generation: The first is to enter the parameters on the command line, and the second is to have a file containing the parameters. These two methods can also be combined; in this case, the command line parameters override those given in the input file.

The scenario generator writes all parameters used to create a certain scenario to a file. In this way, settings are saved and particular scenario parameters can be varied without the need to re-enter all other parameters. Important parameters used with all models are the following: The *node number* is set with $-n$, the *scenario duration* (in seconds) with $-d$ and the $-i$ parameter specifies how many additional seconds at the beginning of the scenario should be skipped. With $-x$ and $-y$, the *width* and *height* (in meters) of the simulation area are set. With $-R$, the *random seed* can be set manually.

Cutting off the initial phase is an important feature and therefore, -i has a high default value: It has been observed for different models that the topology is not in steady state. In Random-Waypoint's steady state, nodes have a higher probability of being near the center of the simulation area, while they are initially uniformly distributed over

Digital Object Identifier: 10.4108/ICST.SIMUTOOLS2010.8684
http://dx.doi.org/10.4108/ICST.SIMUTOOLS2010.8684

Model		temporary	spatial	geographical	Campus	Conference	Pop Concert / Fair	City / Urban	Vehicular	Public Safety	Battlefield	Daily Movement
		Dependencies			**Applications / Scenarios**							
No dependencies												
Random-Waypoint	[28]											
Random-Direction	[51]											
Modified Random-Direction	[51]											
Random-Walk	[12]											
Random-Border-Model	[8]											
Random-Waypoint with attraction points	[8]						√					
Clustered-Mobility	[38]		(√)							√		
Disaster-Recovery	[46]		(√)							√		
General Ripple	[13]											
Temporary dependencies												
Gauss-Markov	[35]	√										
Smooth-Random	[6]	√										
Spatial dependencies												
Reference-Point-Group	[21]	(√)	√	(√)								
Structured-Group	[10]		√							(√)	(√)	
Virtual Track	[59]		√					√			√	
Social-Network-founded	[42]		√									
Mold	[36]		√		√							
Community-based	[43]		√									
Geographic restrictions												
Manhattan-Grid	[17]			√				√				
Graph-based	[54]			√				√				
Obstacle	[26]			√	√							
Weighted-Waypoint	[23]			√	√							
Voronoi	[60]			√				√				
Area-Graph-based	[9]			√	√							
CosMos	[19]			√								
Hotspot	[39]			√			√					
Route	[39]			√				√				
Random-Waypoint-City	[30]			√				√	√			
Agenda Based	[58]			√				√				√
Graph-Random-Waypoint	[40]			√				√	√			
Graph-Random-Walk	[40]			√				√	√			
Subway	[55]			√				(√)				
Hybrid dependencies/restrictions												
Freeway	[4]	√	√	√					√			
User-oriented-Meta-Model	[53]	√	√	√								√
Street-Random-Waypoint	[14]	√	√	√				√	√			
VanetMobiSim	[20]	√	√	√				√	√			
Hostage-Rescue	[25]		√	√							√	
Disaster-Area-Model	[2]		√	√						√		
CORPS	[24]		√	√						√		
Platoon	[49]		√	√							√	
Working-Day-Model	[16]	√	√	√				√				√

Table 1: Survey on existing synthetic mobility models (√:= yes or belongs to; (√) := more than a no, but less than a yes, e.g. a subway is part of a city, but it is no city scenario)

Digital Object Identifier: 10.4108/ICST.SIMUTOOLS2010.8684
http://dx.doi.org/10.4108/ICST.SIMUTOOLS2010.8684

the simulation area. Further details may be found in several studies (e.g. [8], [56], [34]) that analyze the Random-Waypoint model with respect to implicit (unwanted) assumptions and characteristics. Beside Random-Waypoint, many other models are affected in a similar way. In our implementation of the Manhattan Grid model, all nodes start at (0,0) for simplicity. Thus, an initial phase is needed till the distribution is in steady state.

Using BonnMotion is quite simple. The usage example:
bm -f scenario1 RandomWaypoint -n 100 -d 900 -i 3600
creates a Random-Waypoint scenario with 100 nodes and a duration of 900 seconds. An initial phase of 3600 seconds is cut off.

A scenario is saved in two files: The first, with the suffix ".params", contains the complete set of parameters used for the simulation. The second, with the suffix ".movements.gz", contains the (gzipped) movement data. This movement data can be exported into input formats for different simulation tools. Details on data export are described in section 5.

4. MODELS CURRENTLY SUPPORTED BY BONNMOTION

Currently, there are five mobility models publicly available: Random-Waypoint, Gauss Markov, Manhattan Grid, Reference-Point-Group-Mobility (RPGM) and the Disaster Area model. The following subsections will describe the models as well as their realization and usage within BonnMotion. Furthermore, BonnMotion supports the generation of static scenarios.

4.1 The Random-Waypoint model

The *Random-Waypoint* model is a simple stochastic model in which a node perpetually chooses destinations (waypoints) and moves towards them. In the original model [28] the nodes are distributed randomly over the simulation area. After waiting for a constant pause time, each node chooses a waypoint and moves towards it with a speed chosen from an interval $[v_{min}; v_{max}]$. After arriving at the waypoint, the node again waits for a constant pause time and chooses the next waypoint. In [50] it is proposed to also choose the pause time from an interval $[p_{min}; p_{max}]$. The different random variates are generally chosen uniformly distributed. However, arbitrary distributions are possible as well.

In the last years, there were several studies that analyzed the Random-Waypoint model with respect to implicit (unwanted) assumptions and characteristics. As the nodes are initially distributed randomly, it takes some time until the nodes reach a stationary distribution (cf. [45]). Thus, a long enough initial period should be discarded. In [56] it is shown that the average velocity is decreasing over simulation time if $v_{min} = 0$. Thus, $v_{min} > 0$ and $p_{max} < \infty$ should be chosen. Furthermore, in several publications it was shown that the nodes cumulate in the middle of the simulation area (cf. [8], [11], [7]).

A distribution and movement of the nodes across the entire simulation area does not fit to the characteristics of most realistic movements. There are extensions (e.g. [8]) which add attraction points to this model in order to generate more realistic non-equally distributed mobility: The probability that a node selects an attraction point or a point in an attraction area as next waypoint is larger than the choice of other points. The nodes visit some points more frequently than others.

The implementation in BonnMotion supports choosing an initial period by using the $-i$ option. Furthermore, instead of choosing new destinations uniformly distributed from the simulation area, *attraction points* can be defined with the $-a$ parameter, followed by the data characterizing the attraction points. Each attraction point is defined by four floating point numbers: <x-coordinate>, <y-coordinate>, <intensity>, and <standard deviation>. The coordinates give the attraction point's position. The intensity levels weight the attraction points: A point with an intensity x times as high as another point's will also attract a node with a probability which is x times as high. The last parameter is the standard deviation of the Gaussian distribution with mean 0 that is used to determine the nodes' distances to the attraction point on each of the two dimensions. Several attraction points can be simply defined by concatenation.

4.2 The Manhattan Grid model

In the context of the UMTS standardization, the so-called *Manhattan-Grid* model was specified [17]. The simulation area is divided into squared blocks. Nodes are modeled as pedestrians moving on the vertices of the squares (streets). Initially the nodes are randomly distributed on the streets. Each node chooses a direction and a velocity. If a node reaches a corner, the node changes direction with a certain probability. The velocity is changed over time. In this model, nodes move only on predefined paths.

In BonnMotion the arguments $-u$ and $-v$ set the number of blocks between the paths. As an example, $-u3-v2$ places the following paths on the simulation area:

```
+ - + - + - +
|   |   |   |
+ - + - + - +
|   |   |   |
+ - + - + - +
```

Our implementation contains some (reasonable) modifications of the Manhattan Grid model:

An additional parameter we introduce is the minimum speed of a mobile node. This is helpful because the speed of a mobile can be arbitrarily close to 0 and since the model defines that the speed is to be updated in *distance* intervals, there can be very long periods of very slow node movement without this parameter.

The possibility to have nodes pause was added with help of two additional parameters: The pause probability (if a node does not change its speed, it will pause with that probability) and the maximum pause time.

4.3 Gauss-Markov models

Using the Random-Waypoint model, the nodes suddenly may change speed or direction. This is quite unrealistic considering aspects like acceleration and deceleration. In the *Gauss-Markov* model [35] velocity and direction of the future (time interval $t+1$) depend on the current values (time interval t). Initially, for each node, position, velocity, and direction are chosen uniformly distributed. The movement of each node is variated after an interval Δt. The new values are chosen based on a first-order autoregressive process. Further details can be found in [35]. BonnMotion supports two variants of this model:

The *original* Gauss-Markov model ("OriginalGauss-Markov") strictly follows the publication [35]. In this im-

Digital Object Identifier: 10.4108/ICST.SIMUTOOLS2010.8684
http://dx.doi.org/10.4108/ICST.SIMUTOOLS2010.8684

plementation, the mean velocity vector μ is not specified directly; instead, the norm is specified using $-a$ and a random vector with this norm is assigned to each station. Of course, a norm of 0 yields only the vector (0,0). The implementation also allows the user to specify a maximum speed. A velocity vector with a larger norm will be multiplied by an appropriate scalar to reduce the speed to the maximum speed. The model has been adapted to deal with scenario borders in the following way: If a station moves onto the border, its velocity vector as well as its expected velocity vector are "mirrored".

The implementation "GaussMarkov" follows the description in [12]. The main commonalities are that for each mobile node, two separate values are maintained instead of one speed vector: The speed of the node and its direction of movement. Also the default method of handling mobile nodes that move out of the simulation area is closely related to [12]: Nodes may continue to walk beyond the area boundary, which causes the next movement vector update not to be based on the prior angle, but on an angle that brings the nodes back onto the field. Therefore, the field size is automatically adapted to the node movements after scenario generation.

The main difference to [35] is that new speed and direction of movement are simply chosen from a normal distribution with a mean of the respective old value (the standard deviation is specified on the command line using $-a$ and $-s$). Speed values are constrained to a certain interval that can be specified on the command line using $-m$ and $-h$: If a newly chosen speed value is outside of this interval, it is changed to the closest value inside of the interval (which is either the minimum or the maximum value).

The behavior described above can be modified with several command line switches: Using $-b$, the size of the simulation area is fixed and nodes simply "bounce" at the area boundaries. Using $-u$, the speed values outside of the valid speed interval are adapted in a way that leads to a uniform distribution of node speeds (instead of peaks around the interval boundaries).

4.4 The Reference Point Group Mobility model

The *Reference-Point-Group-Mobility* model (RPGM) [21] models the movement of groups of nodes. The movement of the groups is modeled according to an arbitrary mobility model. The movement of the nodes inside a group is realized using a reference point for each node. The actual position of a node is a random movement vector added to the position of its reference point. The absolute positions of the reference points change according to the arbitrary mobility model, but the relative positions of the reference points inside a group do not change. Hence, the spatial dependence is realized using the reference points.

The BonnMotion implementation of this model uses Random-Waypoint for modeling the movements of the reference points. It includes the optional possibility to have "dynamic" groups: When a node comes into the area of another group, it changes to this new group with a probability that can be set with $-c$ <probability>. Note that when this feature is activated, "empty" groups may be moving along the simulation area and nodes coming into their areas may change their memberships to these.

4.5 Disaster Area model

In [2] a quite complex model which realistically represents the movements in a disaster area scenario is provided. This model supports heterogeneous area-based movement on optimal paths avoiding obstacles with joining/leaving of nodes as well as group mobility.

To realize area-based movement, the simulation area is divided into polygonal tactical areas. The tactical areas are classified according to the civil-protection concept *separation of room*. Each node is assigned to one of these tactical areas. For some areas there are both stationary nodes, which stay in the distinct area moving according to a random based mobility model, as well as transport nodes that carry the patients to the next area following a movement cycle. Different areas and classes allow heterogeneous speeds. The area and the class (stationary or transport) the node belongs to define the movement of the node as well as the minimal and maximal speed distinguishing pedestrians from vehicles.

The optimal path for the movement of the transport units between the different areas is determined by methods of robot motion planning. For finding the shortest paths and avoiding obstacles between the tactical areas, visibility graphs are used. A visibility graph is a graph whose vertices are the vertices of the polygons. There is an edge between two vertices, if the vertices can "see" each other - meaning the edge does not intersect the interior of any other obstacle. The shortest path between two points consists of an appropriate subset of the edges of the visibility graph. Thus, after having calculated the visibility graph containing all possible shortest paths between the areas avoiding obstacles, the direct path between two areas for each transport unit can be calculated.

Vehicular transport units (e.g. ambulances) typically leave the disaster area to carry patients to a hospital. Thus, joining and leaving nodes are realized using specific entry and exit points (registration areas).

Group mobility is realized as an optional characteristic for disaster areas, as in civil-protection there may only be one device for each group. Nevertheless, it is realized similar to RPGM [21] using reference points. The units of each area are grouped. The size of the group depends on the type of the area and the group. Similar to RPGM, the nodes follow their reference point. The movement of each node in a group is calculated in relation to the movement of the reference point.

Using BonnMotion, tactical areas can be defined with the $-b$ parameter. All other dependencies can be modeled as well. For further details please have a look at the sample configuration provided with BonnMotion.

4.6 Static scenarios

Before performing evaluations in complex mobile scenarios, it is often necessary to evaluate basic properties in static scenarios. BonnMotion supports the generation of different kind of static topologies. By default, nodes in static scenarios are homogeneously distributed over the simulation area. Furthermore, there are two possibilities for non-homogeneous node distributions: Attraction points can be defined that yield a higher number of nodes near these points. The implementation is similar to the one described above for Random-Waypoint. The simulation area can be divided into several areas with different node densi-

Digital Object Identifier: 10.4108/ICST.SIMUTOOLS2010.8684
http://dx.doi.org/10.4108/ICST.SIMUTOOLS2010.8684

ties along its x-axis. Given the number n of density levels, each of the n areas will contain a fraction of approximately $2 * k/(n * (n + 1)), 1 \leq k \leq n$, of the nodes. The density decreases from left to right.

5. EXPORT FORMATS AND SUPPORTED SIMULATION TOOLS

The native format in which BonnMotion saves the movement traces is node-by-line waypoint based. This means that there is one line for each node. This line contains all the waypoints. A waypoint is a position at which the movement of a node (e.g. direction, velocity) changes. A waypoint consists of:

- the simulation time in seconds at which the waypoint is reached by the node
- the x and y coordinates of the position of the waypoint

Beside its native format, BonnMotion contains different export applications for several simulators.

5.1 ns-2

The network simulator ns-2 [18] is one of the simulators most frequently used for simulative network performance evaluations. BonnMotion's NSFile application is used to generate two files that can be integrated into a TCL script to start an ns-2 simulation via the ns-2 *source* command.

The file with the suffix ".ns_params" sets some variables needed to set up the simulation. The file with the suffix ".ns_movements" schedules the movements of the node objects that are expected to be in an array named "node_", numbered starting at 0. The simulator object is expected to be in the variable "ns_".

Note that the "NSFile" application places an additional margin around the simulation area, because different ns-2 versions may crash when nodes move at the border of the simulation area.

5.2 Glomosim / Qualnet

Glomosim [57] is another network simulator for wireless and wired network systems. BonnMotion supports Glomosim by its GlomoFile application. This application creates files with the suffixes ".glomo_nodes" and ".glomo_mobility", which can be used with Glomosim (2.0.3) and Qualnet (3.5.1). For Qualnet, the $-q$ switch has to be used. This causes nodes to be numbered starting at 1, not at 0.

5.3 Interval format

The native BonnMotion format implies that during the simulation, for each event, the current node positions have to be calculated based on the waypoints. If there are many events, this may have a negative impact on the runtime of a simulation. An alternative is to use an interval based approach. The nodes are regarded as stationary for an interval. The positions of the nodes are updated periodically after each interval by a specific position update event. By doing so, the current node positions do not have to be calculated for each event. However, the number of events is increased, which may also influence the runtime of a simulation negatively. A factor that has a major impact in this context is the interval length. Smaller intervals yield higher accuracy but also more events. Overall, it is a trade-off between the number of events and the runtime per event.

BonnMotion's IntervalFormat application supports interval based trace format. The interval length can be specified using the -I option. The default value is one second. The interval trace format is an interval-by-line based. This means that there is one line for each interval of each node. A line consists of:

- the node number
- the simulation time in seconds (in intervals)
- the x and y coordinates of the position of the node for the interval

The IntervalFormat application prints the waypoints (ordered by node and time) for every interval step.

The interval based trace format is used by COOJA and MiXiM. COOJA [47] is a Java-based sensor network simulator originally designed to simulate networks of nodes running the Contiki operating system. MiXiM [29] is an integration effort that combines several OMNeT++ simulators. One of them is the Mobility Framework (MF) [15], the others are a MAC simulator [32], Positif (a simulator for localization) [33] and ChSim (a simulator targeting radio propagation models) [37].

6. SCENARIO ANALYSIS

For the analysis of the generated scenarios, different metrics can be calculated. BonnMotion supports (protocol independent) metrics of two classes: (1) *pure movement metrics* and (2) *link based metrics*. As pure movement metrics, velocity, relative mobility, and dwell time are supported. Link based metrics depend on links between two nodes. A link between two nodes a and b exists if a is inside the communication range of b and if a and b are switched on. Whether a link exists or not depends on the propagation model assumed. In BonnMotion, all calculation base on a circular range around each node. As link based metrics, BonnMotion supports link duration, time to link break, node degree, partitions, and k-connectivity. For these metrics only symmetric (bi-directional) links are considered. There are different applications to support these statistics. The *Statistics* application can be used to calculate *overall* statistics (averaged over the simulation time) and as *progressive* statistics (values of metrics for certain points in time). Furthermore, there are applications for specific metrics such as *Dwelltime* and *LinkDump* applications. In this section we define the metrics and show examples for different ones. For the examples, we generated Random-Waypoint scenarios. We chose 150 nodes on a 350m × 200m simulation area. We considered speed ranges of pedestrians (1-2m/s) and of vehicles (5-12m/s). Furthermore, we considered attraction points with different attraction grades: $p \in \{0.0, 0.3, 0.6, 0.95, 0.99\}$. For each parameter set, 20 traces were generated.

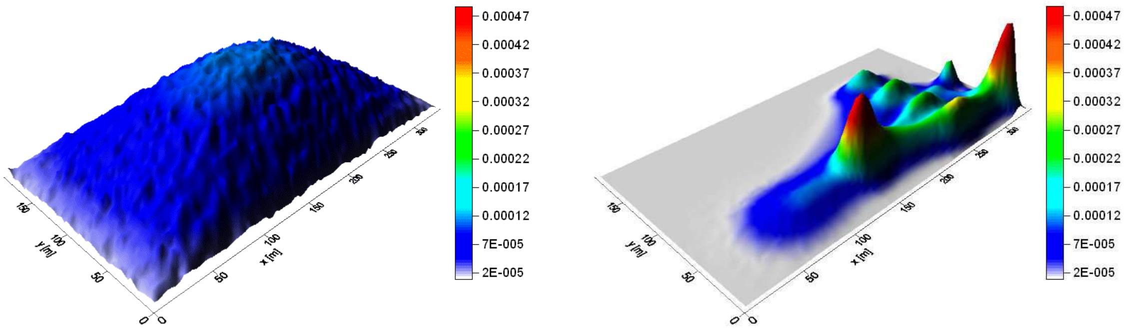
6.1 Velocity and Relative Mobility

A simple but nevertheless useful metric is the velocity.

$$V_n = \frac{1}{n} \sum_{i=1}^{n} \frac{d_i}{t}$$

where n is the number of nodes, d_i the distance a node i moved during the time t.

Digital Object Identifier: 10.4108/ICST.SIMUTOOLS2010.8684
http://dx.doi.org/10.4108/ICST.SIMUTOOLS2010.8684



(a) $RWP_{\mathrm{car.0}}$ (b) $RWP_{\mathrm{car.95}}$

Figure 2: Distribution of nodes - histogram over simulation area visualizing the dwell time

The *relative mobility* is independent of the transmission range and is calculated according to [27].

$$M = \frac{1}{|x,y|}\sum_{x,y} M_{xy} = \frac{2}{n(n-1)}\sum_{x=1}^{n}\sum_{y=x+1}^{n} M_{xy}$$

$$M_{xy} = \frac{1}{T}\int_{t_0}^{t_0+T} |v(x,y,t)|dt$$

$$v(x,y,t) = \frac{d}{dt}(l(x,t) - l(y,t))$$

where T is the observed duration and $l(x,t)$ is the location of node x at time t. The relative mobility M is the average relative velocity of two nodes averaged over all pairs of nodes.

6.2 Dwelltime

As a visual measure for the distribution of the nodes in the simulation area, the dwell time is used. The simulation area is divided into small square cells. Based on the *dwell time* of a node in each cell a two-dimensional histogram over all nodes is created (cf. [8]). The Dwelltime application can be used to calculate the dwell time. Figure 2 shows two histograms for two Random-Waypoint variants aggregated over the 20 traces of each candidate. The impact of the attraction points compared to the classic Random-Waypoint distribution is obvious.

6.3 Link Duration and Time to Link Break

The *link duration* and *time to link break* are metrics that directly base on the links. In contrast to the link duration, for the time to link break only links that break are counted. Perpetual links are not considered. This metric shows how often links and - based on this - routes break. For both metrics, average values and standard deviation over the time and nodes can be calculated. Figure 3 shows the average time to link break for different Random-Waypoint variants. It can be seen that higher speed and attraction grades yield more link breaks. This is quite obvious as higher speeds result in more movements and higher attraction grades cause sparser networks.

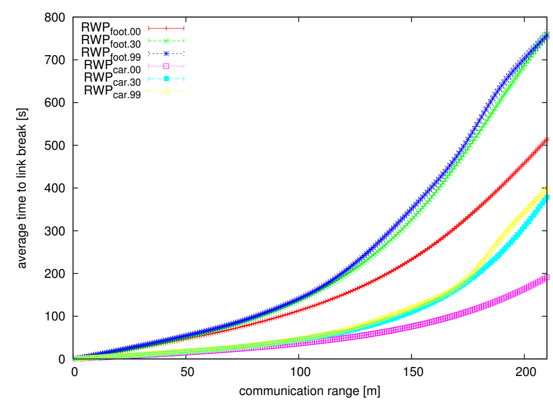


Figure 3: Average time to link break (average and 0.95 confidence interval)

6.4 Node Degree, Partitions, and K-Connectivity

Based on the links, it is also possible to create a connectivity graph $G = (V, E)$. For each node, there is a vertex in the graph. There is an edge between two vertices, if there is a link. The connectivity of the graph can be analyzed at discrete points in time. Metrics that can be calculated are: *node degree*, *number of partitions*, *partitioning degree*, and *k-connectivity*. The *node degree* shows to how many nodes one node is connected. This metric can be calculated on the average over the time. It is a measure for the node density. Figure 4 shows average node degree distribution density functions for the different models for a fixed communication range of 100m. The unimodal densities are quite obvious. They are caused by Random-Waypoint nodes moving over the whole simulation area. Higher attraction grades yield larger node degrees as the nodes tend to stay around the attraction points with higher numbers of neighbors.

The number of partitions is the number of non-connected network parts. If the average number of partitions is one, it means the network is connected at all times. Values larger than one indicate that this is not the case. The *k-connectivity* is the smallest amount of vertices to be removed that yield a partitioned or trivial graph. It is a measure for

Digital Object Identifier: 10.4108/ICST.SIMUTOOLS2010.8684
http://dx.doi.org/10.4108/ICST.SIMUTOOLS2010.8684

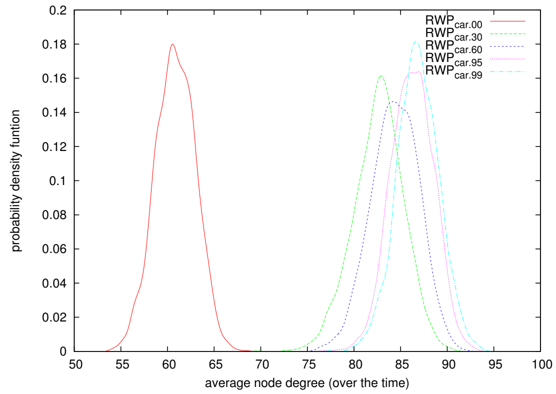


Figure 4: Average node degree distribution for 100m communication range

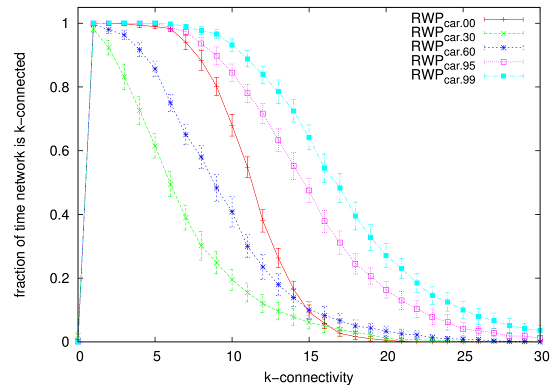


Figure 5: Fraction of time the topology is k-connected over k (average and 0.95 confidence intervals)

the robustness of the topology. Figure 5 shows an example for different Random-Waypoint variants. It can be seen that attraction points with lower attraction grades lead to worse connected networks, while higher attraction grades yield better connectivity. The reason for this is in the positions of the attractions points. The points are modeled mainly in one part of the simulation area. If the attraction grade is high enough, there is no movement in the other parts of the simulation area anymore (cf. fig. 2).

7. CONCLUSION AND FUTURE WORK

After surveying the related work and existing synthetic mobility models, we introduced the tool BonnMotion and described its architecture. BonnMotion is an open-source Java software which can be used to create and analyze mobility scenarios. Currently, the Random-Waypoint, Gauss Markov, Manhattan Grid, Reference Point Group Mobility (RPGM) and the Disaster Area model are supported. The scenarios can also be exported for the network simulators ns-2, GloMoSim/QualNet, COOJA, and MiXiM. Exports to other formats and for other simulators can be added easily. For the analysis of the generated scenarios, different metrics can be calculated as *overall* statistics and as *progressive* statistics. In the current release, velocity, relative mobility, dwell time, link duration, time to link break, node degree, partitions, and k-connectivity are supported.

For the next releases, we are working on integrating further existing models into BonnMotion. Moreover, we are using BonnMotion to develop new models. Our focus is on tactical scenarios such as urban warfare and first responder scenarios. Furthermore, the plan is to extend BonnMotion concerning the scenario analysis by integrating other metrics such as the ones described in [4].

Acknowledgement

The first versions of BonnMotion were designed and implemented by Michael Gerharz and Christian de Waal. The development of these versions were supported in part by the German Federal Ministry of Education and Research (BMBF) as part of the IPonAir project.

Since 2008, BonnMotion is partially supported by CONET, the Cooperating Objects Network of Excellence, funded by the European Commission under FP7 with contract number FP7-2007-2-224053.

Further extensions have been designed and implemented by Nils Aschenbruck, Raphael Ernst, Elmar Gerhards-Padilla, Tim Heinrich, Patrick Peschlow, Lukas Pustina, Christoph Scholz, and Matthias Schwamborn.

8. REFERENCES

[1] N. Aschenbruck, E. Gerhards-Padilla, and P. Martini, "A Survey on Mobility Models for Performance Analysis in Tactical Mobile Networks," *Journal of Telecommunications and Information Technology (JTIT)*, vol. 2, pp. 54–61, 2008.

[2] ——, "Modeling Mobility in Disaster Area Scenarios," *Special Issue on Performance Evaluation of Wireless Ad Hoc, Sensor and Ubiquitous Networks Elsevier Performance Evaluation*, vol. 66, no. 12, pp. 773–790, 2009.

[3] F. Bai and A. Helmy, "A survey of mobility models," 2004, http://nile.usc.edu/˜helmy/important/Modified-Chapter1-5-30-04.pdf.

[4] F. Bai, N. Sadagopan, and A. Helmy, "IMPORTANT: A framework to systematically analyze the Impact of Mobility on Performance of RouTing protocols for Adhoc NeTworks," *Proc. of the IEEE Infocom*, pp. 825–835, 2003.

[5] C. Bettstetter, "Mobility modeling in wireless networks: categorization, smooth movement, and border effects," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 3, pp. 55–66, 2001.

[6] ——, "Smooth is Better than Sharp: A Random Mobility Model for Simulation of Wireless Networks," *Proceedings of the 4th ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems (MSWIM), International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems*, pp. 19–27, 2001.

[7] C. Bettstetter, G. Resta, and P. Santi, "The Node Distribution of the Random Waypoint Mobility Model for Wireless Ad Hoc Networks," *IEEE Transactions on Mobile Computing*, vol. 2, no. 3, pp. 257–269, 2003.

Digital Object Identifier: 10.4108/ICST.SIMUTOOLS2010.8684
http://dx.doi.org/10.4108/ICST.SIMUTOOLS2010.8684

[8] C. Bettstetter and C. Wagner, “The Spatial Node Distribution of the Random Waypoint Mobility Model,” *Proceedings of the 1st German Workshop on Mobile Ad-Hoc Networks (WMAN’02)*, pp. 41–58, 2002.

[9] S. Bittner, W.-U. Raffel, and M. Scholz, “The Area Graph-based Mobility Model and its Impact on Data Dissemination,” *Proc. of the IEEE PerCom*, pp. 268–272, 2005.

[10] K. Blakely and B. Lowekamp, “A structured group mobility model for the simulation of mobile ad hoc networks,” *International Conference on Mobile Computing and Networking - Proc. of the 2nd int. workshop on Mobility management and wireless access protocols*, pp. 111 – 118, 2004.

[11] D. M. Blough, G. Resta, and P. Santi, “A Statistical Analysis of the Long-Run Node Spatial Distribution in Mobile Ad Hoc Networks,” *Wireless Networks*, vol. 10, pp. 543–554, Sep. 2004.

[12] T. Camp, J. Boleng, and V. Davies, “A Survey of Mobility Models for Ad Hoc Network Research,” *Wireless Communication and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, vol. 2, no. 5, pp. 483–502, Sep. 2002.

[13] C.-H. Chen, H.-T. Wu, and K.-W. Ke, “General Ripple Mobility Model of Speed-Time Pairing,” *IEEE Wireless Communications and Networking Conference*, pp. 4057–4062, 2007.

[14] D. R. Choffnes and F. E. Bustamante, “An integrated mobility and traffic model for vehicular wireless networks,” *International Conference on Mobile Computing and Networking, Proc. of the 2nd ACM int. workshop on Vehicular ad hoc networks*, pp. 69–78, 2005.

[15] W. Drytkiewicz, S. Sroka, V. Handziski, A. Koepke, and H. Karl, “A mobility framework for OMNeT++,” in *3rd International OMNeT++ Workshop*, 2003.

[16] F. Ekman, A. Keränen, J. Karvo, and J. Ott, “Working day movement model,” *Proc. of the 1st SIGMOBILE Workshop on Mobility Models for Networking Research*, pp. 33–40, 2008.

[17] *Universal Mobile Telecommunicatios System (UMTS) - Selection procedures for the choice of radio transmission technologies of the UMTS*, Umts 30.03 version 3.2.0, tr 101 112 ed., European Telecommunications Standards Institute (ETSI), 1998.

[18] K. Fall and K. Varadhan, *The ns Manual*, The VINT Project, UC Berkeley, LBL, USC/ISI, and Xerox PARC, 2009, http://www.isi.edu/nsnam/ns/.

[19] M. Günes and J. Siekermann, “CosMos - Communication Scenario and Mobility Scenario Generator for Mobile Ad-hoc Networks,” *Proc. of the 2nd Int. Workshop on MANETs and Interoperability Issues (MANETII’05)*, 2005.

[20] J. Harri, F. Filali, and C. Bonnet, “Mobility Models for Vehicular Ad Hoc Networks: A Survey and Taxonomy,” *Research Report RR-06-168*, 2006.

[21] X. Hong, M. Gerla, G. Pei, and C.-C. Chiang, “A Group Mobility Model for Ad Hoc Wireless Networks,” *Proc. of the ACM Int. Workshop on Modelling and Simulation of Wireless and Mobile Systems (MSWiM)*, pp. 53–60, 1999.

[22] S. P. Hoogendoorn and P. H. L. Bovy, “State-of-the-art of vehicular traffic flow modelling,” *Journal of Systems and Control Engineering - Special Issue on Road Traffic Modelling and Control*, no. 215(4), pp. 283–304, 2001.

[23] W.-j. Hsu, K. Merchant, H.-w. Shu, C.-h. Hsu, and A. Helmy, “Weighted waypoint mobility model and its impact on ad hoc networks,” *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 9, no. 1, pp. 59–63, Jan. 2005.

[24] Y. Huang, W. He, K. Nahrstedt, and W. C. Lee, “Corps: Event-driven mobility model for first responders in incident scene,” *Proc. of the IEEE Military Communications Conference*, pp. 1–7, 2008.

[25] M. Jahnke, A. Wenzel, and G. Klein, “FKIE-Bericht Nr. 163: Verfahren zur Erkennung von Angriffen gegen taktische MANETs,” Forschungsinstitut für Kommunikation, Informationsverarbeitung und Ergonomie (FGAN-FKIE), Tech. Rep., 2008.

[26] A. Jardosh, E. M. Belding-Royer, K. C. Almeroth, and S. Suri, “Towards Realistic Mobility Models for Mobile Ad hoc Networks,” *Proc. of IEEE MobiCom*, pp. 217–229, 2003.

[27] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark, “Scenario-based Performance Analysis of Routing Protocols for Mobile Ad-hoc Networks,” *Proc. of the Mobicom*, pp. 195–206, 1999.

[28] D. B. Johnson and D. A. Maltz, “Dynamic Source Routing in Ad Hoc Wireless Networks,” in *Mobile Computing*, Imielinski and Korth, Eds. Kluwer Academic Publishers, 1996, vol. 353, pp. 153–181.

[29] A. Köpke, M. Swigulski, K. Wessel, D. Willkomm, P. Klein Haneveld, T. Parker, O. Visser, H. Lichte, and S. Valentin, “Simulating Wireless and Mobile Networks in OMNeT++: The MiXiM Vision,” in *First International OMNeT++ Developers Workshop, Marseille, France*, Mar. 2008.

[30] J. Kraaier and U. Killat, “The Random Waypoint City Model - User Distribution in a Street-Based Mobility Model for Wireless Network Simulations,” *Proc. of the 3rd ACM Int. Workshop on Wireless Mobile Applications and Services on WLAN Hotspots*, pp. 100–103, 2005.

[31] D. Krajzewicz, G. Hertkorn, C. Rössel, and P. Wagner, “Sumo (simulation of urban mobility); an open-source traffic simulation,” *Proceedings of the 4th Middle East Symposium on Simulation and Modelling (MESM2002)*, pp. 183–187, 2002.

[32] K. Langendoen et al, “Mac simulator.” [Online]. Available: http://www.consensus.tudelft.nl/software.html

[33] ——, “Positif localization simulation framework.” [Online]. Available: http://www.consensus.tudelft.nl/software.html

[34] J.-Y. Le Boudec and M. Vojnovic, “Perfect simulation and stationarity of a class of mobility models,” *Proc. of the IEEE Infocom*, pp. 2743–2754, 2005.

[35] B. Liang and Z. J. Haas, “Predictive distance-based mobility management for multidimensional PCS networks,” *IEEE/ACM Transactions on Networking*, vol. 11, no. 5, pp. 718–732, Sep. 2003.

Digital Object Identifier: 10.4108/ICST.SIMUTOOLS2010.8684
http://dx.doi.org/10.4108/ICST.SIMUTOOLS2010.8684

[36] S.-T. Liang and W.-H. Sheng, "The mold mobility model for mobile wireless networks," *First International Conference on Wireless Internet*, pp. 79–84, 2005.

[37] H. Lichte and S. Valentin, "Implementing MAC protocols for cooperative relaying: A compiler-assisted approach," in *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems (SIMUTools)*, 2008.

[38] S. Lim, C. Yu, and C. R. Das, "Clustered Mobility Model for Scale-Free Wireless Networks," *Proc. IEEE Conf. on Local Computer Networks (LCN2006)*, pp. 231–238, 2006.

[39] G. Lu, D. Belis, and G. Manson, "Study on environment mobility models for mobile ad hoc network: hotspot mobility model and route mobility model," *International Conference on Wireless Networks, Communications and Mobile Computing*, pp. 808–813, 2005.

[40] P. S. Mogre, M. Hollick, N. d'Heureuse, H. W. Heckel, T. Krop, and R. Steinmetz, "A Graph-based Simple Mobility Model," *Proc. des 4. Workshops zu Mobilen Ad hoc Netzen (WMAN'07) im Rahmen der 15. ITG/GI Fachtagung Kommunikation in Verteilten Systemen (KiVS'07)*, pp. 421–432, 2007.

[41] S. M. Mousavi, H. R. Rabiee, M. Moshref, and A. Dabirmoghaddam, "MobiSim: A Framework for Simulation of Mobility Models in Mobile Ad-Hoc Networks," *Third IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, 2007.

[42] M. Musolesi, S. Hailes, and C. Mascolo, "An Ad Hoc Mobility Model Founded on Social Network Theory," *Proc. of the 7th ACM Int. Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pp. 20–24, 2004.

[43] M. Musolesi and C. Mascolo, "A Community Based Mobility Model for Ad Hoc Network Research," *Proc. 2nd ACM/SIGMOBILE Int. Workshop on Multi-hop Ad Hoc Networks: from theory to reality (REALMAN'06). Colocated with MobiHoc2006*, pp. 31–38, 2006.

[44] M. Musolesi and C. Mascolo, "Mobility Models for Systems Evaluation," *State of the Art on Middleware for Network Eccentric and Mobile Applications (MINEMA)*, 2008.

[45] W. Navidi and T. Camp, "Stationary distributions for the random waypoint mobility model," *IEEE Transactions on Mobile Computing*, vol. 3, no. 1, pp. 99–108, Jan-Feb 2004.

[46] S. Nelson, A. H. III, and R. Kravets, "Event-driven, role-based mobility in disaster recovery networks," *Proc. of Mobicom-Chants*, pp. 27–34, 2007.

[47] F. Österlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, "Cross-level sensor network simulation with cooja," in *Proceedings of the First IEEE International Workshop on Practical Issues in Building Sensor Network Applications (SenseApp 2006), Tampa, Florida, USA*, Nov. 2006.

[48] B. Pazand and C. McDonald, "A critique of mobility models for wireless network simulation," *6th IEEE/ACIS International Conference on Computer and Information Science*, pp. 141–146, 2007.

[49] S. Reidt and S. D. Wolthusen, "An Evaluation of Cluster Head TA Distribution Mechanisms in Tactical MANET Environments," *Proc. of the IEEE Military Communications Conference (MILCOM)*, 2007.

[50] G. Resta and P. Santi, "An Analysis of the Node Spatial Distribution of the Random Waypoint Model for Ad Hoc Networks," *Proc. ACM Workshop on Principles of Mobile Computing (POMC)*, pp. 44–50, 2002.

[51] E. M. Royer, P. M. Melliar-Smith, and L. E. Moser, "An Analysis of the Optimum Node Density for Ad hoc Mobile Networks," *Proc. of the IEEE International Conference on Communications*, vol. 3, pp. 857–861, 2001.

[52] A.-K. Souley and S. Cherkaoui, "Advanced mobility models for ad hoc network simulations," *Proceedings Systems Communications*, pp. 50–55, 2005.

[53] I. Stepanov, J. Hähner, C. Becker, J. Tian, and K. Rothermel, "A Meta-Model and Framework for User Mobility in Mobile Networks," *Proc. of the 11th Int. Conf. on Networking 2003 (ICON 2003)*, pp. 231–238, 2003.

[54] J. Tian, J. Hähner, C. Becker, I. Stepanov, and K. Rothermel, "Graph-based Mobility Model for Mobile Ad Hoc Network Simulation," *Proc. of the 35th Annual Simulation Symposium*, pp. 337–344, 2002.

[55] V. Toubiana, H. Labiod, and B. Fischer, "Event Based Mobility Model for Subway Scenarios," *Third IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, 2007.

[56] J. Yoon, M. Liu, and B. Noble, "Random Waypoint Considered Harmful," *Proc. of the IEEE Infocom*, pp. 1312–1321, 2003.

[57] X. Zeng, R. Bagrodia, and M. Gerla, "Glomosim: a library for parallel simulation of large-scale wireless networks," *Proceedings of the 12th Workshop on Parallel and Distributed Simulations*, 1998, http://pcl.cs.ucla.edu/projects/glomosim/.

[58] Q. Zheng, X. Hong, and J. Liu, "An Agenda Based Mobility Model," *Proc. of the 39th annual Symposium on Simulation*, pp. 188–195, 2006.

[59] B. Zhou, K. Xu, and M. Gerla, "Group and swarm mobility models for ad hoc network scenarios using virtual tracks," *IEEE Military Communications Conference*, pp. 289–294, 2004.

[60] H.-M. Zimmermann and I. Gruber, "A Voronoi-based Mobility Model for Urban Environments," *European Wireless 2005*, 2005.

Digital Object Identifier: 10.4108/ICST.SIMUTOOLS2010.8684
http://dx.doi.org/10.4108/ICST.SIMUTOOLS2010.8684