Shahram JAMALI, Sajjad JAHANBAKHSH[2]

University of Mohaghegh Ardabili (1), Islamic Azad University of Germi (2)

BA-TORA: a Multipath Routing Protocol for MANETs by Inspiration from Bee and Ant Colonies

Abstract. *A mobile ad hoc network (MANET) consists of a set of mobile nodes which communicate over radio. One of the major challenging issues in this kind of networks is to develop an efficient routing protocol. Since MANET nodes have restricted battery capacity, an energy-aware routing protocol can play an important role in its performance. On the other hand, in recent years nature has been emerged as source of inspiration to successfully solve of many scientific problems. Hence in this paper we present BA-TORA, a TORA-based reactive multipath energy-aware routing protocol that has been inspired from bee and ant colonies. Simulation results show that BA-TORA outperforms TORA in terms of number of delivered packets, life time of network and life time of system.*

Streszczenie. *Sieci mobilne i ad-hoc MANET składają się z wielu ruchomych węzłów połączonych droga radiową. Dlatego bardzo ważna rolę odgrywa zastosowanie skutecznego protokołu transmisji. W sieciach MANET mogą być ograniczone pojemności baterii zasilających. W artykule zaprezentowano nowy protokół bazujący na protokole TORA inspirowany koloniami mrówek lub pszczół. Symulacje potwierdziły skuteczność protokołu opisywanego liczba przesłanych pakietów i czasem przesyłania.* (***BA-TORA protokół przesyłania danych w sieciach MANET inspirowany algorytmami mrówkowymi***).

Keywords: MANET, TORA, Routing, Bee Colony, Ant Colony
Słowa kluczowe: MANET, TORA, sieci bezprzewodowe.

Introduction

MANET is a network consisting of a collection of mobile nodes with no centralized management [1]. In the absence of a fixed infrastructure, MANET nodes participate to provide routing services, relying on each other to forward packets to their destination [2]. MANET has characteristic such as dynamic topology, energy-constrained nodes and limited computing capability. Such network is highly suitable for applications involving emergencies, communications, natural disasters and military operations [3]. In this situation, it is difficult for nodes to recharge. So it is very significant to trade off the energy consumption of nodes, in order to prolong the lifetime of the network. All nodes in MANETs have two roles: producer/consumer of data packets streams, and routers for data packets destined for the other nodes [4, 5]. The most important issues in these networks are: mobility and limited battery capacity of the nodes. Mobility of nodes results in continuously evolving new topologies and dynamic topology and the routing algorithms have to adapt the routes according to this dynamic topology [6]. The limited battery capacity poses yet another challenge for the routing algorithms: to distribute the packets on multiple paths in such a manner that the battery of different nodes deplete at same rate, in this way, the lifetime of the network could be increased [3, 7].

Routing protocols for MANETs broadly classified in two classes: proactive and reactive. In proactive routing protocols like [8], mobile nodes update their routing tables by periodically exchanging routing information among themselves. In reactive routing protocols like [9, 10], a route is discovered when it is required.

Multipath routing protocols for MANETs can be broadly divided into five categories [1]: delay aware multipath routing protocols, reliable multipath routing protocols, minimum overhead multipath routing protocols, energy aware multipath routing protocols and hybrid multipath routing protocols. Multipath routing protocols like [11, 12 and 13] that improve energy efficiency of a network called energy aware routing protocols. The main goal of these protocols is to maximize the network life by efficiently utilizing the battery of a mobile node. A multipath routing protocol can save the batteries of the mobile nodes by distributing network traffic uniformly between them [14, 15]. TORA [2] is a reactive multipath routing protocols for MANETs that discovers all paths between source and destination, then uses one of the these paths that is shortest. Compared to other reactive routing protocols, TORA is suitable for high mobility environments. In this paper we inspire from bee and ant colonies to design BA-TORA, a new reactive multipath energy aware MANET routing algorithm that is based on TORA algorithm. The reminder of paper has been organized as follows. Section 2 describes behaviour of a bee colony and an ant colony. In section 3 we review the TORA routing protocol and in section 4 we will describe our routing algorithm. Section 5 explains the experimental framework and discusses the results obtained from the simulation. Finally a conclusion is given in section 6.

Bees and ants in nature

Colonies of societal insects such as bees and ants have inherent ability called swarm intelligence. This highly organized behaviour enables the colonies of insects to resolve problems further than the capability of individual members by functioning collectively and interacting primitively between members of the group. In a honey bee colony for example, this behaviour allows bees to explore the environment for flower patches (food sources) and then inform to the other bees of the colony about food source when they return to the hive. Such a colony is characterized by self-organization and robustness [16].

Bee Colony: A colony of bee can extend itself over wide distances in order to search for food sources at the same time. The foraging process begins in a colony by scout bees that being sent to search for hopeful flower patches. Scout bees move randomly from one path to another. Flower patches with greater amounts of nectar or pollen that can be collected with less challenge tend to be visited by more bees, whereas patches with less nectar or pollen receive fewer bees. Through the harvesting season, a colony continues its search, keeping a percentage of the population as scout bees. When scout bees return to the hive go to the dance floor to do a dance known as the waggle dance [10].

This strange dance is necessary for communication of colony members, and contains three pieces of information about a flower patch [11]: (a) the direction in which flower will be found (b) distance from the hive (c) its quality. This information helps the bee colony to send its bees to flower patches specifically, without using guides or maps.

Each individual's information of the outside environment is gleaned solely from the waggle dance. This dance enables the colony to assess the worth of different patches according to both the quality of the food they provide and the amount of energy needed to harvest it.

Ant Colony: Ants are community insects. They live in colonies and their goal is colony survival rather than individuals' survival. Ants when exploring for food source, they initially search the area adjacent their nest in a random manner. While moving, ants put a chemical pheromone trail on the earth. Ants can smell pheromone. When selecting their way, ants tend to choose, paths with more pheromone. When an ant finds a food source, it calculates the quantity and the quality of the food and brings some of it back to the nest. During the come back to nest, the amount of pheromone that an ant puts on the earth may depend on the quantity and the quality of the food. The pheromone trails will help other ants to the food source [18]. In this way, the indirect communication between the ants via pheromone enables ants to find shortest paths between their nest and food sources [19].

TORA Routing Protocol

TORA [2] is a highly adaptive distributed routing algorithm that is well-suited for use in mobile ad hoc networks. In TORA, each node has five types of information; H_i, N_i, $HN_{i,j}$, RR_i and $LS_{i,j}$:

$H_i = (\tau_i, oid_i, r_i, \delta_i, i)$ is related with each node $i \epsilon N$ (N is the set of nodes in network). In this ordered quintuple τ_i, is logical time of a link failure, oid_i is the ID of originator node, r_i is a bit used to divide each of the unique reference into two unique sub-levels, δ_i is a propagation ordering parameter and i is the unique ID of the node. Initially, the height of each node in the network other than the destination node is set to NULL, H_i =(-, -, -, -, i). The height of the destination node is always ZERO, $H_{did} = (0,0,0,0, did)$, that did is the ID of destination node. N_i is the set neighbors of node i. $HN_{i,j}$ is a height array for each neighbor $j\epsilon N_i$. Initially the height of each neighbor node is set to NULL, $HN_{i,j} = (-,-,-,-,j)$. If the destination is a neighbor of node i, node i sets the height entry of the destination node to ZERO, $HN_{i,did} = (0,\ 0,\ 0,\ 0, did)$. RR_i is a route-required flag (RR_i) which is initially un-set. $LS_{i,j}$ is a link-state array with an entry for each link $(i,j)\epsilon L$, where $j\epsilon N_i$. The state of the links is determined by the heights H_i and $HN_{i,j}$. Each link is going from the higher node to the lower node. If a neighbor j is higher than node i, the link called upstream (UP). If a neighbor j is lower than node i, the link called downstream (DN). If the neighbors height entry, $HN_{i,j}$, is NULL, the link is distinct undirected (UN).

TORA can be divided into three phase: creating routes, maintaining routes and erasing routes. Details of these phases are exactly brought from [2] as follow.

Creating Routes: This phase requires use of the QRY packets and UPD packets. A QRY packet consists of an ID of destination ($\boldsymbol{did}$), which distinct the destination node that algorithm is running. An UPD packet consists of a $\boldsymbol{did}$, and the height of the node $\boldsymbol{i}$ which is broadcasting the packet, $\boldsymbol{H_i}$. When a node that has undirected links and its route required flag is 0, need a route to the destination node, it broadcasts a QRY packet and set its route-required flag with 1. When a node $\boldsymbol{i}$ receives a QRY packet, it operates as follows: (a) if it has no downstream links and its route required flag is 0, it re-broadcasts the QRY packet and sets its route-required flag with 1. (b) If it has no downstream links and its route-required flag is set with 1, it rejects the QRY packet. (c) If it has at least one downstream link and its height is NULL, this node sets its height to $\boldsymbol{H_i = (\tau_j, oid_j, r_j, \delta_j + 1, i)}$assuming $\boldsymbol{HN_{i,j} = (\tau_j, oid_j, r_j, \delta_j, j)}$ is the minimum height of its neighbours that are non-NULL, then broadcasts an UPD packet. (d) If it has at least one downstream link and its height is non-NULL, it first compares the time the last UPD packet was broadcast to the time the link over which the QRY packet was received became active. If an UPD packet has been broadcast since the link became active, it rejects the QRY packet; otherwise, it broadcasts an UPD packet. If a node has the route-required flag with 1 when a new link is established, it broadcasts a QRY packet.

When a node i receives an UPD packet from a neighbor $j \in N_i$, node i first updates the entry $HN_{i,j}$ with the height contained in the received UPD packet and then operates as follows. (a) If the route required flag is set with 1, node i sets its height to $H_i = (\tau_j, oid_j, r_j, \delta_j + 1, i)$, that j is its non-NULL neighbour and height of it is minimum in between neighbour nodes , then updates all the entries in its link-state array LS, un-sets the route-required flag and then broadcasts an UPD packet which contains its new height. (b) If the route-required flag is 0, node i simply updates the entry $LS_{i,j}$ in its link-state array.

Maintaining Routes: Routes Maintenance is only performed for nodes that their height is non-NULL. Furthermore, any neighbour's height which is NULL is not used for the computations. A node i is said to have no downstream links if H_i is lower than $HN_{i,j}$ for all non-NULL neighbors$j\epsilon N_i$. This will result in one of five possible actions depending on the state of the node and the preceding event. Each node (other than the destination) that has no downstream links modifies its height, $H_i = (\tau_i, oid_i, r_i, \delta_i, i)$, as follows.

Case 1: Node i has no downstream links due to a link failure. In this case: $(\tau_i, oid_i, r_i) = (t, i, 0)$ and $(\delta_i, i) = (0, i)$, where t is the time of the failure

Case 2: Node i has no downstream links due to a link reversal following reception of an UPD packet and the ordered sets (τ_j, oid_j, r_j) are not equal for all $j\epsilon N_i$. In this case: $(\tau_i, oid_i, r_i) = max\{(\tau_j, oid_j, r_j) | j\epsilon N_i\}$

$$(\delta_i, i) = \left(min\left\{ \delta_j \middle| \begin{array}{l} j \in N_{i\, with\, (\tau_j, oid_j, r_j)} \\ = \max\{(\tau_j, oid_j, r_j)\} \end{array} \right\} - 1, i \right)$$

Case 3: Node i has no downstream links due to a link reversal following reception of an UPD packet and the ordered sets $(\tau_j, oid_j, r)_j$ are equal with $r_j = 0$ for all $j\epsilon N_i$. In this case, $(\tau_i, oid_i, r_i) = (\tau_j, oid_j, 1)$ and $(\delta_i, i) = (0, i)$.

Case 4: Node i has no downstream links due to a link reversal following reception of an UPD packet, the ordered sets (τ_j, oid_j, r_j) are equal with $r_j = 1$ for all $j\epsilon N_i$ and $oid_j = i$. In this case $(\tau_i, oid_i, \Box r_i) = (-,-,-), (\delta_i, i) = (-, i)$

Case 5: Node i has no downstream links due to a link reversal following reception of an UPD packet, the ordered $(\tau_j, oid_j, r)_j$ are equal with $r_j = 1$ for all $j\epsilon N_i$ and $oid_j \neq i$. In this case: $(\tau_i, oid_i, \Box r_i) = (t, i, 0)$ and $(\delta_i, i) = (0, i)$.

Erasing Routes: If case 4 of maintaining routes phase are detected, node $\boldsymbol{i}$ sets its height and the height entry for each neighbor $\boldsymbol{j\epsilon N_i}$ to NULL unless the destination is a neighbor, in which case the corresponding height entry is set to ZERO, updates all the entries in its link-state array $\boldsymbol{LS}$, and broadcast a CLR packet. The CLR packet containing of a $\boldsymbol{did}$ and the reflected level of node $\boldsymbol{i}$, ($\boldsymbol{\tau_i}, \boldsymbol{oid_i}$□□□. When a node $\boldsymbol{i}$ receives a CLR packet from a neighbor $\boldsymbol{j \epsilon N_i}$ it perform as below. (a) If the reference level in the CLR packet matches the reference level of node $\boldsymbol{i}$, it sets its height and the height entry for each neighbor $\boldsymbol{j \epsilon N_i}$ to NULL unless the destination is a neighbor, in which case the corresponding height entry is set to ZERO, updates all the entries in its link-state array $\boldsymbol{LS}$ and broadcasts a CLR packet. (b) If the reference level in the CLR packet does not match the reference level of node $\boldsymbol{i}$, it sets the height entry for each neighbor $\boldsymbol{j \epsilon N_i}$ to NULL and updates the matching link-state array entries. Thus the height of each node in the part of the network which was partitioned is set to NULL and all invalid routes are erased. If (b) causes node $\boldsymbol{i}$ to lose its last downstream link, it perform as in case 1 of maintaining routes.

Fig. 1 shows an example in which node S wants to send data to node D. In this example, all routes from source to the destination are discovered by using TORA routing protocol.

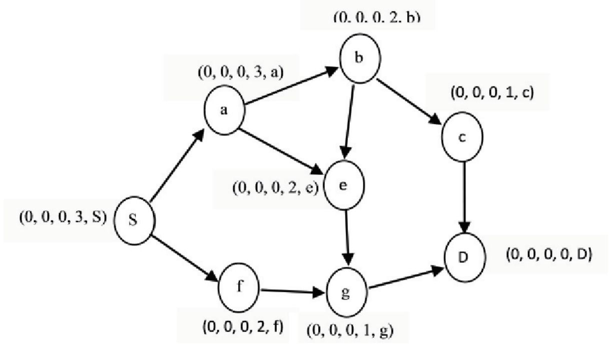

Fig. 1 Routes creating process in TORA

Proposed Routing Protocol: BA-TORA

Although TORA can typically provide multiple routes for any source/destination pair, it always chooses route with fewer hops when the network topology doesn't change. We believe this can put some routes under heavy load and hence energy of nodes in these routes deplete earlier than other nodes. This unbalanced loads lead to decreased network life time [3] and as a result the throughput of network decreases. To solve this problem we borrow some principles of bee and ant colonies to develop a novel protocol that distributes the traffic uniformly between nodes and selects the routes with the high level of energy. This causes the energy of nodes to deplete with same rates and leads to increased lifetime of the network.

Mapping from Nature to the MANET: Consider a scenario in which ants and bees help each other to find food sources, and update quality of paths to these food sources, continually. Suppose that in this process bees are responsible to discover new food sources and ants update information about quality of those food sources during the time. We can describe this collaboration as follows:

1. Scout bees travel randomly from one path to another to search for flower patch.
2. When scout bees return to the hive, it perform dance to announcement three pieces of information about flower patch: direction, distance from hive and quality of food source.
3. By using these information colony sends its bees to the food source.
4. When bees arrive at the food source, some ants are sent from the source to hive that update information about quality of paths by using pheromone.
5. Some scout bees continues its search and announces new information of nature. In other words, if new paths exist, these paths are announced by scout bees.

Steps 3, 4 and 5 are repeated while there is any food source.

We believe that a routing process in ad-hoc networks are so similar to mentioned scenario. In a typical routing protocol we need to find the best route to destination (the shortest one, the most stable one, the route with highest energy level, etc.) and monitor its quality during the time. This is so similar to mentioned scenario in which bees find paths to food sources and ants monitor paths' quality, continually. This similarity motivates us to design a new routing protocol, namely BA-TORA, which is inspired from bee and ant colonies and acts as follows:

6. QRY packets are broadcasted from source to various paths.
7. When QRY packet arrives at destination, UPD packets are sent from destination to the source. These UPD packets contain three pieces of information about route: direction of routes, hop counts of routes from the source to the destination and level of energy in routes.
8. By using this information, data packets are sent over routes with maximum level of energy and minimum hop count.
9. Every time data packets arrive at destination, energy levels of active routes are updated by sending some special packets to the source.
10. When new routes are created, it is announced to the source by using UPD packet.

Steps 3, 4 and 5 are repeated while there is any data to be sent from the source node to the destination node.

In this point we are going to draw BA-TORA details. As we saw, H_i in TORA includes $(\tau_i, oid_i, r_i, \delta_i, i)$, since BA-TORA uses energy level of nodes as another metric to measure quality of routes, hence we add a new field to H_i. In BA-TORA, H_i will be defined as $H_i = (\tau_i, oid_i, r_i, \delta 0_i, \delta 1_i, i)$, in which $\delta 0_i$ is considered same as δ_i in TORA and $\delta 1_i$ is sum of energy of nodes from destination to node i.

Initially, the height of each node excluding destination is set to NULL, $H_i = (-,-,-,-,-,i)$. H_i is modified according to the rules of BA-TORA during the time. Also, like TORA the height of the destination node is always ZERO, H_{did}=(□, 0, 0, 0, 0,did). Note that BA-TORA is build based on TORA but divided in four phases: creating routes, maintaining routes, erasing routes and updating energy level of routes. Here, we only focus on the differences between BA-TORA and TORA:

Creating Routes: When a node i receives a QRY packet and the receiving node has at least one downstream link and its height is NULL, it sets its height to $H_i = (\tau_j, oid_j, r_j, \delta 0_j + 1, \delta 1_j + energy\ of\ this\ node, i)$, where $HN_{i,j} = (\tau_j, oid_j, r_j, \delta 0_j, \delta 1_j, j)$ is the minimum height of its neighbours that are non-NULL, and broadcasts an UPD packet.

Also, when a node i receives an UPD packet from a neighbor $j \in N_i$ and the route required flag is set to 1 (which implies that the height of node i is NULL), node i sets its height to

$H_i = (\tau_j, oid_j, r_j, \delta 0_j + 1, \delta 1_j + energy\ of\ this\ node\ , i)$
where: $HN_{i,j} = (\tau_j, oid_j, r_j, \delta 0_j, \delta 1_j, j)$ is the minimum height of its non-NULL neighbors, updates all the entries in its link-state array LS, un-sets the route-required flag and then broadcasts an UPD packet which contains its new height.

Maintaining Routes: Each node i (other than the destination) that has no downstream links modifies its height, $H_i = (\tau_i, oid_i, r_i, \delta 0_i, \delta 1_i, i)$ and has no downstream links due to a link reversal following reception of an UPD packet and the ordered sets (τ_j, oid_j, r_j) are not equal for all $j \epsilon N_i$. In this case:

$$(\tau_i, oid_i, r_i) = max\{(\tau_j, oid_j, r_j) | j \epsilon N_i\}$$
$$(\delta 0_i, \delta 1_i, i) = \left(min\left\{ \delta 0_j \middle| \begin{array}{l} j \in N_{i\ with\ (\tau_j, oid_j, r_j)} \\ = \max\{(\tau_j, oid_j, r_j)\} \end{array} \right\} - 1,\ \delta 1_i - energy\ of\ this\ node, i \right)$$

Erasing Routes: This phase is performed exactly like TORA. In this phase $\boldsymbol{\delta 1_i}$ is set same as $\boldsymbol{\delta 0_i}$ of TORA.

Updating Energy Level of Routes: In this phase, BA-TORA updates energy levels of all active routes. Energy level of any route is defined as sum of energies exist in various nodes of the route. For this purpose, when destination node receives a data packet sends its own energy level to the neighbour node. Subsequently, each node in the route from destination to the source uses equation (1) to compute energy level of that portion of route that starts from destination and continues to this node. It is clear that when equation (1) is running in the source node, it computes total amount of energy on the route.

$$\delta 1_i = \delta 1_j + energy\ of\ this\ node \quad (1)$$

Where: node j is neighbour of node i that is earlier than node i in the route from destination to the source.

Fig. 2 shows an example in which node S wants to send data to node D. All routes from source to the destination are discovered by using BA-TORA routing protocol. In this example we have assumed that energy of any node is 40J.

In this point that BA-TORA has discovered direction, length and quality of all possible routes from source to the destination, source selects a route that has maximum energy level (maximum of $\delta 1_i$) and shortest length (minimum of $\delta 0_i$).

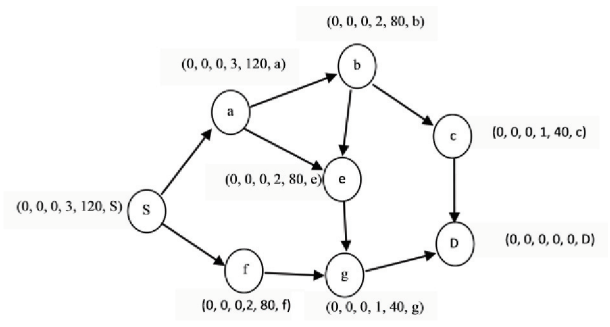

Fig. 2 Routes creating process in BA-TORA

Simulation results

To investigate the impact of our enhancements, we implement BA-TORA by making some modification over TORA module of ns-2 [20] simulator. In this simulation 25 nodes were simulated for 500s over a network space of 1000×1000m. The network traffic was modeled as 10 CBR sources with data sent in 512 byte packets with rate 4 packet/sec. Five movement patterns were produced based on a random waypoint model [21] with pause time: 0, 10, 20, 30, 40 seconds. The primary energy of all nodes is 40J. The IEEE 802.11 is used as the medium access protocol. The interface queue is 50-packet drop-tail priority queue.

Three performance metrics [22] that are used to compare TORA and BA-TORA are as follow:

- The network lifetime: it is defined as the time when a node finished its own battery for the first time [23].
- The system lifetime: it is defined as the time when 20% of nodes finish their own battery.
- The total data delivered: it is defined as the total number of data packets that delivered during of system lifetime.

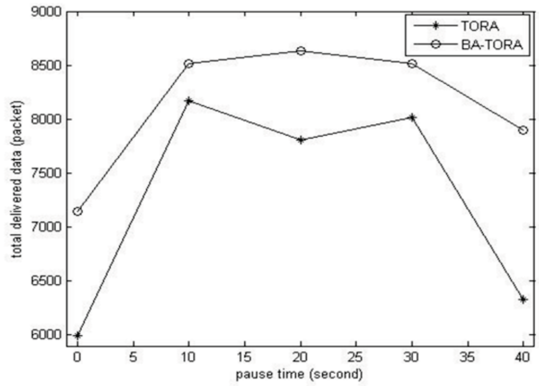

Fig. 3 Total delivered data

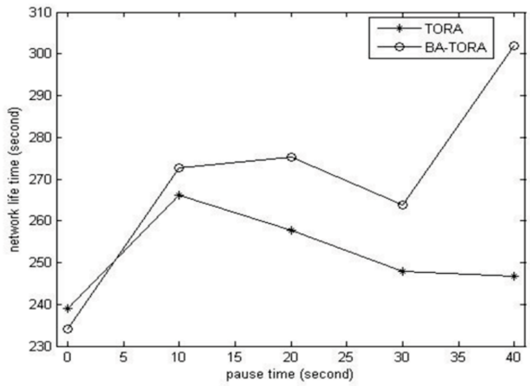

Fig. 4 Network life time

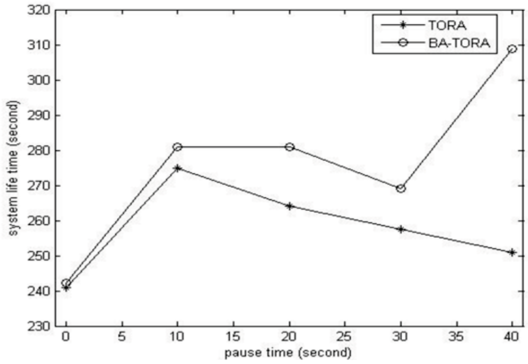

Fig. 5 System life time

We simulate this network once under TORA protocol and then under BA-TORA protocol to compare their performance. Simulation results are shown in Figs. 3-5. Fig. 3 shows that BA-TORA increases delivered data in compare with TORA protocol. On the other hand, Fig. 4 and Fig. 5 show that BA-TORA outperforms TORA in terms of network life time and system life time. Better performance of BA-TORA is due to this fact that it uses those routes that have higher level of energy. This leads to balanced consumption of energy in various routes and nodes; hence, BA-TORA experiences less route breakages and achieves better performance.

Conclusion

In this paper we proposed BA-TORA as an improvement over TORA routing protocol. It inspires from bee and ant colonies and proposes a procedure, in which, bees discovers routes and ants monitors routes' quality. The main feature of BA-TORA is that it distributes load among routes considering energy of routes. This causes the load to be distributed uniformly among routes and leads to long life of network. The simulating results indicated that the total delivered data, network life time and system life time in BA-TORA are better than TORA routing protocol.

REFERENCES

[1] M. Tarique, K.E. Tepe, S. Adibi and S. Erfani, "Survey of multipath routing protocols for mobile ad hoc networks", *Network and Computer Applications*, vol. 32, 2009.

[2] V.D. Park and M.S. Corson, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks", *INFOCOM*, vol. 97, 1997.

[3] F.Yu, Y.Li, F.Fang and Q.Chen, " A New TORA-based Energy Aware Routing Protocol in Mobile Ad Hoc Networks", 2007.

[4] R.Menchaca-Mendez and J.J. Garcia-Luna-Aceves, "Hydra: Efficient multicast routing in MANETs using sender-initiated multicast meshes", *Pervasive and Mobile Computing*, vol. 6, 2010.

[5] J.A. Torkestani and M.R. Meybodi, "Mobility-based multicast routing algorithm for wireless mobile Ad-hoc networks: A learning automata approach", *Computer Communications*, vol. 33, 2010.

[6] W. Kiess and M. Mauve, "A survey on real-world implementations of mobile ad-hoc networks", *Ad Hoc Networks*, vol. 5, 2007.

[7] O. Souihli, M. Frikha and M.B. Hamouda, "Load-balancing in MANET shortest-path routing protocols", *Ad Hoc Networks*, vol. 7, 2009.

[8] C.E. Pins and P. Bhagwat , "highly dynamic destination sequence distance vector (DSDV) routing for mobile computers", *ACM SIGCOMM*, 1994.

[9] J. Broch, D.B. Johnson and D.A. Maltz, "The dynamic source routing for mobile ad hoc networks", 1998.

[10] C. Perkins and E. Royer,"Ad-hoc on-demand distance vector", *IEEE Workshop on Mobile Computing Systems and Applications*, 1999.

[11] S. Dulman, J.Wu and P. Having, "An energy efficient multipath routing algorithm for wireless sensor networks", *International symposium on autonomous decentralized systems with an emphasis on advanced distributed transportation system*, 2003.

[12] O. Liang and O. Ren, "Energy and mobility aware geographical multipath routing for wireless sensor networks", *IEEE WCN*, 2005.

[13] M. Tarique and K.E. Tepe, "Minimum energy hierarchical dynamic source routing for Mobile Ad Hoc Networks", *Ad Hoc Networks*, vol. 7, 2009.

[14] H.F.Wedde, M. Farooq, T. Pannenbaecker, B. Vogel, C. Mueller, J. Meth and R. Jeruschkat, "BeeAdHoc: An Energy Efficient Routing Algorithm for Mobile Ad Hoc Networks Inspired by Bee Behavior", *ACM*, 2005.

[15] R. Ramanathan and J. redi , "A Brief Overview of ad Hoc Networks: Challenges and Directions", *IEEE Communications Magazine* , 2002.

[16] Z. Z. Abidin, M.R. Arshad and U.K. Ngah, "A Survey: Animal-Inspired Metaheuristic Algorithms", *EEPC 2009*, 2009.

[17] K. von Frisch, "The Dance Language and Orientation of Bees", *Harvard University Press*, 1967.

[18] Ch. Blum, "Ant colony optimization: Introduction and recent trends", *Physics of life Review*, vol. 2, 2005.

[19] M. Dorigo, Ch. Blum, "Ant colony optimization theory: A survey", *Theoretical Computer Science*, vol. 344, 2005.

[20] NS-2, "The Network Simulator", http://www.isi.edu/nsnam/ns/.

[21] D.B Johnson and D.A Maltz, "Dynamic source routing in ad hoc wireless networks", Mobile Computing, 1996.

[22] W. Minqiang and Z. Bsoyu, "A New DSR-Based Routing Protocol with Energy-Aware in Ad Hoc Networks", *Journal of Nanjing University of Posts and Telecommunications*, vol.25(1), 2005.

[23] Q. Li, J. Aslam and D. Rus, "Online Power-aware Routing in Wireless Ad-Hoc Networks", *MOBICOM*, 2001.

Authors

Shahram Jamali, University of Mohaghegh Ardabili, Ardabil, Iran, E-mail: jamali@iust.ac.ir; Sajjad Jahanbakhsh, Islamic Azad University of Germi, Germi, Iran, E-mail: sa_jahanbakhsh@yahoo.com.