Journal of Computer Sciences 1 (2): 215-220, 2005
ISSN1549-3636

An Adaptive Routing Algorithm for Ad hoc Mobile Networks

Hamed S. Al-Bdour
Department of Information Technology, Mutah University, Karak, Jordan

Abstract: This research has examined the issues of routing in multi-hop wireless networks of mobile nodes without the need for any existing network infrastructure or administration. This is called ad hoc. The key point of this research article are selecting routing protocols which adapts quickly to routing changes when host movement is frequent, yet requires little or no overhead during periods in which hosts move less frequently. This research article considers different methods of optimization of selected algorithm. The Dynamic Source Routing is a simple and efficient routing protocol designed specifically for use in multi-hop wireless ad hoc networks of mobile nodes. The DSR protocol is designed mainly for networks of up to about two hundred nodes. It works well with even very high rates of mobility.

Key words: Routing Algorithm, Dynamic Source Routing Protocol, Mobile Nodes

INTRODUCTION

Today, mobile subscriber systems (SsS) and hardware for organization of wireless access to networks are becoming rather accessible for their broad propagation. Frequently, people, who move constantly, need to establish a connection in a condition where the fixed wired infrastructure is unavailable. For example, friends or business partners at the meeting need to receive shared access to some files or a rescue group which should be deployed fast in extreme conditions. In such situations, many mobile SsS should ensure standalone operation without using any fixed infrastructure or centralized administration. Such wireless networks are known as ad hoc. The routing tools in such networks are subject to additional requirements. This problem is so different from the classic problem of routing that it requires separate studies and development of new algorithms for its solution.

The purpose of this research study is to consider different algorithm to solve the routing problem on the condition of providing its operation without using the fixed wired infrastructure and centralized administration, and also to select the specific algorithm to solve the stated problems and investigation of possible alternatives of its optimization.

Wireless mobile networks are modern technology, allowing to receive access to information and services by using wireless interfaces beyond the dependence upon geographical provisions of the subscriber systems. Such networks may be divided into two types: structured and non-structured (special, ad hoc networks). Mobile SsS in structured network interact by using a base station, located in the operation area of their transceivers. Mobile SsS using communication may move in the borders of certain territory. When they leave the operation area of one base station, they connect to another. In this case the base stations have fixed location and are connected by terrestrial or wireless communication channels.

In ad hoc networks, all nodes are mobile and may be connected dynamically in absolutely arbitrary manner. All nodes of these networks act as routers and participate in detection and support of routes leading to other nodes. When the network has only two SsS nearby located, this requires no means to solve the routing problem. However, frequently, there should establish a connection SsS located beyond the operation area of the one another's transceivers, for this purpose it is necessary that SsS located between them, also participate in ad hoc networks and sent packets for them.

Review and Analysis of the Existing Solutions: The routing protocols for ad hoc may be divided into two core categories: table-based protocols and on-demand routing protocols; this split is based on when and how the route is discovered. In the table-based protocols, matched and up-to-date information is transmitted to each node, opposite to on-demand routing, when the route is established only when it is necessary for the sender. The table-based protocol usually maintains a routing table for the whole network, and the on-demand protocol study routes as far as necessary.

Table-based Routing: In table-based routing protocol, each node maintains one or more tables, containing route information to each other node on the network. All nodes update these tables so that to ensure new view of the network. When the network topology changes, the nodes propagate updating messages throughout the network so that to save the new routing information on the whole network. This class includes DSDV [1], WRP [2], CGSR [2], GSR, FSR, HSR, and ZHLS. They differ in network organization, distributed information and set of routing tables stored.

The majority of table-based protocols works well in relation to route optimality. However, this usually means rather high scope of control information for provision of the exact information on the network. As soon as changes in the connection status take place, they are notified immediately. In a large network with rather fast nodes, the scope of control information would "starve" the network. Hierarchical network structure may be used to limit the control traffic; however, there are some costs of its creation and maintaining, as this should be made using a dynamic method.

DSDV[1] is a distance vector routing protocol, based on usual routing protocol RIP adapted for ad hoc networks. Routing is made by using tables, stored in each node. The fundamental complexity DSDV lies in generating and maintenance of these broadcasting tables sent, as loops may arise. DSDV requires periodic transmission of packets updating the routing table, independently of the network traffic. The overhead is the core shortcoming of DSDV. In addition, DSDV is non-functional in case of changes in topology until updating packet propagate across the network. DSDV fits well for small networks where changes in the topology are limited.

On-demand Routing: These protocols use "lazy" approach towards routing. Routes are established when required. When a source wishes to send a packet to the target, it invokes the route discovery mechanisms in order to find a path to the target. The route remains in effect, until the target is reachable. This class includes AODV [3], DSR [4, 5], TORA [6], CBRP [7], ABR [8], SSR. They differ in methods for packet propagation, navigation and gathering the routing information, interaction with the neighbors and route maintenance mechanism.

AODV and DSR use similar mechanism for route search. However, DSR [5] includes into each packet complete routing information and saves the complete route in the host's memory. AODV [3] generates much more control traffic in comparison with DSR. Therefore, in small networks DSR is more efficient, however in case of growth in traffic, the efficiency of AODV grows too. DSR reaches the best results in comparison with AODV at 20 traffic sources and their rather high mobility [4]. Therefore, we should note that DSR should be used for networks with limited number of nodes, where overhead of packet delivery should be minimal and AODV shall be used for networks with many hosts.

An advantage DSR over some other on-demand protocols is that it requires no periodic routing notifications whereby saving bandwidth and reducing power consumption. Hence, the protocol does not cause any overhead, when no changes is made to the network topology. A specific feature of DSR is possibility to store in the nodes' caches many routes to any node. Hence, when the first route does not work, the source may find in its cache another suitable route. In this case, the route is restored faster than in many other on-demand protocols.

TORA is one of largest and most complex routing protocols for ad hoc [6]. Each node should maintain a structure describing the rank of the node and status of all connections. High requirements are presented to processor and bandwidth, as each node should be in constant coordination with the adjacent nodes, to discover changes in the topology. As with DSDV, loops may arise during the network response on changes in the topology. TORA suits better to large, dense node clusters with very low mobility. TORA is encumbered by hierarchical view of the network that results in significant congestion during the response to nodes movement.

Comparing DSDV, TORA, DSR, and AODV [2, 9], it was detected that DSDV and DSR are capable of finding routes, closest to optimum. A lot of studies was performed to compare the efficiency of DSR vs. AODV, DSDV, TORA, and it was established that DSR is more efficient [2, 9]. TORA is believed to suit large networks with dense arrangement of nodes. However, for small networks composed by several dozens of relatively fast moving mobile nodes, TORA has the highest number of control packets. ABR [8] should be separately mentioned as it uses a degree of association stability to provide metrics for route selection. Hence, perhaps ABR is capable of selecting routes having higher stability than routes selected using any other of the mentioned protocols.

Investigation of the Methods for Boosting the Efficiency of the Dynamic Source Routing Algorithm: So, traditional routing algorithms, using distance vector concept or connection status concept do not work well for ad hoc networks, as they do not provide sufficient efficiency in case of route changes in the network, caused by movement of subscriber systems. Another shortcoming of them is a necessity to transmit regular updates of the routing information even at minor movement of the subscriber systems. In the algorithms using distance vector, each router transmits to all the neighbors a distance vector from it to all subscriber systems known to it and each router calculates the shortest path to each SsS based on this information. Using connection status, each router transmits instead to all other routers on the network a set statuses for all its connections, however each router, to calculate the shortest distance to each SsS, requires complete information on the network topology generated based on the connection statuses of all routers. At the same time, the service traffic in them grows sharply at SsS movement, using bandwidth too uneconomically. This results in excessive consumption of SsS batteries.

Source routing is a routing procedure in which the packet's sender determines the completed sequence of nodes through which it is necessary to transmit a packet; the sender explicitly sets this route in the packet header, identifying each transition by the address of the next node, so that, as a result, to transmit a packet to the target subscriber system. Source routing was used in a number of routing problems in wired networks, using statically determined or dynamically created routes from the source. The Dynamic Source Routing [5] protocol is intended directly for using in wireless ad hoc networks.

Algorithm Description: Dynamic Source Routing protocol delivers a number of potential advantages over the known routing protocols such as distance vector for ad hoc networks. DSR may quickly adapt to SsS movement. At the same time, contrary to the traditional routing protocol, the protocol under consideration uses no periodic notification of routing information thus reducing the network bandwidth overhead, especially when the network has low or minor SsS mobility. At the same time, the power supply battery life of mobile SsS is extended. Each subscriber system in case of necessity may dynamically define a route to any SsS as a result of the route discovery initiated. Using distance vector algorithms, redundant paths boost updating of the routing information and require central processor overhead for their processing, while DSR has no this shortcoming.

In wireless environment, network transmission between two SsS does not necessarily work equally well in both directions, due to differences in propagation or interference around these two SsS. For instance, distance vector routing algorithm may cause a situation that a subscriber system may receive routing messages from another mobile SsS, however the packet transmitted in reverse direction may not reach the target. The protocol under consideration does not require possibility of bidirectional transmission between subscriber systems, though it may be used when media access control protocol used guarantee it.

In DSR, in order to send a packet to another subscriber system, the sender builds the source route in the packet header, setting the addresses of all SsS in the network via which the packet should be transmitted so that, as a result, to reach the target. The sender performs only the first sending and transmits the packet to the first node, preset in the initial route. When a subscriber system has received a packet, if it is not the ultimate target, it simply transmits a packet to the next transition preset by the initial route in the packet header. When the packet reaches the ultimate target, it goes to network layer software of this SsS.

Each mobile SsS on the wireless network maintains a route cache which caches routes from the source. When one subscriber system sends a packet to another SsS, the sender first looks up in the route cache the route from the source to the target. If the route is found, the sender uses this route for packet transmission. If no route is found, the sender may try to open it, using route discovery protocol. Waiting for the route discovery, SsS may keep current processing and may send and receive packets from other SsS. Subscriber systems may buffer initial packet so that to transmit it, as soon as the route is investigated upon opening the route or to delay the packet, based on higher level protocol software and, in case of necessity, to retransmit it. Each record in the route cache is related with the period of its storage upon which the record will be deleted from the cache.

Opening the route allows any subscriber system on the ad hoc networks to dynamically discover routes to any accessible SsS on the same network. For this purpose, route request packets identifying the target of opening the route, are used. Except for the request address initiator and the target of the request, each route request packet contains route record which accumulates a sequence of transitions recorded in the route request packet, as far as it propagates across the wireless network in the course of route discovery. Each route request packet also contains a unique request identifier set by the initiator. For detection of twice received requests, each SsS uses the list of recently received pairs (initiator address, request identifier).

SsS, initiating the route discovery, transmits a route request packet to the next SsS. Each of them propagates the request, if it is not its target and if this request is not redundant. The request is denied, if the address of this subscriber system is already listed in the route records, that guarantees impossibility of repeated propagation of copies of any request. In addition, the request is denied if the subscriber system has already received it recently with the same properties, in this case later copies of the request arriving using another route, are deleted. Therefore, the route request propagates across the network, until the target SsS is reached, which responds to the initiator, communicating the route. If the route discovery is successful, initial SsS receives the route reply packet, containing a sequence of network transitions. If the network maintains identical quality of service in both directions, then reply uses the reverse route. Otherwise, in order to return the route reply packet, the target SsS should maintain a route to the initiator. If the record for this target in is its route cache, then the route reply packet may be sent by using this route. Otherwise, this packet is used as a request to receive the reverse route.

At the time of using any source route by the subscriber system, it keeps checking the correctness of performance of this route. For instance, if the sender, the target, or any other SsS, being a part of the route is beyond the coverage area of wireless transmission of the next or previous chain in the route, the route may not be used to reach the target any more. In addition, the route will also be non-functional, if any subscriber system in the route is damaged or switched off. This monitoring of route performance correctness is called route maintenance. When in the process of route maintenance there is a problem with using the route, the route discovery may be re-initiated to discover new correct route to the target.

In the traditional routing protocols, the route discovery and route maintenance are incorporated and for their performance, periodic route updating is used. In the long run, periodic updating will propagate changes to all routers that would result in calculation of a new route. DSR requires no periodic exchange with routing information. Instead, the route maintenance procedure checks operation of the route at the time of using it and informs the transmitter on any originating errors.

As wireless network are inherently less reliable than wired networks, they frequently use hop-by-hop acknowledgement at the channel layer, to ensure early detection and exclude repeated transmission of lost or destroyed packets. In such cases, the route maintenance may be easily organized; both for each transition, the transmitting subscriber system may determine its operability. If the channel layer informs on a transmission problem which it may not handle the SsS send a route error packet to the initial transmitter of the packet whose transmission caused error. A route error packet contains the addresses of the subscriber systems on the both sides of the wrong transition: the system detected the error and the system tried to transmit the packet. Having received the error packet, wrong transition is deleted from the route cache of the sender: all routes containing this transition should be reduced at that point.

If the low level acknowledgement is not supported, the equivalent signal may be provided at higher level. There may be used replies of transport or application level and acknowledgement from the final target. In case of necessity, the packet header may set a bit requesting obvious acknowledgement from the recipient for this transition. If an acknowledgement signal is not received for some period of time, SsS may without extra costs establish the status of this transition of the route.

If wireless network interfaces or medium in which they are used, is unable to work with identical quality in both directions, the route maintenance may be provided by using end-to-end acknowledgement instead. Maintenance is possible, if some route exists between ultimate SsS (there are possible various routes in each direction). In order to signal the information on route status, the replies of transport or application level or acknowledgement from the final target may also be used. The transmitter may only make a conclusion that at least one transition in the route is wrong, being unable to identify it.

Algorithm Optimization: This study investigates the following optimization methods for the base algorithm:

* Extended caching and extended use of cache.
* Improved error handling
* Load balancing
* Route change during transmission
* Route change notification

Based on the suggested optimization methods, we modeled three versions of the algorithm:

* Base algorithm;
* Optimized algorithm 1 (is distinguished by extended caching and extended use of cache and improved error handling), these changes are suggested in [10];
* Optimized algorithm 2 (is distinguished from the previous in using load balancing, route change opportunity during transmission and use of the route change notifications in this case) these changes are suggested in this article.

Let us consider the researched optimization methods in more detail.

Extended Caching and Extended use of Cache: A subscriber system may use own route cache so that to avoid propagation of the route request packet received from another system. Let us assume that an SsS receives a route request packet for which it is not a target and it not is listed in the packet routing record and pairs for it (initiator address, the request identifier) are not found in the list of the recently noted system requests; if the route cache has a record for the target of the request, the system may add this cached route to the accumulated routing record in the packet and then return the obtained route in the route reply packet to the initiator not resorting to propagation (relaying) of the route request. However, in this case there is a particular problem when a number of mobile systems receive from the initiator broadcasting transmission with the route request packet, and all respond based on the routes found in their route caches, especially if more than two mobile subscriber systems in such situation. Such simultaneous reply from the cache from several SsS may cause collision of some of these reply packets and local overload in the wireless network. In addition, various replies may specify routes of various length.

A problem of many simultaneous replies may be avoided and replies pointing to longer routes may be eliminated having forced each mobile SsS to execute a small delay before reply from the cache. Before reply from the route cache, SsS does the following:

Selects the period of delay $d = H * (h-1 + r)$, where h is the number of network transitions for the route which will be returned to the system in this reply, r is the casual number between 0 and 1, and H is the minimum value of delay.

Delay in the route reply transmission by this system for time d.

Within this period of delay, listens to all packet for the initial subscriber system.

If in this period of delay, this system receives a packet addressed by the target of this route discovery (the target is the ultimate destination address for the packet across any sequence of intermediate hops), and if the route length for this packet is less than h, the delay is canceled, and the route reply with this system is not transmitted; this system makes a conclusion that the initiator of this route discovery has already received the route reply ensuring equal or better route.

Extended Error Handling: Let two systems, which should establish a connection, are beyond the coverage area of the one another's transmitters, and there are no sufficient other SsS between them to generate a transition sequence. If we initiate new route discovery for each packet, sent by the system in such situation then there will propagate a lot of redundant route request packets in the sub-network accessible from this system. In order to reduce losses from such route discovery, there may be used delay for limiting the discovery rate of the new routes from this subscriber system to the same target. If SsS attempts to send additional data packets to one and the same system more frequently than set in this limit, the subsequent packets will be buffered until the route reply is received and opening of the new route will not be initiated until the minimum allowable interval between the new route discovery to this the target has expired. Additional improvement of error handling is possible by using listening mode allowing the system to listen to packet route errors sent by other systems. As the route error packet identifies both nodes in transition causing the error, any system received the packet may update the cache for reflection of this fact. The system receiving route error packets may simply look up in its route cache all routes using this transition and to exclude this transition in each of routes so found.

Load Balancing: At extended route caching in dynamic source routing algorithm, several routes to the same target may be in the cache. However, the base algorithm suggests that there should be used only one route with minimum length; at the same time the first arbitrary shall be selected from several routes with the same length. Such approach allows improving the situation with rerouting time minimization; however it does not take into account the non-uniformity of the network load at route selection using such method.

In order to solve this problem, this article suggests that each route in the cache be matched to use counter. It should show how many times this route was used. In case of necessity to select a route to the target, the following factors are used: route length and time it was used. There shall be selected a route with the least values of both parameters. Thereafter, the use counter is incremented. This way, cyclic route retrieval is organized.

Route Change During Transmission: Another optimization suggested in this article is oriented towards improvement in the rerouting time. For this purpose, it is suggested to supplement the base algorithm with a possibility to route changes during transmission. This innovation contradicts the principle of routing from source; however it is oriented towards improvement of the base algorithm in the long run using advantages of AODV algorithm closest to DSR protocol.

The essence of the suggested improvement is that if during data packet transmission the route has any inoperable chain and current node cache has a route to the required target, then it shall replace the inoperable chain of the route with the route taken from the cache. At the same time, the packet routing record replace only transitions following the inoperable chain, while previous transitions are retained. Further advance of the packet is made as in the source routing algorithm. Current node may delete inoperable transition from the cache. All nodes in the new route may also update the route caches. In addition, this possibility may use the route change notification.

Route Change Notification: This change in the algorithm is oriented towards use of the obvious route change notifications in case of using the above described route change during transmission. In this case, when a route changes, the current node should send route change notification the sender. It is a combination of error packet identifying the non-functional transition and the reply packet to the route request containing the new route. This is also a certain deviation from the source routing principles, however it, jointly with the route change during transmission allows minimizing the rerouting time. Using the route change notification produces some additional costs, however they are minimized in case of traffic growth.

Subscriber station shall solve two problems simultaneously: packet transmission and reception. In conformity with this, the formal algorithm for the said problems is indicated below.

Packet Transmission Algorithm

1. i:=0. For each packet in the queue of packets ready to transmission:
   - 1.1 Determine the address of the recipient specified in the packet header;
   - 1.2 Retrieve the route from the cache;
   - 1.3 If there is no route to the preset recipient, go to 1.5;
   - 1.4 Delete the packet in the queue of packets ready to transmission and go to 1.7;
   - 1.5 If the route search has already been initiated, go to 1.7;
   - 1.6 Create the route request packet and transmit it;
   - 1.7 i: = i+1, go to 1.1.
2. i:=0. For each packet in the received packets queue being the data packets:
   - 2.1 Delete the i-th packet from the queue;
   - 2.2 Send the data packet further en route;
   - 2.3 i: = i+1, go to 2.1.
3. i:=0. For each packet in the received packets queue being the error packet:
   - 3.1 Delete the i-th packet from the queue;
   - 3.2 Propagate error packets back using the route and the neighbors;
   - 3.3 i: = i+1, go to 3.1.
4. For each packet in the received packets queue being the reply to the route request:
   - 4.1 Delete the i-th packet from the queue;
   - 4.2 Send the packet back on the route;
   - 4.3 i: = i+1, go to 4.1.
5. For each packet in the received packets queue being the route request:
   - 5.1 Delete the i-th packet from the queue;
   - 5.2 Add new transition to the route in the packet routing record;
   - 5.3 Propagate the route request packet;
   - 5.4 i: = i+1, go to 5.1
6. Go to 1.

Packet Reception Algorithm

1. Receive the packet
2. If the received packet is an error packet, delete from the route cache all routes which contain faulty transitions.
3. If the received packet is an error packet and the current host is not its recipient, add the error packet to the queue of packets ready to transmission.
4. If the received packet is a reply to the route request, add route from the packet routing record in the route cache. If there has not been sent a reply yet then add the packet reply to the queue of packets ready to transmission.
5. If the received packet is a route request, reply to which has not been sent yet, then add the packet request to the queue of packets ready to

transmission (add another transition to the routing record).
6. If the received packet is a data packet and this host is not a recipient then add the packet data to the queue of packets ready to transmission.
7. Go to 1.

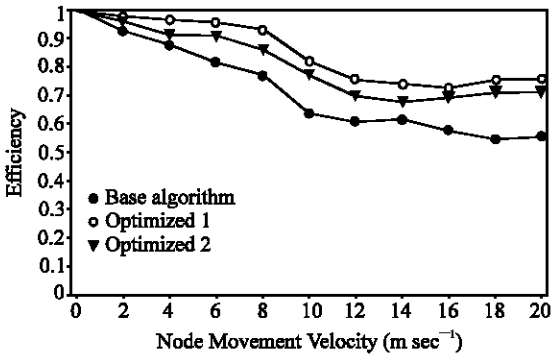

Fig. 1: Efficiency Algorithm as a Function of SsS Movement Velocity

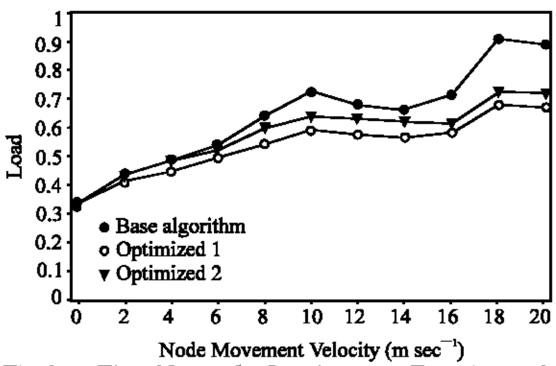

Fig.2: The Network Load as a Function of SsS Movement Velocity

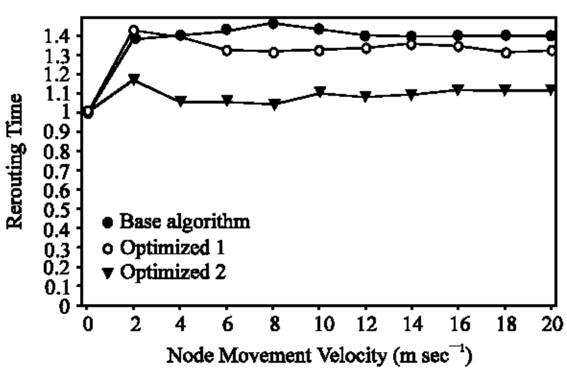

Fig.3: The Re-routing Time as a Function of SsS Movement Velocity

Simulation: We performed a research of efficiency of the selected solution depending upon various environment parameters (SsS mobility, their concentration, traffic intensity).

To estimate the efficiency, we used an efficiency factor equal to the ratio between the number of data packet transmissions and the total number of transmissions (Fig. 1). For estimation of the load, we used a load factor, equal to the ratio between the total number of transmissions and the total network bandwidth (Fig. 2).

Estimation of the rerouting time was made by averaging the number of steps required for receiving of each new route. At the same time, it was considered reception of the route for which no request has been sent (i.e. out of routing records of the packets addressed to other SsS) takes one step (Fig. 3).

CONCLUSION

There was simulated the base algorithm and two optimized implementations of the selected algorithm. It was performed by using specially developed software models intended for investigation of the algorithm efficiency under the preset parameters as a function of the subscriber systems movement velocity, their number and traffic intensity. Based on the results of simulation, the following conclusion may be made. Using extended caching and extended use of cache gives efficiency boost by 5-10%. Use of the suggested improvements allows reducing the rerouting time by 20-30%. The obtained difference in productivity reveals the expediency of using the suggested modifications.

Dynamic source routing algorithm shows good results at operation with both low number of the subscriber systems and with high their number (up to 200). The routes used are optimal with high probability. This algorithm has a lot of advantages. It promptly adapts to route change at fast SsS movement and at low movement velocity works with minimum costs or without them, disables infinite looping of routes without additional overhead, allows using unidirectional communication. In this study, the base algorithm was supplemented with an opportunity of load balancing, changes in the route while sending, and the route change notification. Using the algorithm modifications described in this work boosts its efficiency in comparison with other routing algorithms.

REFERENCES

1. Perkins, C.E. and P. Bhagwat, 1994. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. ACM SIGCOMM Symp.
2. Royer, E.M. and C.K. Toh, 1999. A review of current routing protocolsfor ad hoc mobile wireless networks. IEEE Personal Communications.
3. Perkins, C.E. and E.M. Royer, 2003. Ad hoc on demand distance vector routing (AODV). MANET Working Group, IETF..
4. Johnson, D.B. and D.A. Maltz, 1997. Dynamic source routing in ad hoc wireless networks. Carnegie Mellon University, Pittsburgh.
5. Johnson, D.B., D.A. Maltz and Yih-Chun Hu, 2003. The Dynamic Source Routing Protocol (DSR) for mobile ad hoc wireless networks. MANET Working Group, IETF.
6. Park, V.D. and M.S. Corson, 1997. Temporally-Ordered Routing Algorithm (TORA)-Internet draft.
7. Jiang, M., J. Li and Y.C. Tay, 1999. Cluster based routing protocol. MANET Working Group, IETF.
8. Toh, C.K., 1999. Long-lived ad-hoc routing based on the concept of associativity. MANET Working Group, IETF.
9. Thampuran, S.R., 1999. Routing protocols for ad hoc mobile wireless networks. University of Massachusetts, Amherst.
10. Tannenbaum, A., 2002. Computer Networks. SPb.: Piter, pp: 848.