ITHEA

International Journal

INFORMATION TECHNOLOGIES & KNOWLEDGE

2018 Volume 12 Number 2

International Journal
INFORMATION TECHNOLOGIES & KNOWLEDGE

Volume 12 / 2018, Number 2

International Journal "INFORMATION TECHNOLOGIES & KNOWLEDGE" (IJ ITK)
is official publisher of the scientific papers of the members of the ITHEA International Scientific Society

IJ ITK rules for preparing the manuscripts are compulsory.
The **rules for the papers** for IJ ITK are given on www.ithea.org

Responsibility for papers published in IJ ITK belongs to authors.

International Journal "INFORMATION TECHNOLOGIES & KNOWLEDGE" Volume 12, Number 2, 2018
Edited by the **Institute of Information Theories and Applications FOI ITHEA**, Bulgaria, in collaboration with:
University of Telecommunications and Posts, Bulgaria, V.M.Glushkov Institute of Cybernetics of NAS, Ukraine,
Universidad Politécnica de Madrid, Spain, Hasselt University, Belgium,
University of Perugia, Italy, Institute for Informatics and Automation Problems, NAS of the Republic of Armenia
St. Petersburg Institute of Informatics, RAS, Russia,

Printed in Bulgaria
Publisher ITHEA®
Sofia, 1000, P.O.B. 775, Bulgaria. www.ithea.org, e-mail: info@foibg.com
Technical editor: Ina Markova

Издател: ИТЕА®, София 1000, ПК 775, България, www.ithea.org, e-mail: info@foibg.com

ISSN 1313-0455 (printed) **ISSN 1313-048X (online)**

AN INVESTIGATION OF ROUTING ALGORITHMS IN SON BY SIMULATION SYSTEM TRIADNS

Elena Zamyatina, Alexander Mikov, Sergei Kalashnikov

***Abstract**: It is well known that mobile ad hoc networks are widespread nowadays. Such networks are created in a short time and function during short time. The number of nodes and interconnections between these nodes change all the time. The algorithms for ad hoc networks management change too. Thus, the software tools and language of the simulation systems must correspond to the dynamically changing elements of the ad hoc system and dynamically changing structure. The paper gives an example of modeling the routing algorithms in ad hoc networks and presents simulation software for the investigation of routing algorithms functioning in ad hoc networks.*

***Keywords**: ad hoc networks, simulation modeling, simulation systems, dynamic geometric graph, routing algorithm*

***ACM Classification Keywords**: I.6 SIMULATION AND MODELING I.6.8 Types of Simulation - Distributed : I.2 ARTIFICIAL INTELLIGENCE I.2.5 Programming Languages and Software - Expert system tools and techniques*

Introduction

Nowadays mobile ad hoc systems are widely used. Such systems are created in a short time and function for a short time. Ad hoc system is adaptive, i.e. customizable and dynamically structured. The dynamic nature of the structure of ad hoc networks corresponds to the agent programming paradigm. This allows to separate tasks: one part of the agents is responsible for solving application problems, and the other - ensures the ad hoc system's operability, solving the operational management tasks of this system.

To effectively solve the ad hoc management problem, an ad hoc model is required. Moreover, unlike ordinary information systems (IS), when the model is used mainly at the design and implementation stage, here the model is needed at the stage of the functioning of the IS.

To effectively solve the ad hoc management problem, an ad hoc model is required. Moreover, unlike ordinary IS, where the model is used mainly at the design and implementation stage, here the model is needed at the stage of the IS functioning.

There are a lot of simulation tools for designing and analyzing computer networks, these simulation systems may be divided into specialized ([COMNET, 2018], [OMNet++,2018], [OPNET,2018], [NS-2,2018] and etc.) and general purpose one ([ANYLOGIC,2018], for example). Some of them allow analyzing networks; others are focused on solving network design problems.

The system of simulation modeling used in this work is the system of computer-aided design and simulation of computer networks TriadNS [Mikov, 2013; Zamyatina E.,2013]. The main feature of this system is the division of the model into layers: a layer of structures, routines (behaviors) and messages. A layer of structures describes nodes of computer network and interconnection between them, the routine layer defines the principle of the behavior of each node and the message layer describes the structure of messages transmitted between nodes. To describe the simulation model, the Triad [Mikov,1995,1987] modeling language is used, the syntactic constructions of which will be described below.

Self-organizing networks (SON)

One approach to classifying wireless communication networks is to divide networks into centralized infrastructures and self-organizing ones. A distinctive feature of self-organizing networks (SON) [Proskochylo A.V.,2015; Belfer R.A., 2012] is the ability, in the absence of a centralized infrastructure, to exchange data for any pair of network nodes in the radio coverage zone.

Nodes in SON can be both end hosts and routers. The connection is organized over long distances with the help of specialized routing protocols in intermediate router nodes. This connection is called "multi-stage or multi-step" (multi-hop). The stage is the participation of one router node in this connection. The nodes of these networks have the ability to find each other and form a network, and in the event of a failure of any node, they can establish new routes for the transmission of messages.

In the class SON there are such networks as:

a) Wireless Mobile Ad Hoc Network (MANET)[Li M.,2017, Liu X.,2017; Ali H.A.,2018];
b) Wireless Sensor Network (WSN) [Grigoriev A.A., 2013];
c) Wireless Mesh Network (WMN) [Vinokurov V.M., 2010];
d) Ubiquitous Sensor Networks (USN)[Koucheryavy A., 2014];
e) Vehicular Ad Hoc Network (VANET)[Boussoufa-Lahlaha S.,2018; Kitsisa R. , 2018; Dharani Kumari D.V., 2017;Cunha F., 2016];
f) Heterogeneous ad hoc networks (HANET) [Trinh B., 2016];
g) Flying Ad Hoc Networks (FANET)[Oubbatia O.S.,2017;Leonov A.V., 2015]

SON-networks can be used in many application areas, such as:

a) the defense systems and the systems of security;
b) environmental monitoring;
c) internet of things;
d) monitoring of an industrial equipment and etc.

As it was mentioned above, the nodes of these networks may find each other and form a network, and in the event of a failure of any node, they can establish new routes for the transmission of messages. So node's coordinates may be changed. The structure of classic wired network may be presented as graph G=(V,E), V – a set of nodes, E – a set of edges (arcs) between nodes. SON-networks can be constructed by using dynamic geometric graphs. Now we'll consider dynamic geometric graphs more precisely.

The dynamic geometric graph

The dynamic geometric graph is a set: $GG = \{V, P, r, S\}$, where V is the set of vertex, P is the mapping $P: V \rightarrow S$ (vertex localization), r is the action radius of the transceivers of computer nodes (graph vertices).

S is an oriented bounded surface on which the vertices are located, and the map P and, possibly, the surface S depend on the time [Mikov A.I., 2015; Mikov A.I., 2017].

In the dynamic geometric graph ad hoc of the network, the node coordinates are, as a rule, continuous functions of time, but the change in the graph $G(t)$ is discrete in all cases, i.e. The dynamic graph reduces to the sequence G_0 , G_1 , G_2 , ..., G_i , ... of static graphs. In the case of static random graphs, the positions of vertices in the graph G_0 are considered random vectors with a given probability distribution. In studies, as a rule, all the vertices are assigned the same distribution, for a limited area of the vertices of the network - uniform.

Dynamic geometric graphs are used in mobile wireless computer networks. In mobile wireless computer networks there is a movement of nodes, as a result of which the conditions for receiving and transmitting signals change. When two nodes approach each other, the signal level becomes sufficient to establish a two-way communications; the connection is lost when the connection is deleted.

The following formulations of problems for dynamic geometric graphs with different constraints are of interest:

Network nodes move in space (on the surface, on the plane). A possible change in the coordinates of a node is determined by the constraints of the environment and the tasks that the node decides.

Environmental constraints:

a) no restrictions, moving is possible in any direction;

b) in the space there is a graph - a network of roads through which nodes can be moved;

c) in the space there are forbidden areas or hard-to-reach areas, to move through which you need permission or special equipment.

In the network, some distributed algorithm D solves the general problem P. All nodes of the network participate in the process of solving the problem. Initially, the network N = N (0) is connected, which allows us to begin the solution of the problem.

As a result of moving nodes at time t1, the network N (t1) becomes disconnected.

There may be several options for continuing:

- by the time t_1 , the algorithm *D* has completed its work, continuation is not required;
- algorithm *D* is not completed and can resume its work at the time t_2 of network connectivity restoration without loss of the work part already done;
- algorithm D is not completed and can resume its work at the time t2 of network connectivity restoration with a partial loss of the work part already done earlier;
- algorithm D is not completed and should begin its work at the time t2 of reconnecting the network again (all intermediate results are lost or cannot be used due to a change in the situation);
- algorithm D continues its work in the time interval from t1 to t2 in the k components of the network, but at the time t2 the partial solutions must be coordinated, to which the time δ (k) is required.

Loss of network connectivity leads to:

- impossibility of solving of the problem P by the algorithm D;
- reducing efficiency, delays in solving problem P by algorithm D;
- a possible loss of individual network nodes N, which turned out to be isolated, and the gradual destruction of network N (for example, during combat operations).

Now let us consider specific features of routing algorithms in ad hoc networks. It is a lot of routing algorithms functioning in classic wired networks and it is clear that centralized routing algorithms in ad hoc networks are inefficient and do not provide the necessary efficiency. Routing algorithms in ad hoc networks may be divided according to *topology* into topological (use information about existing network connections between the nodes of the network) and geographical (use data on the geographical location of nodes). According to *principals of functioning routing algorithms* may be classified as proactive, reactive and hybrid. According to *best route* routing algorithms may be divided into algorithms of distance vector (distance-vector, hop-count) and algorithms with complex route metrics or link-state (use a comprehensive assessment of routes by several parameters)[Mikov A.I.,2015;2017]

As was mentioned above there are several types of ad hoc networks, each of them has features that must be taken into account when developing the routing algorithm. Now we'll consider opportunistic networks and the corresponding algorithms.

Routing algorithms in opportunistic networks

Routing on mailing [Vahdat A., 2000]. Routing based on mailing s deliver the message to its intended purpose by simply distributing it on the entire network. The heuristic that underlies this policy is that, since there is no information on the possible path to the destination or the corresponding node of the next transition, the message should be sent everywhere. It will ultimately reach the destination by transferring from one node to another.

Methods based on distribution are very resource intensive. In addition, due to the large number of transmissions, propagation-based methods suffer from a high level of competition and can potentially lead to network congestion. In order to increase the network capacity, the message propagation radius is usually limited to the introduction of the maximum number of relay transitions for each message or even by limiting the total number of copies of messages present on the network at the same time. If relaying is not allowed further, the node can only send the message directly to the recipient when it meets, if it occurs.

Routing based on network coding. Routing based on network coding [Musolesi M.,2005] also belongs to the category of algorithms based on the dissemination of data, but uses its own approach to limit the flow of messages. Messages before sending are combined (encoded) in nodes. Then the received codes are sent instead of the original messages. Codes are propagated in different directions, as in other routing protocols based on the distribution. The number of generated codes is higher than the number of original messages combined, in order to provide much greater resistance to packet and path losses. Coding is performed at both the source and intermediate nodes.

Context-based routing [Leguay J., 2006]. In Context-Aware Routing (CAR), each node on the network is responsible for creating its own delivery probabilities to each known destination node. There is a periodic exchange of supply probabilities, so that, ultimately, each node can calculate the best carrier for each destination node. The best carriers are calculated based on the context of the nodes.

Among the context attributes required to select the best media, there are, for example, the remaining battery level, the rate of change of connectivity, the probability of being within reach of the destination, the degree of mobility. When the best operator receives a message for forwarding, it saves it in the local buffer and, ultimately, forwards it to the destination node, if available, or to another node with a higher probability of delivery. In fact, CAR provides the basis for calculating the next jumps in opportunistic networks based on the theory of attributes with several attributes applied to the attributes of the general context.

In *MobySpace Routing* [Widmer J., 2005] the site mobility template represents the context information used for routing. The protocol creates a multidimensional Euclidean space called MobySpace, where each axis represents a possible contact between two nodes, and the distance along the axis serves as a measure of the probability of this contact. Two nodes that have similar sets of contacts and probabilities of communication with them are close in MobySpace. The best forwarding node for a message is a node that is as close as possible to the destination in that space. This actually increases the likelihood that the message will ultimately reach the destination. Obviously, in the above-described virtual contact space, knowledge of all axes of space also requires knowledge of all contacts between nodes).

Simulation and computer-aided design system TriadNS

Network simulator TRiadNS was designed and implemented on the base of CAD TRIAD [6, 7]. Let us consider how simulation model is presented in TriadNS. Moreover let us show linguistic constructions aiming for description of computer networks and demonstrate program tools which are useful for computer network design.

Simulation model in TriadNS is represented by several objects functioning according to some scenario and interacting with one another by sending messages. These objects and interconnections between them may be described by linguistic constructions of the level of structure.

The behavior of objects presenting nodes of computer networks may be described in a layer of routines. Third layer is a layer of messages. It is dedicated for a description of messages with complicate structures.

One can describe the structure of a system to be simulated using such a linguistic construction:

structure <name of structure> ***def*** (<a list of generic parameters>) (<a list of input and output parameters>) <a list of variables description> <statements>) ***endstr***

Input and output parameters may be associated with input and output poles of an object presenting computer node of network. Indeed every computer node may send messages through output poles to another node and receive messages through input poles. Each object may be described with the structure of lower layer. Thus, the simulation model is a hierarchical one.

A structure of computer network may be built using statements: add or delete a node, add or delete an edge (arc), poles, union or intersection of graphs. Structure may be built using graph constants.

Let us present example of structure description which one can see at fig.1.:

Structure Net1 (***integer*** number_of_nodes) def

Net1:= ***compl***(number_of_nodes)

(V[1:number_of_nodes])(P[1:number_of_nodes-1])

endstr

Thus a designer of network presented at fig.1. may use several linguistic constructions.

Model M

…….

M := Net1(4);

….

Endmod

This example demonstrates the description of structure with the help of Triad language. Program unit «structure» is a procedure with parameters. Graph constants allow to build graphs with an arbitrary number of nodes. A designer has the ability to quickly change the number of nodes by specifying it as a parameter.

Simulation system TriadNS provides text and graphic editors to the designer. The graph presenting network at fig.1. was designed with the help of a graphical editor.

Special algorithm (named "routine") defines the behavior of an object. Each routine is specified by a set of events (*E*-set), the linearly ordered set of time moments (*T*-set), and a set of states {*Q*-set}. State is specified by the local variable values. Local variables are defined in routine. The state is changed if an event occurs only. One event schedules another event. Routine (as an object) has input and output poles (Pr_{in} and Pr_{out}). An input pole serves to receive messages, output – to send them. One can pick out input event e_{in}. All the input poles are processed by an input event, an output poles – by the other (usual) events.

The special statement out (***out*** <сообщение> ***through*** <имя полюса>) serves for sending a message.

A set of routines defines a routine's layer.

Let us present a fragment of routine describing a behavior of computer nodes implementing the routing algorithm based on mailing.

Routine Every_Node_Behavior (…..

….

Event It_is_MyCoordinates;

//передача координат

Cord[0]:=id; //node's coordinate}

Cord[1]:=X;

Cord[2]:=Y;

out RealArrayToStr(Cord); //send message to all neighbors//

ende

…

endrout

It is very important to assess the simulation model and it is behavior in defined conditions and in accordance to appropriate restrictions.

Special program component – the condition of simulation defines the scenario of simulation experiment, the criterions of simulation run termination, a list of simulation model elements (the variables, the events, the input and output messages) which are have to be examined and processed during simulation run with the help of the information procedures and the scenarios of the completing processing of the results of these information procedures.

A component "***conditions of simulation***" contains a set of another programs. This programs - "***information procedures***" needing for data collection and processing of data during simulation run. Conditions of simulation and information of procedures present an "algorithm of investigation". Simulation system Triad includes a library of standard information procedures but an investigator may describe an information procedure (and conditions of simulation too) with the help of Triad language:

The investigation of the structure layer only is static process. The simulation process may take place only after the definition of the behavior of all nodes of model in structure layer. As it was noted above the behavior is determined by the statement Put. It is well known that a simulation is a set of object functioning according to some definite scenarios controlled by synchronizing algorithm. The simulation run is initialized by the statement simulate:

Simulate <a list of models> ***on condition of simulation*** <имя условия моделирования>(<настроечные параметры>) (<параметры интерфейса>) (<список информационных процедур>; <последовательность операторов>).

One can pay an attention to the fact that the several models may be simulated under the same conditions of simulation simultaneously.

Moreover it is possible to define several parts of linguistic construction “conditions of simulation” in the statement simulate. It is rather important. Let us suppose that an investigator wish to design computer network, so it is necessary to define the structure of computer network and characteristics of workstations and other devices of computer network. Let us suppose that an investigator want to define the configuration of computer network and technical characteristics of cheapest network, of more efficient network, of more secure one and etc. So it is necessary to define different criteria during simulation run and one can do it using different “conditions of simulation” in statement simulate.

Let us show the ability of TriadNS for a design and analyses of routing algorithm based on mailing.

Implementation of the routing algorithm of opportunistic networks in TiadNS

So, the TriadNS modeling system was chosen to implement the routing algorithm for opportunistic networks.

Let us remember the algorithm based on mailing.

Each node, sending its message from a specific node, makes a dispatch to all the nearest nodes. The closest nodes perform the same actions, excluding the initial sending node from the transfer, and the node from which the message was sent, if it were different nodes. Thus, at a certain point in time, all copies of the message must be at the recipient. If copies do not reach the recipient, they are destroyed after the expiration of their life. The lifetime is the number of possible message forwarding of messages. The lifetime is specified in the incoming parameters and must necessarily be common for all nodes.

Let's present the results of algorithm testing.

The behavior of an element can be specified through the context menu, all possible behaviors are determined from the ontology. All the instances of the routine of this semantic type go there. Each routine has a number of parameters that the user can change after applying a routine to the element.

Thus it is necessary to define input parameters. One may define the coordinates of concrete node, identifier of this node, the radius of visibility, message lifetime.

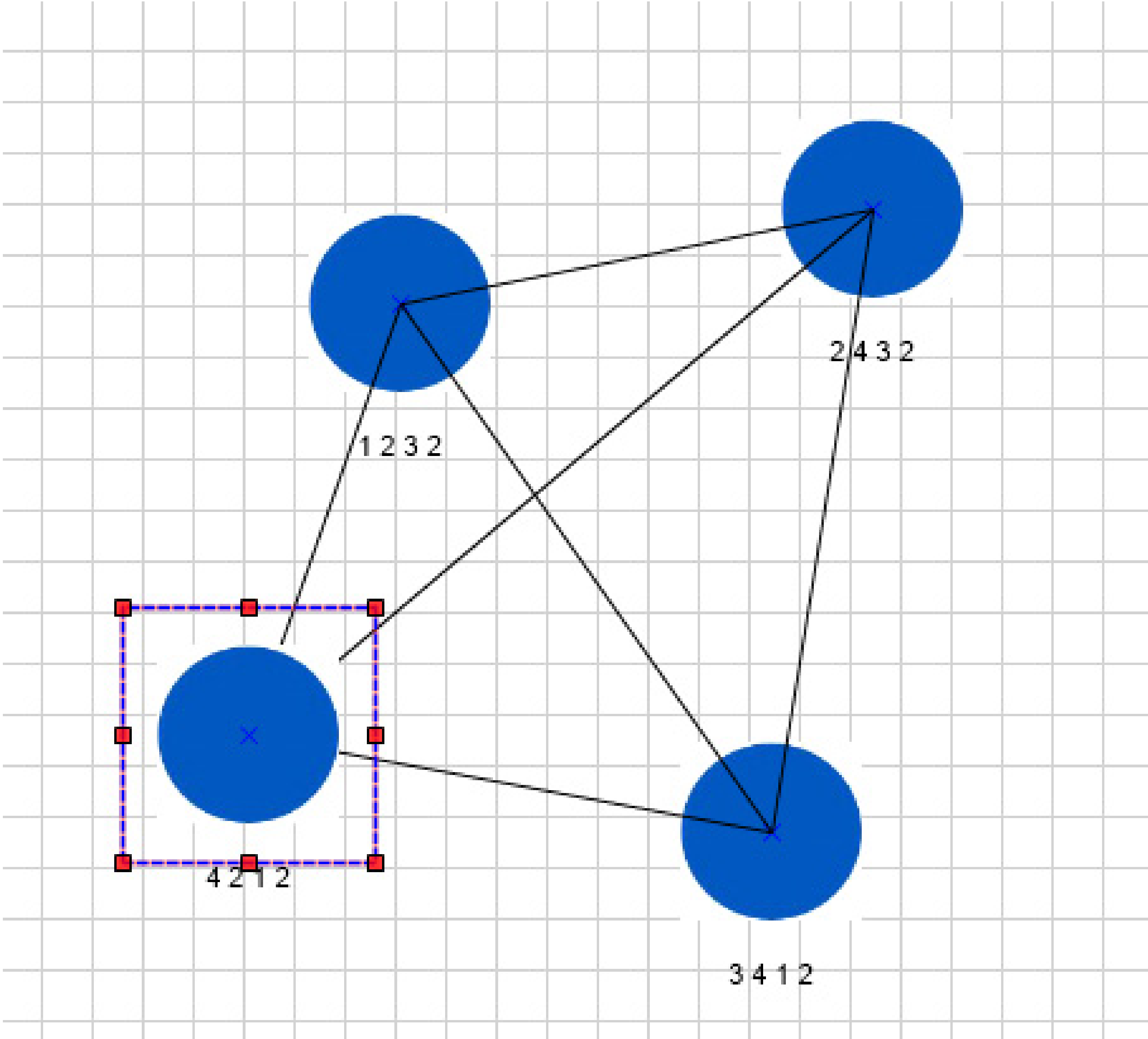
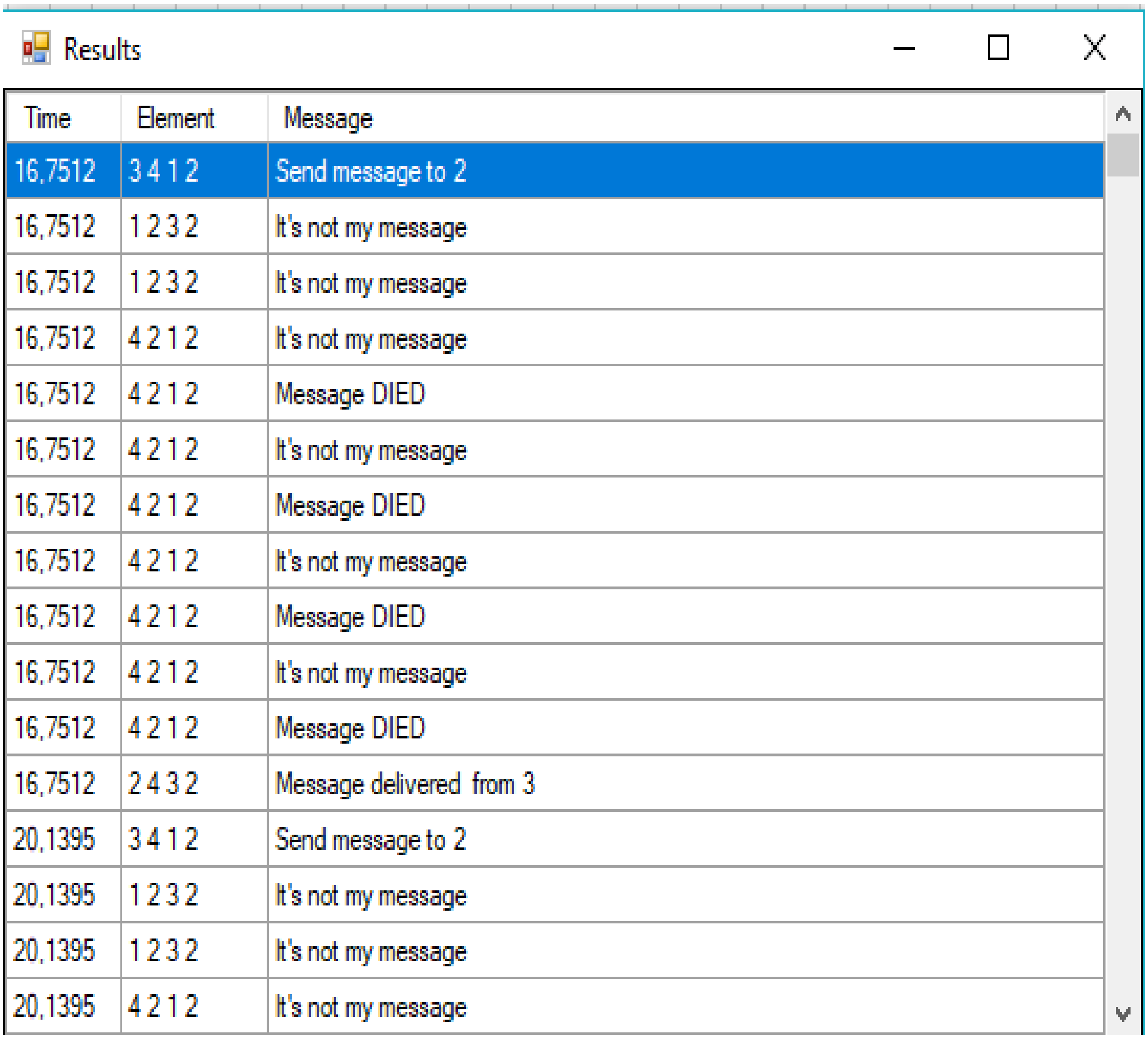


Fig. 1. Test №1. Tracing messages when executing the algorithm based on mailing

First test is very simple. There are four nodes which radius of visibility allows them to reach only two other nodes. The message lifetime in this case is only one forwarding.One may see results below.



Time	Element	Message
16,7512	3 4 1 2	Send message to 2
16,7512	1 2 3 2	It's not my message
16,7512	1 2 3 2	It's not my message
16,7512	4 2 1 2	It's not my message
16,7512	4 2 1 2	Message DIED
16,7512	4 2 1 2	It's not my message
16,7512	4 2 1 2	Message DIED
16,7512	4 2 1 2	It's not my message
16,7512	4 2 1 2	Message DIED
16,7512	4 2 1 2	It's not my message
16,7512	4 2 1 2	Message DIED
16,7512	2 4 3 2	Message delivered from 3
20,1395	3 4 1 2	Send message to 2
20,1395	1 2 3 2	It's not my message
20,1395	1 2 3 2	It's not my message
20,1395	4 2 1 2	It's not my message

Fig. 2. Test №1. The results of algorithm

The results of test №1: there are a lot of "dead" messages. Only a small fraction of messages were delivered to the right place.

The second test is more complicate. In this test we have 8 nodes, but now each node sees three other nodes. The lifetime of the message remains the same as in the previous example.

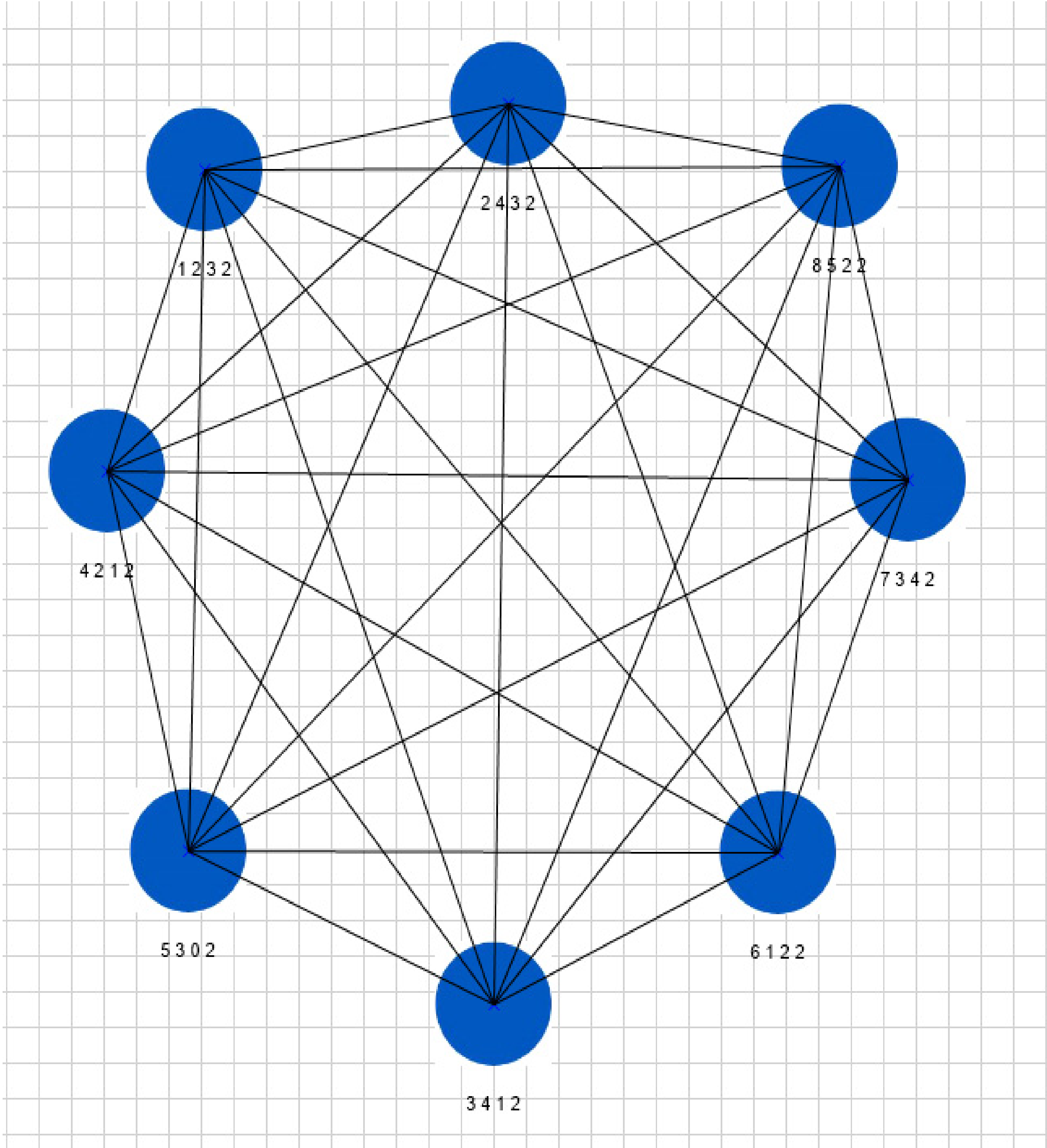
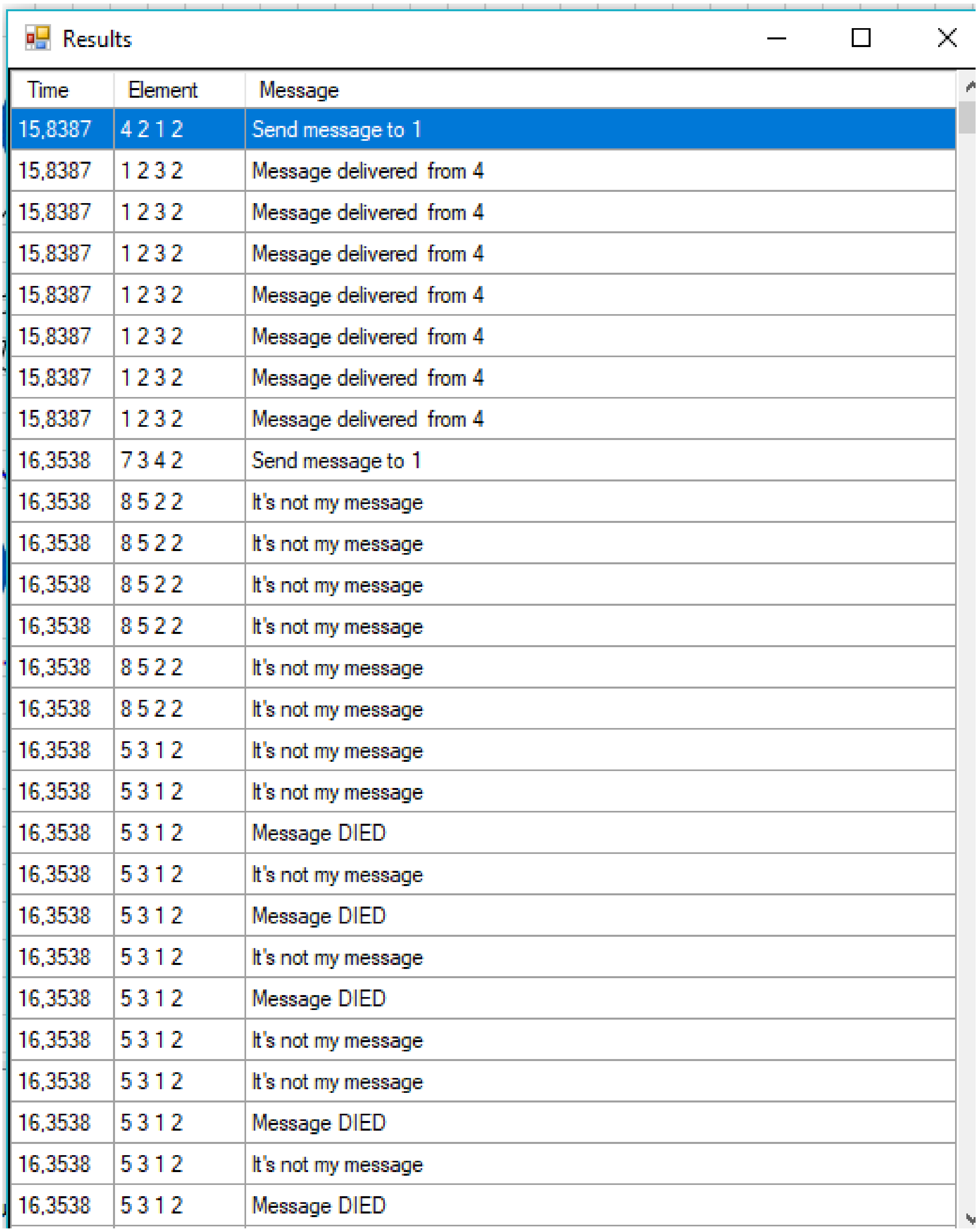


Fig. 3. Test №2. Tracing messages when executing the algorithm based on mailing. 8 nodes.

At the conclusion, the gap between the sending of the message and its arrival increased, as well as the number of destroyed messages.



Results

Time	Element	Message
15,8387	4 2 1 2	Send message to 1
15,8387	1 2 3 2	Message delivered from 4
15,8387	1 2 3 2	Message delivered from 4
15,8387	1 2 3 2	Message delivered from 4
15,8387	1 2 3 2	Message delivered from 4
15,8387	1 2 3 2	Message delivered from 4
15,8387	1 2 3 2	Message delivered from 4
15,8387	1 2 3 2	Message delivered from 4
16,3538	7 3 4 2	Send message to 1
16,3538	8 5 2 2	It's not my message
16,3538	8 5 2 2	It's not my message
16,3538	8 5 2 2	It's not my message
16,3538	8 5 2 2	It's not my message
16,3538	8 5 2 2	It's not my message
16,3538	8 5 2 2	It's not my message
16,3538	5 3 1 2	It's not my message
16,3538	5 3 1 2	It's not my message
16,3538	5 3 1 2	Message DIED
16,3538	5 3 1 2	It's not my message
16,3538	5 3 1 2	Message DIED
16,3538	5 3 1 2	It's not my message
16,3538	5 3 1 2	Message DIED
16,3538	5 3 1 2	It's not my message
16,3538	5 3 1 2	It's not my message
16,3538	5 3 1 2	Message DIED
16,3538	5 3 1 2	It's not my message
16,3538	5 3 1 2	Message DIED

Fig. 4. Test №3. The results in log. The lifetime of message is 2 forwarding

One may see the results of third test on fig.4.

The number of nodes was the same that it was in second test, but the lifetime of messages was increased and was equal to the time of two forwarding.

Judging by the results log, the number of messages sent from one node to another multiplies when the life of messages increases.

On the basis of three tests, one can say that if one want to speed up and make sure that messages are delivered when you increase the nodes, you should increase the message's life cycle, but you may encounter multiple deliveries in different directions.

Aknowledgement

The work was supported by RFBR and the Administration of Krasnodar Region (grant 16-47-230336).

Conclusion

So, this paper briefly presents computer networks of a new generation: self-organizing networks (SON) and several algorithms dedicated for self-organizing networks management.

The main idea of this paper – to present the high-level language Triad for a description of routing algorithms in SON and consider flexible and effective software simulation system TriadNS.

SON is widely used now, so it is very important to have softwarem tools for the investigation of these networks.

References

[COMNET III] COMNET III. Planning for Network Managers. Release 1.3. Available from: http://eent3.lsbu.ac.uk/staff/baoyb/acs/Comnet/ comnet%20III.pdf (Accessed: 10.06.18)

[OMNeT++] OMNeT++ Community Site. Available from: http://www.omnetpp.org (Accessed: 10.06.18)

[OPNET] OPNET. http://www.riverbed.com/products/performance-management-control/opnet.html?redirect =opnet (Accessed: 10.06.18)

[ANYLOGIC] ANYLOGIC. Available from: www.anylogic.ru. (Accessed: 10.06.18)

[Mikov A.] Mikov A., Zamiatina E. Program Tools and Language for Networks Simulation and Analysis. // In: Proceedings SDN & NFV – The Next Generation of Computational Infrastructure: 2014 International Science and Technology Conference «Modern Networking Technologies (MoNeTec)», October 27-29, 2014, Lomonosov Moscow State University pp. 94-102

[Zamyatina E.] Zamyatina E., Mikov A., Mikheev R. Linguistic and Program Tools For Debugging and Testing Of Simulation Models Of Computer Networks // In: International Journal "Information Models and Analyses". 2013. Vol. 2. No. 1. P. 70-80.

[Mikov.A.I.] Mikov A.I. Simulation and Design of Hardware and Software with Triad // In: Proc.2nd Intl. Conf. on Electronic Hardware Description Languages. Las Vegas, USA. 1995. P. 15 20.

[Mikov A.I.] Mikov A.I. Avtomatizatsiya sinteza mikroprocessornyh upravlyaushih system. // Irkutsk: izdatelstvo Irkut. un-ta, 1987.

[Proskochylo A.V.] Proskochylo A.V., Vorobyov A.V., Zriakhov M.S., Kravchuk A.S.. Analysis of state and development perspectives of self-organizing networks//Nauchnye vedomosti, Seriya Ekonomika.Informatika.2015.№19(216).Vypusk 36/1, 177-187

[Belfer R.A.] Belfer R.A.. Seti SON. //M.: Material is Nacionalnoy biblioteki im. N.E.Baumana, 2012. https://ru.bmstu.wiki/Seti_SON. (Accessed: 10.06.18)

[Oubbatia O.S.] Oubbatia O.S. , Lakas A., Zhouc F., Güne M. ¸ Yagoubia M.B.. A survey on position-based routing protocols for Flying Adhoc Networks (FANETs)// Vehicular Communications, 10, Elsevier, 2017, pp. 29–56

[Qiu T.] Qiu T., Chen N., Li K., Qiao D. , Fu Z.. Heterogeneous ad hoc networks: Architectures, advances and challenges/Elsevier Ad Hoc Networks, 55, Elsevier, 2017, 143–152.

[Koucheryavy A.] Koucheryavy A.. State of the Art and Research Challenges for USN Traffic Flow Models //16th International Conference on Advanced Communication Technology (ICACT). 2014. pp. 336–340.

[Trinh B.] Trinh B., Paramonov A.. Research of parameters of routes in self-organizing networks. Information technologies and telecommuncatons. V. 4. № 4.- Информационные технологии и телекоммуникации. 2016, SPbSUT, pp.81-88.

[Li M.] Li M., Salmanian M., Willink T.J. . Network layer connectivity awareness with application to investigate the OLSR protocol in tactical MANETs.//In Proceedings of the 2017 Winter Simulation Conference, W.K.V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds. pp.4093-4105

[Liu X.] Liu X. , Li Z., Yang P., Dong Y. Information-centric mobile ad hoc networks and content routing: A survey// Ad Hoc Networks 58, Elsevier, 2017. pp. 255–268.

[Ali H.A.] Ali H.A., Areed M.F. , Elewely D. . An on-demand power and load-aware multi-path node-disjoint source routing scheme implementation using NS-2 for mobile ad-hoc networks// Simulation Modelling Practice and Theory. 80. Elsevier, 2018 pp.50–65

[Boussoufa-Lahlaha S.] Boussoufa-Lahlaha S. , Semchedinea F. , Bouallouche-Medjkounea L. . Geographic routing protocols for Vehicular Ad hoc NETworks (VANETs): A survey// Vehicular Communications, 11, Elsevier, 2018, pp. 20–31

[Kitsisa R.] Kitsisa R. , Dattaa S.. Layer 3 Enhancements for Vehicular Ad Hoc Networks//In Proceedings of the 9th International Conference on Ambient Systems, Networks and Technologies (ANT 2018), Elsevier. Procedia Computer Science, 130, (2018), pp. 628–635

[Dharani Kumari N.V.], Shylaja B.S.. AMGRP: AHP-based Multimetric Geographical Routing Protocol for Urban environment of VANETs//Journal of King Saud University – Computer and Information Sciences, 2017, http://dx.doi.org/10.1016/j.jksuci.2017.01.001

[Cunha F.] Cunha F., Villas L., Boukerche A., Maia G., Viana A., Mini R., ,Antonio A. Loureiro F.. Data communication in VANETs: Protocols, applications and challenges//Ad Hoc Networks, 44, Elsevier, 2016, pp. 90–103

[Grigoriev A.A.] Grigoriev A.A. Algoritmy marhrutizacii v opportunisticheskikh setyakh//Trudy MFTI, 2013,T. 5, №3, 112-120

[Vinokurov V.M.] Vinokurov V.M., Pugovkin A.V., Pshennikov A.A. , Usharova D.N. , Filatov A.S.. Marshrutizaciya v besprovodnykh setyah//Trudy TUSUR, №2 (22), 2010, 288-292

[Leonov A.V.] Leonov A.V, Chaplyshkin V.A.. Seti Fanet. Omskiyi nauchnyi vestnik №3 (143) 2015, 297-301

[Mikov A.I.] Mikov A.I.. Svyasnost autonomnyh besprovodnyh kompiuternyh setey v mesnostyah s plohoi infrastrukturoy.//Ekologicheskii vestnik nauchnyh centrov CHES, 2015, pp.70-75.

[Mikov A.I.] Mikov A.I.. Dinamicheskie geometricheskie graphy mobilnyh ad hoc setey. // Informatizacia i svyaz. 2017. № 2. C. 66-70.

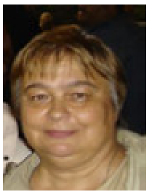
[Vahdat A.] Vahdat A., Becker D. Epidemic routing for partially connected ad hoc networks : Tech. Rep. CS-2000-06 //Department of Computer Science, Duke University. – Durham, NC, 2000.

[Musolesi M.] Musolesi M., S. Hailes, C. Mascolo. Adaptive Routing in Intermittently Connected Mobile Ad Hoc Networks. // Proceedings of the 6th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2005). – Naxos, 2005.

[Leguay J.] Leguay J., Friedman T., Conan V.. Evaluating Mobility Pattern Space Routing for DTNs. // Proceedings of the IEEE Infocom 2006. – Barcelona, 2006, pp.1-10.

[Widmer J.] Widmer J., Le Boudec J.-Y.. Network Coding for Efficient Communication in Extreme Networks // Proceedings of ACM SIGCOMM 2005 Workshop on delay tolerant networks. – Philadelphia, 2005.

Authors' Information



Elena Zamyatina *– Perm State National Researching University, Associated Professor, PhD,Bukireva,15,Perm,Russian Federation; e-mail: e_zamyatina@mail.ru*

Fields of Scientific Research: Simulation, Distributed and Parallel Simulation, Artificial Intelligence, Computer Networks Simulation

Alexander Mikov *– Cuban State Conference, Full Professor, Stavropolskaya, 149, Krasnodar,Russian Federation; e-mail: alexander_mikov@mail.ru*

Fields of Scientific Research: Simulation, Distributed and Parallel Simulation, Artificial Intelligence, Computer Networks Simulation, Graph Theory

Sergei Kalashnikov - *Perm State National Researching University, Student, Bukireva,15,Perm,Russian Federation; e-mail:* ser.00@list.ru

Fields of Scientific Research: Simulation, Artificial Intelligence, Computer Networks Simulation, Ontologies