Paper

Parallel and Distributed Simulation of Ad Hoc Networks

Andrzej Sikora and Ewa Niewiadomska-Szynkiewicz

Abstract— Modeling and simulation are traditional methods used to evaluate wireless network design. This paper addresses issues associated with the application of parallel discrete event simulation to mobile ad hoc networks design and analysis. The basic characteristics and major issues pertaining to ad hoc networks modeling and simulation are introduced. The focus is on wireless transmission and mobility models. Particular attention is paid to the MobASim system, a Java-based software environment for parallel and distributed simulation of mobile ad hoc networks. We describe the design, performance and possible applications of presented simulation software.

Keywords— ad hoc network, distributed simulation, mobile network, software systems.

1. Introduction

Ad hoc networks are the ultimate technology in wireless communication that allow network nodes located within its transmission range to communicate directly to each other without the need for an established infrastructure such as base station, and centralized administration. For communicating with nodes located beyond the transmission range, the node needs to use intermediate nodes to relay messages hop by hop, thus, in general, routes between mobile nodes may include multiple hops.

A mobile ad hoc network (MANET) [1] is formed through the cooperation of an arbitrary set of independent nodes – mobile, wireless devices. The nodes are free to move randomly and organize themselves. The network's wireless topology may change rapidly and unpredictably. There is no prearrangement assumption about specific role each node should perform. Each node makes its decision independently, based on the situation in the deployment region and its knowledge about the network. Mobile ad hoc networks may operate in a standalone fashion, or may be connected to the Internet. The above description outlines the features of a typical MANET application scenario:

- **Wireless network**: nodes communicate wirelessly and share the same media (e.g., radio).
- **Heterogenous network**: a typical MANET is composed of heterogenous devices.
- **Infrastructureless network**: nodes operate in peer-to-peer mode, act as autonomous routers, and generate independent data; a network does not depend on any fixed infrastructure. MANETs are easy to deployment.
- **Dispersed network and multihop routing**: nodes composing the network are geographically dispersed, thus, multihop communication is necessary – each node may act as a router.
- **Time varying topology**: the topology is dynamic in nature due to the constant movement of the participating nodes.

By exploiting ad hoc wireless technology, various portable devices and fixed equipment can be connected together, forming a sort of ubiquitous network. MANETs enable devices to create and join networks on the fly – any time and anywhere for a given application. Potential applications of wireless ad hoc networks are numerous. Among them, we can cite following: delivery of location-aware information, traffic or health monitoring, intrusion detection, ubiquitous Internet access, etc.

Ad hoc architecture has many benefits, however its flexibility come at a price. A number of complexities and design constraints are concerned with the features of wireless communication (limited transmission range, limited link bandwidth and quality of transmission, constrained resources), mobility and multihop nature of the network [2]–[6].

Currently research effort is directed toward these specifics and constraints in mobile ad hoc networks. Although mathematical modeling and analysis allow to solve many problems and bring some insights into the design of MANETs, the complexity and scale of modern ad hoc networks limit the applicability of purely analytic approaches. Thus, computer simulation can significantly help to obtain crucial performance characteristics.

Computer simulation has been widely recognized as an important tool for researchers and engineers that allow to design and analyze the behavior and performance of cable and wireless networks, and verify new ideas (new protocols, mechanisms, network services, etc.) [7]–[10]. The main difficulty in large scale networks simulation is the enormous computation power, i.e., speed and memory requirements needed to execute all events involved by internodes communication and nodes' mobility. As a consequence, the developments of methods to speed up calculations has recently received a great deal of interest. Parallel and distributed discrete event simulation has already proved to be very useful when performing the analysis of different network systems [3], [8], [9], [11]. It allows to reduce the computation time of the simulation program, and to better reflect the structure of the simulated physical system. Parallel execution of computations can improve the scalability of the network simulator both in term of network size and execution speed, enabling large scale networks and more network traffic to be simulated in real time.

In this paper, we discuss some guidelines related to wireless, mobile, and ad hoc networks modeling and simulation. We model MANET application using discrete event systems methodology (DEVS) and address the challenges to design high-performance simulation of MANETs' systems. Finally, we describe organization, implementation, usage, and practical application of our ad hoc networks simulator called MobASim.

2. Mobile Ad Hoc Network Modeling

In the performance evaluation of an ad hoc network application, simulations should be done under a variety of modeling parameters and conditions, in order to capture effects of the simulated real life system. In MANETs a correct model design should evaluate a priori any possible relationship among simulated area, network topology, mobility levels, wireless transmission, power consumption, etc.

2.1. Mobility Models

Modeling of movement of network nodes plays the crucial role in almost every simulation experiments of MANETs. The dynamic topologies due to nodes' mobility introduces adaptive behavior of users, control mechanisms, and communication protocols. The mobility models should resemble the real life movements, and at the same time be simple enough for simulation. In general, two types of mobility models have been adopted in the simulation of MANETs [3], [6], [12].

- **Motion traces** that provide accurate information about mobility patters and behavior of the nodes in the considered environment (e.g., streets, highways). Traces define the positions of nodes in time, so they require long files depending on the time granularity of samples. It is good description of steady-state mobility if the motion samples are collected for considered time intervals.
- **Synthetic models** are analytical random-motion models that describe mobility without using real traces. We can distinguish several less and more realistic synthetic models. The random mobility model is a discrete implementation of a Brownian-like motion. In the random waypoint model each node chooses uniformly at random a destination point and velocity, and moves toward it along a straight line. The random direction model is similar to the previous one, but in this model each node chooses uniformly at random a direction.

The *map-based mobility models* are used for applications in which nodes are constrained to move within defined paths. Most of presented models describe an obstacle-free movement.

2.2. Wireless Transmission Modeling

A simulation of wireless communication, including propagation, mobility, and interference is very difficult and computationally expensive task. The main problem in wireless communication modeling is estimation of the size of the transmission area of a transmitter. This area can be defined as the area where the transmitted signal between any two nodes u and v propagates and can be correctly detected and decoded. We can define the signal degradation $PL(d)$ with a distance d:

$$PL(d) = \frac{P_t}{P_r}, \tag{1}$$

where d denotes the distance between nodes u and v, P_t power used by u to transmit the signal and P_r power of the signal received by v. $PL(d)$ is called "path loss" with a distance d.

A path loss modeling is difficult but very important task. If we know the model of $PL(d)$ we can predict the occurrence of a radio channel between any two nodes in the network. Over time, many less and more detailed propagation models have been introduced [3], [6], [13], [14]. In practice, three techniques for path loss estimation are extensively used: long-distance path loss models, log-normal shadowing, and fading models. The long-distance models predict variations of the signal intensity over large distances. They have been developed as a combination of analytical and empirical methods. In these models the average large-scale path loss is expressed as a function of a distance d raised to a certain exponent n ("distance-power gradient"), which indicates the rate at which the path loss increases with a distance:

$$PL(d) = PL(d_0)\left(\frac{d}{d_0}\right)^n, \tag{2}$$

$$PL(d)[\mathrm{dB}] = PL(d_0)[\mathrm{dB}] + 10n\log\left(\frac{d}{d_0}\right), \tag{3}$$

where d_0 denotes a close-in reference distance determined from measurements close to transmitter, d a distance between transmitter and receiver.

The log-normal shadowing model considers the fact that the transmission area of a transmitter may be different at two different locations, which leads to measure signals that are different than the average value calculated by Eq. (3). In this model path loss at distance d is modeled as random variable with log-normal distribution:

$$PL(d)[\mathrm{dB}] = PL(d_0)[\mathrm{dB}] + 10n\log\left(\frac{d}{d_0}\right) + X_\sigma, \tag{4}$$

where X_σ is a zero-mean Gaussian distributed random variable with standard deviation σ (all in dB).

The fading models predict variations of the signal intensity over very short distance.

3. MobASim – Software System for Ad Hoc Networks Simulation

The MobASim system provides a framework for mobile ad hoc networks simulation performed on parallel computers or computer clusters. It can help testing of various technologies designed for ad hoc networks application scenarios. The considered network to be simulated is described by different parameters defined by the user, thus we can perform the experiments for various topologies, wireless devices, mobility models, routing protocols, localization capabilities, etc. In this section we present the design and implementation of MobASim and comparison of our project to the other existing tools for ad hoc networks simulation.

3.1. Related Works and Comparison

Today, many software tools for wireless networks simulation are proposed. Some popular network simulators like OPNET [15], ns-2 [16], OMNeT++ [17] or GloMoSim [18] can simulate ad hoc networks. The others are dedicated to MANETs [19] or wireless sensor networks [10] simulation. The simulators provide the facility to simulate protocols in different layers, nodes mobility, energy consumption and various ad hoc networks application scenarios. Different tools are optimized for different purposes.

However, most of available simulators require costly shared-memory supercomputers to run even medium size network simulation. We are involved to large scale network systems simulation and their practical applications, and our goal was to develop scalable simulator operating in real time. Hence, to provide high performance and scalability we utilized the paradigm of federating disparate simulators [20] and asynchronous distributed simulation technology [21]–[23]. This is the main difference between our software and the other tools. The other reason for developing a new simulator was the complicated architecture of available systems and limitations in results visualization and user-system interaction. In case of OPNET, OMNeT++ or ns-2 systems a user must read a large number of manuals to learn how to use the tool. The source coding is specialized and it is not easy to implement a given example and add modules developed by the user.

Moreover, many systems do not support both the user interactions during the experiments and animation of network topology changes. Users set configuration parameters before starting the simulation, and they can see computation results after the experiment is terminated. In addition, most existing ad hoc networks simulators focus on the MAC protocols implementation with the lack of the radio management and mobility modeling. Usually only simplified wireless transmission models and obstacles free simple mobility models are provided (ns-2, OMNeT++).

The MobASim is a general purpose federated simulator which elements can be easily reused in many computations. The process of implementing a given application for MobASim is quite straightforward and convenient especially thanks to GUI (graphical user interface) and dedicated language ASimML – the XML (extensible markup language) schema specification. MobASim supplies the library of classes to implement the user's modules, which are specific to a given application. Hence, the current version of our simulator provides different models of radio management, mobility models handling obstacles, user-friendly interface and tools for results visualization and animation. The open design of MobASim architecture, easy usage, and its extensibility to include external modules, was chosen in the hope that the system will be a useful platform for research and education in ad hoc networks modeling and testing. The software will be free available for researchers and students.

3.2. MobASim Overview

The discrete event systems methodology is applied to model mobile ad hoc network operation, i.e., the process being modeled is understood to advance through events [21], [23]. The major concept is defined as follows:

- **System**: a collection of entities that interact together over time to accomplish a set of goals or objectives.
- **Model**: a representation of the system in terms of its entities and their events, attributes and objectives.
- **Entity**: component of the system that requires the explicit representation in the model.
- **System state**: a collection of variables which values define the sate of the system at a given point of time.
- **Event**: an instantaneous occurrence in time that alters the state of the system.

In our application the *system* denotes the wireless, mobile, and ad hoc network, *entities* are components of this network responsible for different functionalities. We distinguish three types of such components:

- **Node**: a mobile device that performs the assigned task. It can change dynamically its position in the deployment region and can interact with other nodes in the system.
- **Communication manager**: an object that models the wireless communication between all nodes.
- **Mobility manager**: an object responsible for tracking the nodes on the map and collision avoidance.

The MANET simulator developed in the MobASim system consists of logical processes (LPs) implementing the operation performed by three listed types of entities: *nodes*, *communication managers* and *mobility managers*. Hence, LPs are divided into three groups of computing processes, adequately responsible for:

- N – tasks to be performed by mobile nodes,
- CM – internode wireless communication and the network communication topology updating,
- MM – mobile nodes movement and providing the access to information about the terrain (deployment region) and other nodes location in the network.

Each process from the group MM can implement one of three mobility models. It is possible to combine various models in one simulator, i.e., the model of mobility can switch w.r.t. the current state of the node. The processes from the group CM implement one of two wireless communication models.

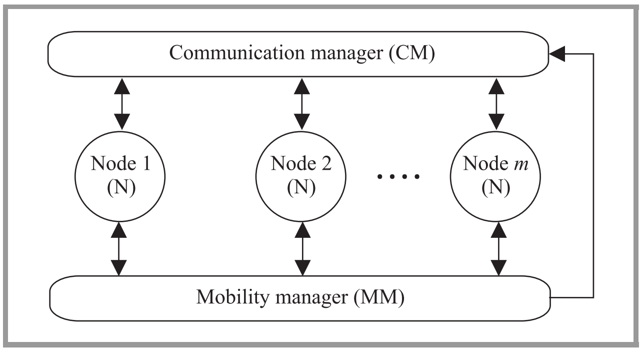


Fig. 1. The architecture of MobASim application.

The structure of a given application (MANET simulator) implemented in MobASim is presented in Fig. 1. We can see that every simulator of MANET is composed of one logical process from the group CM, one process from the group MM, and several processes from the group N. The number of N processes is equal to the number of wireless devices in the simulated network.

3.3. Mobility Models in MobASim

The popular commercial and publicly released software tools for networks simulation support mobility models based on motion traces, see OPNET [15], ns-2 [16], GloMoSim [18]. These models describe an obstacle-free movement. The user has to provide accurate information about mobility patterns. Our MobASim simulator provides three types of mobility models. In all cases the obstacles are accounted. The obstacles are generated by the user or are localized based on a real map. They are accounted for also when simulating the radio signal propagation. It is assumed that wireless signal is obstructed by the obstacles.

The state of each mobile node is described by four state variables:

- location within the deployment region,
- orientation (an angle between X axis and the direction of node movement),
- speed of movement,
- energy stored in the node.

It is assumed that generated movement paths are dynamically changed taking into account the state of the nodes and surroundings (obstacles and neighbouring nodes). All data concerned with the deployment region and all nodes in the network are stored in the data base served by the logical process MM (mobility manager). The DEVS methodology is used to implement mobility models. The following types of events are defined:

- MStart – start the movement,
- MC – continue the movement,
- MStop – stop the movement,
- LT – track the changes in the node location,
- DU – update the MobASim data base (changes in nodes' location and their surrounding),
- CA – alert the node to a collision (the movement directions of at least two nodes are crossed),
- CO – a collision between two nodes was occurred.

All presented events are served by logical processes from two groups: N (node) serves, respectively, MStart, MC, MStop, LT, DU, CA, CO events, and MM (mobile manager) serves DU, CA and CO events. Hence, the motion trajectory is generated dynamically and results the following events execution: MStart, several MC and CA, and finally MStop. The number of MC events depends on the distance to the destination point, node velocity, obstacles occurrence and sampling intervals. In the case of collision the CO event is executed.

Three types of mobility models are implemented. Types MM1 and MM2 are modified versions of the random waypoint model (RWP). Each node chooses at random a destination point and velocity, and next moves toward the direction. In addition, in our model the user can define the specific destination instead of random generation. The main difference to RWP is that in case of MM1 and MM2 the obstacles are considered. Hence, the shortest (if possible) path is calculated from the current position of the node to the destination to avoid the collision, while in case of the RWP model the node moves toward the destination along a straight line. The differences between models MM1 and MM2 are such that in case of MM1 we assume the access to the information about the whole deployment region, i.e., we know the locations of all obstacles and current positions of all neighbor nodes in the network. In case of MM2 we assume the restricted access to the data about the environment. Only the knowledge about the obstacles and other nodes located in the surrounding of a given node is available. The path has to be dynamically changed after possibility of collision identifying. It is not guaranteed that the shortest path will be realized in case of this model.

Both in models MM1 and MM2 nodes are free to move within the deployment region. In case of model MM3 (map-based mobility) nodes are constrained to move within specified paths. All these paths are stored in the MobASim data base.

Each MobASim application can implement all described models: MM1, MM2 and MM3. The mobility model can dynamically change w.r.t. the current state of the node.

3.4. Wireless Transmission Modeling in MobASim

Most of the available software platforms for mobile ad hoc network simulation implement only large-distance wireless transmission model Eq. (3) in its simplest version. MobASim simulator implements two of the transmission models described in the Subsection 2.2: long-distance Eq. (3) and shadowing Eq. (4).

The medium access control (MAC) layer is of fundamental importance in wireless ad hoc networks. MAC protocols are responsible for controlling the access to wireless channel. MobASim provides the implementation of MAC protocols from three categories based on the method that they handle the hidden and the exposed terminal problems: class 1 – protocol assuming random access to the wireless channel (the hidden and exposed node problem is unsolved), class 2 – the protocol solves the hidden node problem but leaves the exposed node problem unsolved, class 3 – the protocol solves both the hidden node and the exposed node problems, but requires the deployment of an additional signaling channel. The MobASim user can choose the protocol suitable to designed application. The currently available version of MobASim implements the simplified models of the physical layer and the interference management. We assume that the accurate model of MAC layer can be adopted from the other open source simulators, if necessary.

4. MobASim System Design and Implementation

The MobASim system is completely based on Java. At the heart of its technology is the asynchronous simulation Java (ASimJava) library – collection of Java-based procedures that can be used to develop general purpose discrete-event parallel and distributed simulators designed as federations of disparate simulators, utilizing runtime infrastructure (RTI) to interconnect them. Each simulator is described in terms of logical processes that communicate with each other through message-passing. LPs simulate the real life physical processes. The federation paradigm described in [20] allows to perform parallel or distributed calculations, i.e., each simulator can be executed in a separate processor or machine. The goal is to speed up calculations and perform real time simulation. The synchronous and asynchronous variants of simulation are provided [21], [23]. ASimJava technology was described in details in [11], its application to computer networks simulation in [8].

Composition and implementation of MobASim. The MobASim software provides tools to build simulators utilizing ASimJava library and runtime infrastructure, thus we can develop our application as a federation of simulators implementing the subnetworks that compose the considered MANET or a federation of simulators of independent, geographically dispersed MANETs or WSNs (wireless sensor networks) that cooperate from time to time (see Fig. 2). When consider the simulation of mobile networks we have to generate a map of deployment area. The MobASim user can define the simple objects in the domain as polygons. For more detailed description of a terrain to be considered the MobASim simulator provides the interface to the GeoTools toolkit. The GeoTools [24] is an open source Java coded library containing standard methods for the manipulation of geospatial data. All geographical information are stored in the MobASim database.

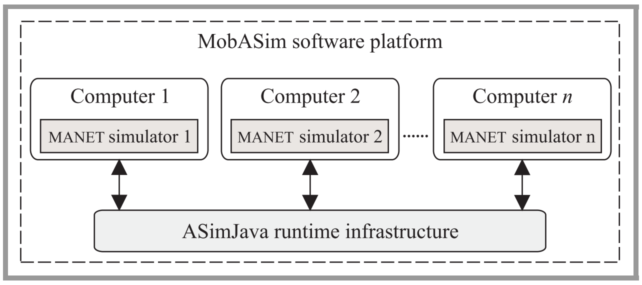
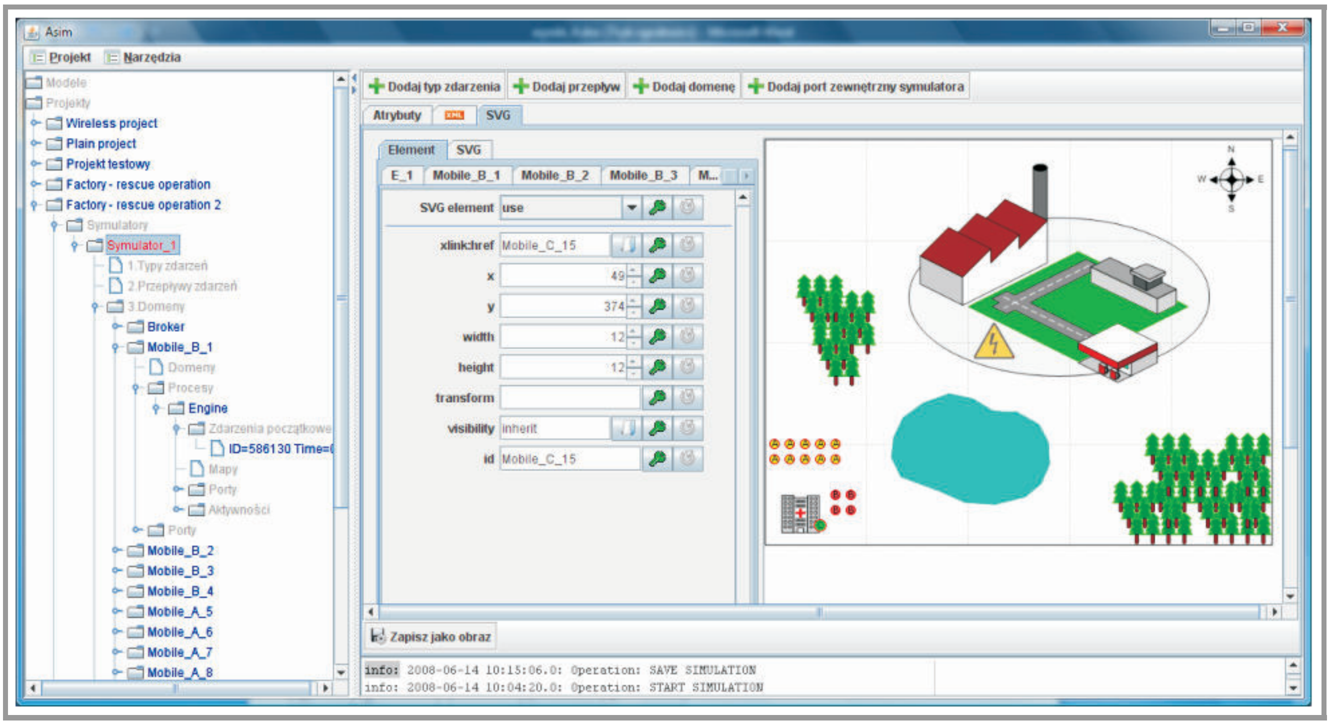


Fig. 2. A federation of simulators in MobASim.

In summary, the MobASim simulator is composed of: a runtime platform of ASimJava, a set of libraries of functions for parallel discrete event systems calculation provided in ASimJava, a set of libraries of functions for mobile and wireless applications, and a set of tools mainly to support the interaction with the user and visualisation tool for the runtime monitoring (see Figs. 3 and 4).

The user GUI is organized in a set of nested windows. The setting windows are used to facilitate the configuration phase. The network is constructed graphically. The user can enter parameters concerned with the whole network (number of nodes, wireless transmission model), network nodes (radio communication range, minimal and maximal speed, mobility model, routing protocol, MAC protocol, energy reserve, etc.), and deployment area (type of geographical data). The dedicated setting windows are used to insert the parameters specific to chosen mobility and wireless communication models provided in the system. Finally, the user is asked to configure the experiment (simulation time, number of processes, number of machines, etc.). After completing the initial settings, MobASim starts the simulation experiment. The results of simulation – time varying topology (animation of nodes) and adequate statistics are displayed. The configuration of the system to be simulated can be loaded and saved into the disc file in the XML format.



SVG element: use

xlink:href: Mobile_C_15

x: 49

y: 374

width: 12

height: 12

transform:

visibility: inherit

id: Mobile_C_15

```
info: 2008-06-14 10:15:06.0: Operation: SAVE SIMULATION
info: 2008-06-14 10:04:20.0: Operation: START SIMULATION
```

Fig. 3. MobASim graphical user interface.

5. Simulation Experiments

In order to evaluate the efficiency of MobASim, and indicate the usefulness of such software systems to support the decision making process in case of real-life problems the simulations of several ad hoc network topologies were performed. In this paper the application of our tool to support the design of MANET for the rescue action is discussed. Let us consider the following situation. The explosion in the factory devastated its surrounding. Most of the communication infrastructure, i.e., base stations for cable networks, wired phone lines, etc., was destroyed. Thus, two of priorities in the disaster management are to organize on-line monitoring of the situation on the disaster scene and to organize a relief effort for explosion victims by dispatching several rescue teams to the disaster area. The efforts of the rescue teams should be coordinated. It can be achieved only if rescuers are able to communicate, both within their team and the members of the other teams. To carry out these goals it is necessary to reinstall the communication infrastructure as quickly as possible. It can be done by deploying temporary communication equipment (vehicles equipped with transceivers), and creating an ad hoc network. By using multihop wireless communication and mobile nodes acting as communication relay stations, even relatively distant rescuer will be able to communicate. The communication will be possible without the need for rebuilding the fixed communication infrastructure.

The ad hoc network designed for reestablishing the communication for the discussed example consists of three types of nodes (see Fig. 4):

- A: the mobile node – wireless router (e.g., vehicle equipped with a transceiver) that provides the communication between nodes B and C;
- B: the mobile node – the rescue unit (e.g., the rescuer equipped with a transceiver) working in the disaster area;
- C: the base station – the rescue center that coordinates the rescue action, controls the nodes A and B and collects the data (monitoring of the situation) transmitted by nodes B.

The ad hoc network of ten wireless routers (nodes A), four rescue units (nodes B) and one rescue center (node C) was used for the rescue action. For the purpose of simulations we assumed following values of parameters used in wireless communication model: distance power gradient $n = 2$ in Eq. (2) and standard deviation $\sigma = 6$ dB in Eq. (4). The map of the deployment area was generated based on MobASim GUI and saved to the MobASim database.

Several simulation experiments were performed. The objective was to design the mobile ad hoc network that provide the continuous communication with all rescuers during the rescue action.

During the simulations the bandwidth of all links and current traffic are calculated, and the critical paths are pointed. The animation of time varying network topology – all nodes moving from the initial position to the destination, avoiding the obstacles – are displayed in MobASim main window. The user can keep track how the communication network created by a set of nodes A adopts to the new positions

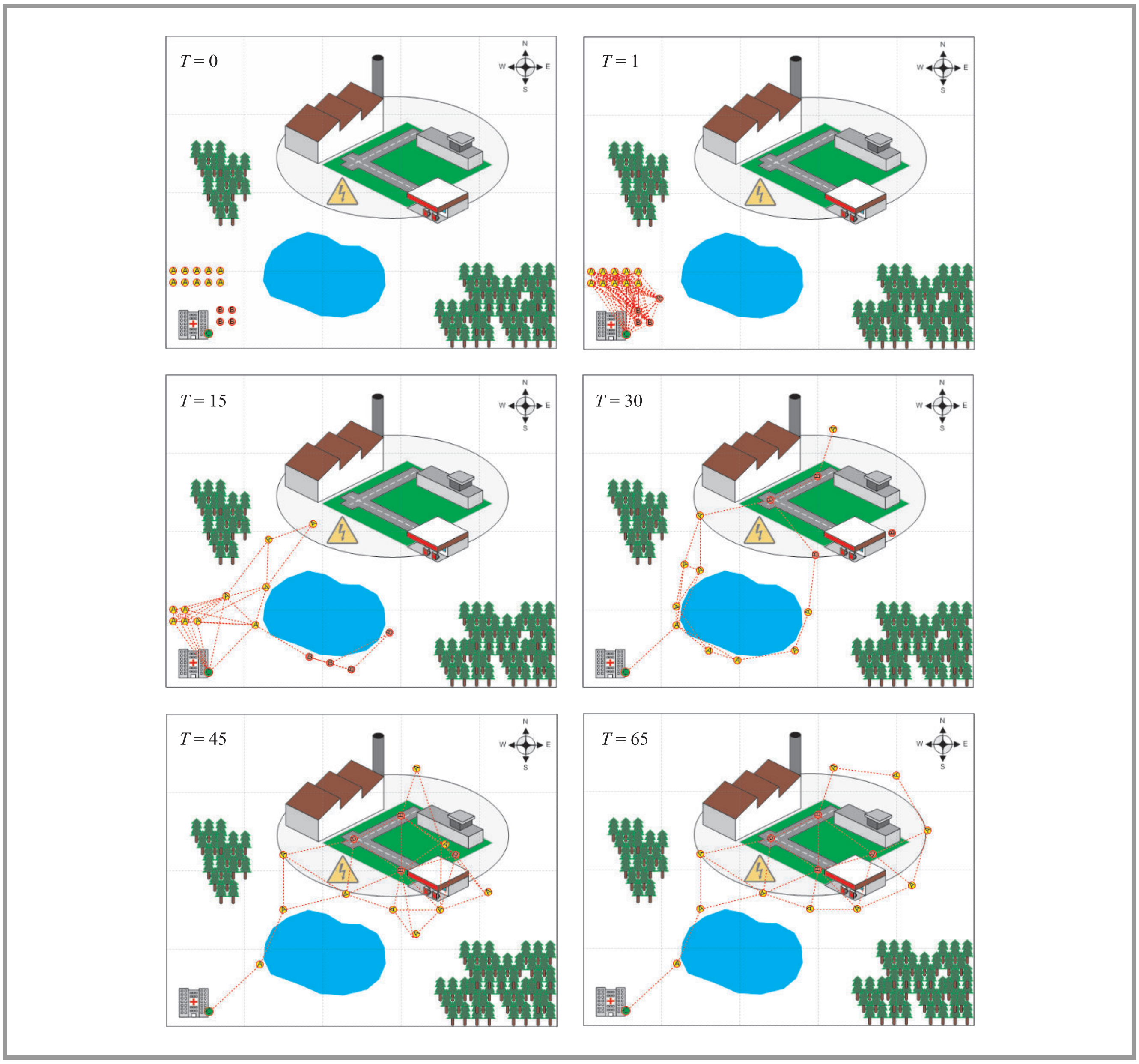


Fig. 4. MobASim simulator: application of ad hoc network.

of rescue teams (nodes B). The current network connectivity is marked by lines connecting the nodes. They appear when the communication between two nodes is possible (the distance is less than the radio range and the node is not a hidden one).

In our experiments we considered various range assignments of radio transmitters in nodes A and B. We tested the efficiency of the mobility manager algorithm implemented in the rescue center (node C), which goal is to calculate the paths that substitute the critical ones. It allows to create the robust and optimal network topology for the current time period.

The results are presented in a figure and two tables. Figure 4 shows the dynamically changing network topology during the entire network operational lifetime. The snapshots of initial, temporary and final topologies calculated for 0, 1, 15, 30, 45 and 65 steps of simulated time units are presented.

The initial and final positions of all nodes in the network are collected in Table 1.

Next, we compared the results obtained for MAC protocols from three categories: class 1, 2 and 3. MAC protocols restrict the number of simultaneous signal transmissions per unit of area and consequently restrict the number of interfering nodes. The number of wireless connections and interfering nodes in case of various classes of MAC protocol for several time steps (every 5 units of simulated time) are presented in Table 2. It can be seen from the table that, as expected, the number of interfering nodes is highest in MAC class 1 and lowest in MAC class 2. For the calculation of the interference power in ad hoc and sensor networks, the density and the distribution of the interfering nodes must be known. If the density of nodes increases the number of nodes falling within the prohibited transmission areas increases. The density of interfering nodes is not expected to increase linearly with the increase in the density of nodes.

Table 1
Simulation results

Node name	Time [s]	Initial position (x, y) [m]	Destination position (x, y) [m]
B1	0	(85,350)	(240,160)
B2	2	(85,365)	(300,130)
B3	4	(70,350)	(300,200)
B4	6	(70,365)	(370,180)
A1	7	(70,300)	(320,70)
A2	9	(70,315)	(150,180)
A3	11	(55,300)	(230,230)
A4	13	(55,315)	(120,320)
A5	15	(40,300)	(150,250)
A6	17	(40,315)	(290,250)
A7	19	(25,300)	(350,250)
A8	21	(25,315)	(420,220)
A9	23	(10,300)	(440,150)
A10	25	(10,315)	(400,80)
C1	–	(55,380)	(55,380)

Table 2
Results for various categories of MAC protocol

Time step	Wireless connections	Interfering nodes (MAC classes)		
		class 1	class 2	class 3
1	105	1	0	1
5	99	2	0	1
10	75	3	0	1
15	45	3	0	3
20	30	6	1	3
25	28	6	1	3
30	24	6	1	4
35	25	5	1	4
40	28	5	0	4
45	30	5	0	3
50	32	6	0	2
55	29	6	0	3
60	26	6	0	3
65	25	6	0	3

From the simulation results we see that by using multihop wireless communication and mobile nodes, the communication between the rescue center and rescue teams is possible without the need for reestablishing the fixed communication infrastructure.

6. Summary and Conclusions

The evolution of wireless, mobile ad hoc networks and improved designs will strongly depend on the ability to predict their performance using analytical and simulation methods. In this paper we described the software platform MobASim for mobile ad hoc networks simulation. MobASim was designed to be powerful, effective, scalable, flexible, and easy to use ad hoc network simulator. It can support researches and engineers during the design and implementation of MANETs applications and verification of new MANET's technologies. The tool is especially useful in large scale applications in which the speed of simulation is of essence, such as real time ad hoc networks simulation. MobASim is a general purpose federated simulator, which elements can be easily reused in many computations. The federated approach to parallel and distributed simulation of networks, provided functionality, easy usage and its extensibility to include other open source modules or modules developed by the user, which are specific to a given application, make different our tool from the popular software systems for simulation.

Acknowledgment

This work was supported by TINFO Project 2008.

References

[1] MANET Group, http://www.ietf.org/html.charters/manet-charter.html

[2] G. Aggelou, *Mobile Ad Hoc Networs. From Wireless LANs to 4G Networks*. New York: McGraw-Hill, 2005.

[3] S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic, *Mobile Ad Hoc Networking*. New York: Wiley, 2004.

[4] S. Basagni and A. Capone, "Recent research directions in wireless ad hoc networking", *Ad Hoc Netw.*, vol. 5, iss. 8, pp. 1205–1348, 2007.

[5] A. Hac, *Wireless Sensor Network Design*. New York: Wiley, 2003.

[6] P. Santi, *Topology Control in Wireless Ad Hoc and Sensor Networks*. New York: Wiley, 2005.

[7] M. Małowidzki, "Network simulators: a developer's perspective", in *Proc. Int. Symp. Perform. Eval. Comput. Telecommun. Syst. SPECTS'04*, San Jose, USA, 2004.

[8] A. Sikora and E. Niewiadomska-Szynkiewicz, "FR/ASimJava simulator: a federated approach to parallel and distributed network simulation in practice", *J. Telecommun. Inform. Technol.*, no. 4, pp. 53–59, 2006.

[9] B. K. Szymanski, A. Saifee, A. Sastry, Y. Liu, and K. Mandnani, "Genesis: a system for large-scale parallel network simulation", in *Proc. Paral. Distrib. Simul. PADS 2002*, Washington, USA, 2002.

[10] B. K. Szymanski and G. G. Chen, "Sensor network component based simulator", in *Handbook of Dynamic System Modeling*, P. Fishwick, Ed. Boca Raton: Taylor and Francis Publ., 2007, pp. 35-1–35-16.

[11] E. Niewiadomska-Szynkiewicz and A. Sikora, "ASim/Java: A Java-based library for distributed simulation", *J. Telecommun. Inform. Technol.*, no 3, pp. 12–17, 2004.

[12] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research", *Wirel. Commun. Mob. Comp. (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, vol. 2, no. 5, pp. 483–502, 2002.

[13] B. A. Forouzan, *Data Communications and Networking*. New York: McGraw-Hill, 2004.

[14] *Mobile Radio Communications*, R. Steele, Ed. New York: IEEE Press, 1994.

[15] OPNET Modeler, http://www.opnet.com/products/modeler/home.html

[16] Network Simulator ns-2, http://www.isi.edu/nsnam/ns/

[17] OMNeT++, http://www.omnetpp.org/

[18] GloMoSim, http://pcl.cs.ucla.edu/projects/glomosim/

[19] T. Facchinetti, G. Buttazzo, and L. Almeida, "A flexible visual simulator for wireless ad-hoc networks of mobile nodes", in *Proc. IEEE Conf. Emerg. Technol. Fact. Autom.*, Catania, Italy, 2005, vol. 1, pp. 50–54.

[20] S. L. Ferenci, K. S. Perumalla, and R. M. Fujimoto, "An approach for federating parallel simulators", in *Proc. 14th Worksh. Paral. Distrib. Simul. PADS 2000*, Bologna, Italy, 2000.

[21] S. Ghosh and T. S. Lee, *Modeling and Asynchronous Distributed Simulation*. New York: IEEE Press, 2000.

[22] D. M. Nicol and R. Fujimoto, "Parallel simulation today", *Ann. Oper. Res.*, vol. 53, pp. 249–285, 1994.

[23] B. P. Zeigler, H. Praehofer, and T. G. Kim, *Theory of Modeling and Simulation*. London: Academic Press, 2000.

[24] GeoTools The Open Source Java GIS Toolkit, http://geotools.codehaus.org/

Andrzej Sikora received his M.Sc. in computer science from the Warsaw University of Technology, Poland, in 2003. Currently he is a Ph.D. student in the Institute of Control and Computation Engineering at the Warsaw University of Technology. Since 2005 he works at the Research and Academic Computer Network (NASK). His research area focuses on parallel and distributed simulation, computer networks, ad hoc networks and database systems.

e-mail: A.Sikora@nask.pl
Research Academic Computer Network (NASK)
Wąwozowa st 18
02-796 Warsaw, Poland

e-mail: A.Sikora@elka.pw.edu.pl
Institute of Control and Computation Engineering
Warsaw University of Technology
Nowowiejska st 15/19
00-665 Warsaw, Poland

Ewa Niewiadomska-Szynkiewicz – for biography, see this issue, p. 67.