Development simulator of wireless sensor network

Lateef Abdzaid Qudr

The latest wireless technology and progress in chip manufacturing in the past few years allowed to move to the practical development and implementation of a new class of distributed communication systems - wireless sensor networks (WSN).

The aim of this work — creation of wireless sensor network simulator, which makes it possible to simulate its operation in different modes, including using various routing protocols. This simulator considered like the basis tool for analysis and optimization of wireless sensor networks . One of the objectives of the developed simulator is to analyze the impact of different routing protocols on the separate/overload nodes, involved in retransmission packets, sourced from other network nodes. Need to develop a simulator due to the fact that the choice of many technical solutions arising in the design of information systems with the WSN. can not be only based on laboratory tests conducted on a small network. Real information systems, such as environmental monitoring system, consist of a large number of nodes, which is impossible to reproduce in the laboratory. Therefore, the task of developing simulator WSN is relevant.

Currently, there are several approaches and appropriate tools for simulation of wireless sensor networks. So, for the development of simulator, you can use the package modeling networks ns-2/ns-3 [1]. This package is a general-purpose simulation system, strength of which is the possibility of a realistic simulation of the physical propagation of signals; however, the use of this system for simulation WSN requires writing extensions. Other than that ns--2/ns-3 allows only simulating network, but can not stimulate the work of software nodes. This deficiency is deprived simulator TOSSIM [2]. It allows simulating the work as separate nodes, and large networks, consisting of hundreds or even thousands of nodes; and gives the developer the ability to analyze and test code, which is designed for the real hardware. For the compiler, included in TOSSIM, replaces the low-level application components that interact with hardware resources node, components, interacts with the implementation of these devices in the simulator. With this simulator executes the same code as the real network nodes. However, the simulator has a relatively low productivity and is not suitable for multiple simulations of large sensor networks, which is essential in the development of tools to optimize WSN. Therefore, along with the use of this simulator [3], we decided to develop a simpler, but at the same time, more rapid and scalable simulator.

Choice of simulation technology

Because in modern wireless sensor networks, and besides, there is no centralized control node network, for modeling of the network is sufficient to describe and implement a model of a typical node. Therefore, for the implementation of simulation used the principle of multi-agent simulation [4 — 6].

Agent-based model (ABM) — simulation method that studies the functioning of decentralized agents, and how the work of the individual agents determines the operation of the system as a whole. This approach is successfully used recently not only in the natural Sciences, but also in the Humanities [7, 8]. In our case, these decentralized agents are nodes WSN. In contrast to the system dynamics, in this approach, the rules for the agents is defined at the individual level, and the rules functioning of system as a whole appear as a result of joint activity of multiple agents. In other words, modeling is used "bottom-up" approach.

In this work multiagent approach used to simulate wireless sensor networks.

Agent simulation is used for modeling complex systems, consisting of a large number of interacting subsystems. It is very well suited for the case simulation WSN with a lot of nodes. Agent-based simulation is well applicable in the case when too difficult or impossible to formalize the behavior of the system at the global level. So complex distributed systems, which include the WSN is impossible to formalize global. So complex distributed systems, which include the

WSN is impossible to formalize in higher level. The use of simulation approach for modeling distributed systems is a standard practice. In this approach, WSN is considered as a collection of interacting parts – agents, each of which acts independently on pre-defined algorithms and can interact with other agents. Thus, the behavior of the entire system consists of the interaction of its parts. It is now widely applied in various fields: from simulation of processes the stock Bourses, to prediction of disease.

Agent-based simulation involves elements of game theory, complex systems, multi-agent systems and evolutionary programming. It applies Monte Carlo method, use the theory of probability and mathematical statistics.

For the application of the principle of multi-agent simulation has been developed object-oriented model of the wireless sensor network.

Object-oriented model the WSN

In developing the simulator has been used object-oriented approach. Isolated model components which are designed for WSN classes.

In the simulator has been implemented model WSN node and the station base , and also develop environment of their interaction. Each device is represented by a separate class with its own set of properties, determine the state of a set of methods and define their behavior. Interaction environment represents a class that contains the parameters that can not be described in each device. These parameters define the network at a higher level. These parameters are for example the number of nodes and data transmission protocols. Classes of model components WSN are presented in Figure 1.

Each node WSN interacts with the model test and with model of the channel. In the model of the phenomenon described by the change of environmental parameters relevant to this phenomenon. For example, the laws of changes in temperature or the level of chemical contamination.

In the design of the simulator one of the main requirements was to ensure its scalability. For example, the extension of the model data channel to account for noise, or features of the signal propagation in the physical layer can be made by replacing the corresponding class without affecting other components of the model. Developed simulator was initially designed to expand, is achieved by using an object-oriented approach.

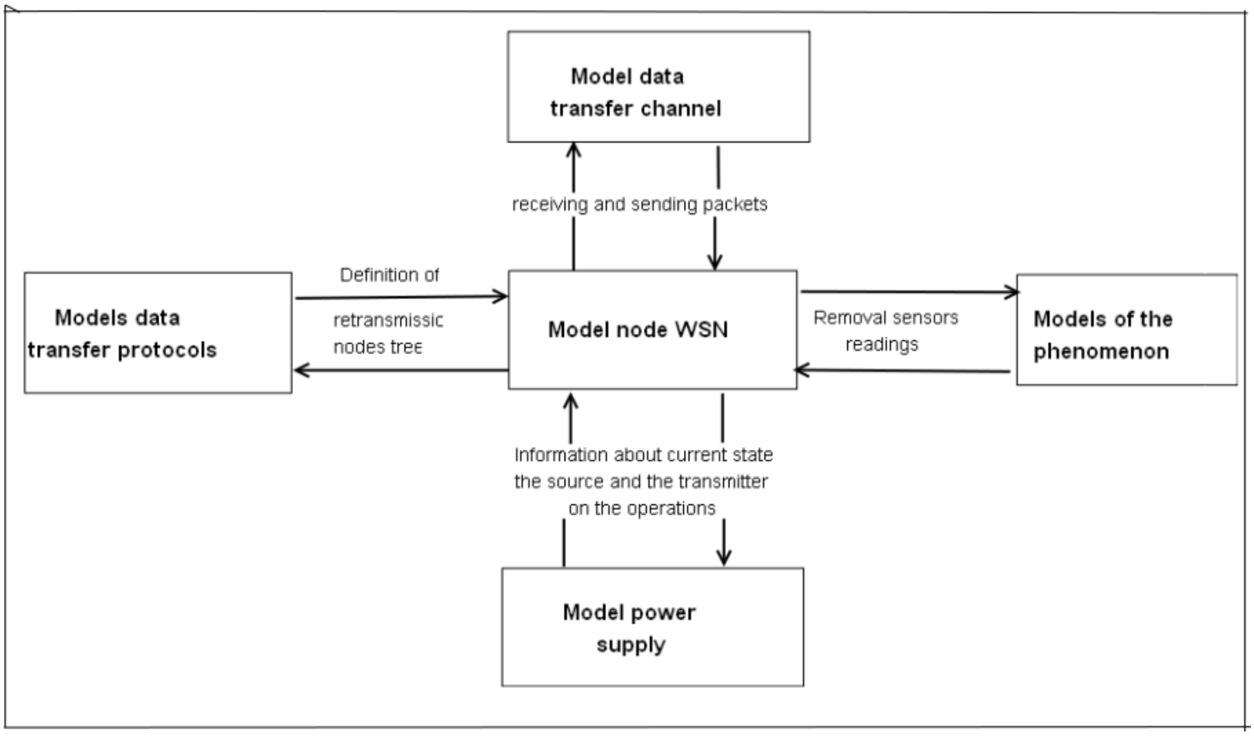

Figure 1 - diagram of classes of the object model of the WSN

This model allows describing the processes occurring in WSN. Consider, for example, the formation of the tree protocol retransmissions CTP, which is used in many implementations of WSN operating system TinyOS [9].

Each node in the CTP contains neighbor table, which stores the node ID and the cost of transmission, as a metric for the cost of the transfer value is ETX (Expected Transmissions).

Wireless sensor networks under the control of the protocol form a tree structure, which gateways are the roots, and sensory nodes - leaves. To generate routing nodes use gradient routes - ETX. The root has a value ETX = 0. When selecting a parent node preference to where the value of ETX is smaller.

Principles of choice retransmitted node in CTP can be described the following points:

each node independently determines the minimum total cost of the transmission;

Each node periodically sends them to the calculated minimum total cost of its neighboring nodes;

Receiving the lowest total cost of neighboring nodes, the node updates the information on its minimum value, if it is found the best path and sends the new value to neighbors.

During the initialization of each node sets the "uncertainty" for their total cost of transmission to the ETX base node. . The base node sets its ETX to 0. . When neighbors get the message from the base node, they update their transfer costs. Nodes update and disseminate this information on the network. Step by step, each node determines the minimum distance to the base node. Figure 2 shows the sequence diagram of the initialization process. Figure 3 shows the graph of the network and the final form of the tree routing.

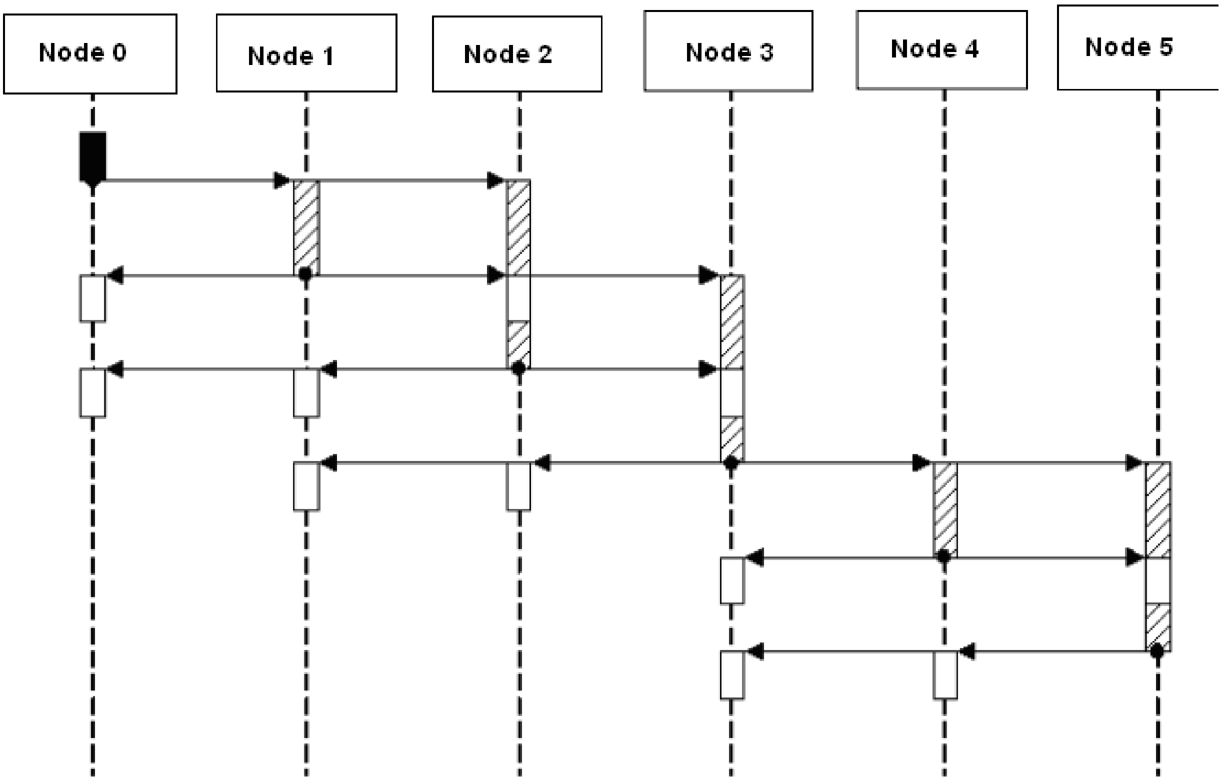

Figure 2 -diagram of sequence the process of formation of the tree retransmission

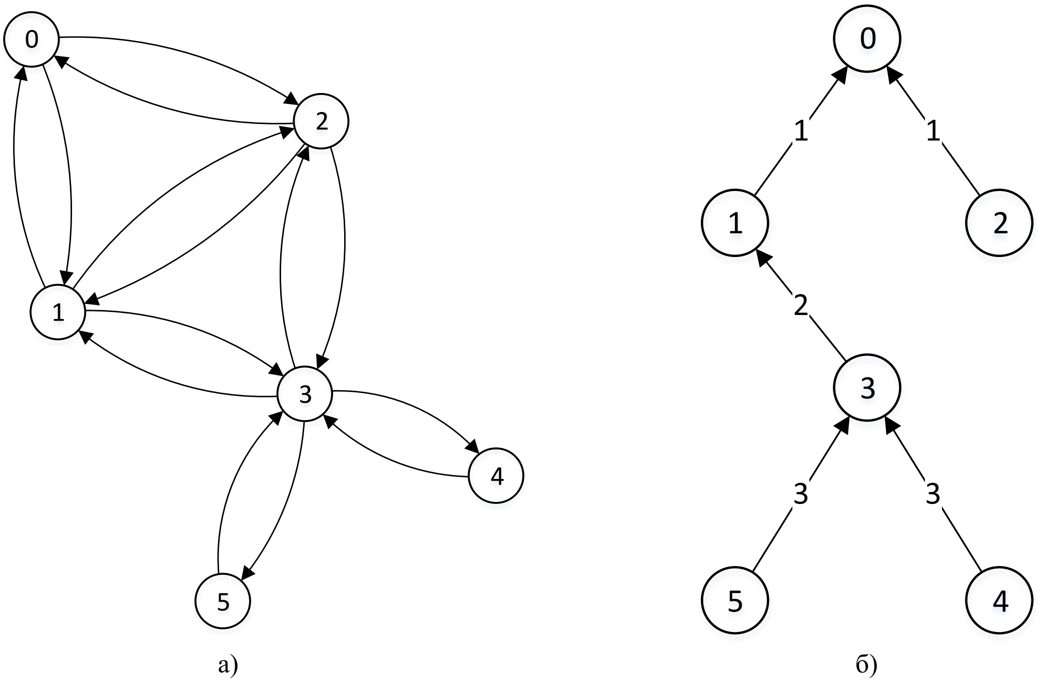

Figure 3 - Formation of the tree routing
a) The graph of the network, b) formed by tree routing.

Program implementation of the simulator

Software implementation of the simulator consists of a core and visualizer. In implementing the core use popular open source Java library for multi-agent simulation MASON. Library developed by our colleagues at George Mason University (USA) http://cs.gmu.edu/~eclab/projects/mason/.

MASON is a multi-agent simulation tool designed for simulation of systems described by a large number of agents. This system has object-oriented functions; it belongs to a class of simulation-independent domain, like Repast, Ascape, StarLogo, NetLogo, and SWARM [6].

A distinctive feature MASON is that it is intended for applications that include a large set of relatively simple agents, performing their tasks. It is often used in areas such as the modeling of traffic flows, population genetics, modeling of social processes, artificial intelligence, especially swarming intelligence.

Most of these highly specialized tools were developed to solve relatively small tasks. MASON unlike them is for problems related to the large number of agents and require multiple model runs, with the possibility use of a client-server architecture, where the server can be a distributed system or a supercomputer. It implements a variety functions that make this possible:

- model is completely separated from the visualizer, it means there can be run model absolutely without the visualizer, or using different types of visualizers with the possibility switch between them;
- model serializable, so any time you can save the current state of the model and, in the future, resume simulation from the breakpoints, while the model can be transferred if necessary, to another machine;model are independent from the platform - thanks to Java model can run on machines with different architectures and with different operating systems; model is fully encapsulated - it means that it is possible to run multiple models on the same machine, but they will run independently of each other.

In developing the visualizer used by the JUNG framework (Java Universal Network/Graph Framewark), which is a programming library for simulation, analysis and visualization of data presented in the form of graphs. It is written in the Java programming language that allows JUNG- applications use the built-in Java API, and also other Java connected libraries. JUNG architecture is designed to support different representations of entities and their relationships, such as oriented and non oriented graphs, multigraphs, graphs with parallel edges, and hypergraphs. WSN structure in our case is an oriented graph.

Currently JUNG distribution includes an implementation of several of algorithms from graph theory, data mining and social network analysis, namely: clustering algorithms, decomposition, optimization, and generation of random graphs, statistical analysis and calculation of distances between nodes in a graph, find the maximum flow, etc.

JUNG includes graphical visualization system that makes it easy to create interactive applications for working with network data. We used this framework for visualization FSU, based on information about the level of the node which receives signal from neighboring nodes. It used algorithms location of vertices of the graph, realized in JUNG.

As a result JUNG has given us a common basis for writing a visualizer wireless sensor network.

The simulator was developed on the Java programming language in the environment Eclipse 4.2 with using the above libraries.

Experimental studies of the developed simulator

For experimental studies were selected two networks. The first network is small - it consists of 6 nodes. This network is selected for debugging purposes. The second network consists of more than 200 nodes. The network was simulated to verify the developed tools on large-scale systems. In figures 4 and 5 shows the visualization of the simulated wireless sensor networks. In this case, used Kamada-Kawai graph layout algorithm [10].

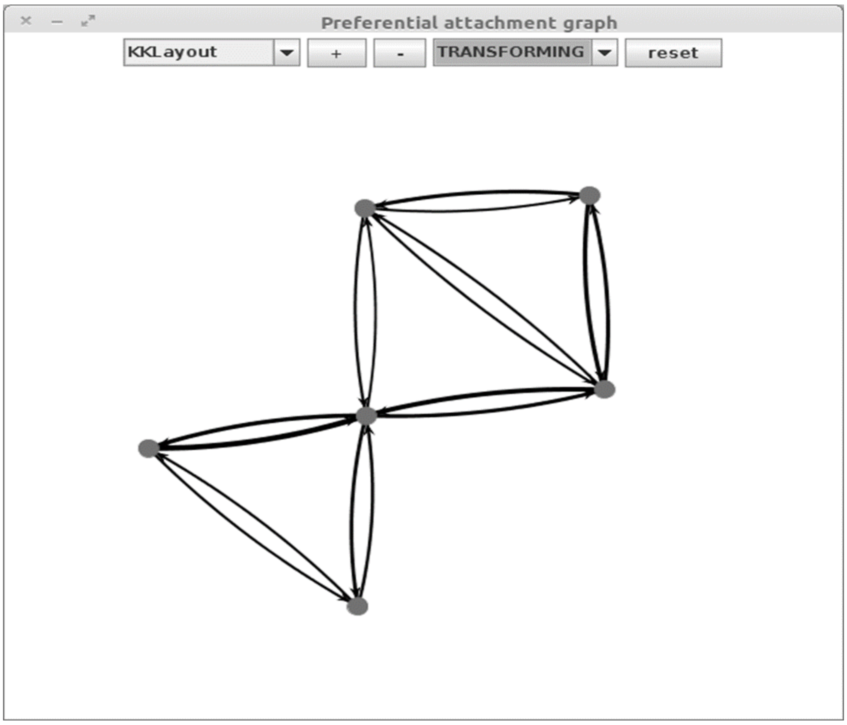

Figure 4 - Simulation of wireless sensor network from six nodes

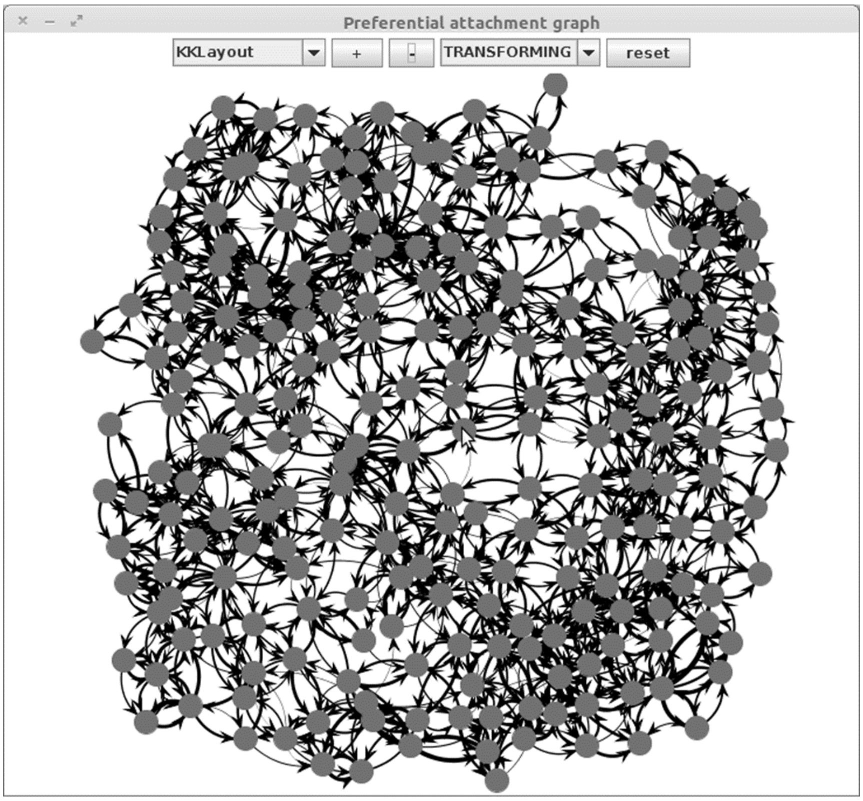

Figure 5 - Simulation of wireless sensor network from 225 nodes

Is difficult to visualizing of networks has a large number of nodes, it possible to simulate a large network with the help of developed software.

Developed simulator a sufficiently allowed us to simulate large wireless sensor network.

Conclusion

In this work was created and studied simulator wireless sensor networks based on the principle of multi-agent simulation. During the implementation following results were obtained:

- analyzed the current state of the modeling of wireless sensor networks;
- develop object-oriented model of the WSN;
- based on object-oriented model of the WSN program implementation of the simulator by using the principle of multi-agent simulation;

experimental studies developed simulator.

Analysis of experimental results showed the following.

- Developed simulator can be used to model the WSN at the design stage.
- The simulator is enough to simulate large network, Consisting of hundreds of nodes.
- Visualizing of networks is only applicable for relatively small networks, but it allows to study the details the functioning of the selected protocols data transmission.

References

1. Lacage, M. and Henderson, T.R., "Yet another network simulator", proceeding from the 2006 workshop on ns-2: the IP network simulator, ACM, 2006. Lang. Eng.
2. Simulating TinyOS Networks [Electronic resourse] - Lang. Eng. http://www.cs.berkeley.edu/~pal/research/tossim.html[checked 10/01/2012]- Lang. Eng.
3. Telyatnikov A.O., Qudr Lateef simulation of wireless sensor networks/ Problems of Information Technologies/- 2009. - No.2 (006). 106-113c- Lang. Rus.
4. Salamon, Tomas. Design of Agent-Based Models. Publisher TOMAS BRUCKNER. 2011. P. 220- Lang. Eng.
5. Shoham, Yoav, and Kevin Leyton-Brown, Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations. Cambridge University Press, 2009 P. 532- Lang. Eng.
6. From the system dynamics and traditional IT - to the practical agent-based models: Causes, technology, tools [Electronic resourse] /: http://www.gpss.ru-/paper/borshevarc.pdf -Lang. Eng.
7. Claus N. G., Svechkarev V.P**.** Modeling Dynamic Violence: Adaptive Agent-based Models, Inženernyj vestnik Dona//-2012.-No.2 [Electronic resourse] //: http://www.ivdon.ru/magazine/archive/n2y2012/844 -Lang. Rus.

8. Timchuk D.A., Svechkarev V.P**.** Multiagent simulation of the critical social behaviors, Inženernyj vestnik Dona//-2010.-No.1 [Electronic resourse] //: http://www.ivdon.ru/magazine/archive/n1e2010/175-Lang. Rus.
9. K. Akkaya, M. Younis. A Survey on Routing Protocols for Wireless Sensor Networks, Ad Hoc Networks, 3:325-349, 2005-Lang. Eng.
10. Kamada, T., Kawai, S., An algorithm for drawing general undirected graphs, Information Processing Letters,Vol. 31. – pp. 7-15. – 1989. Lang. Eng.