Journal of Mobile Communication 2 (4): 105-111, 2008
ISSN: 1990-794X

MARP: Multi Ant Based Routing Protocol for Mobile AdHoc Networks

[1]D. Siva Kumar and [2]R.S. Bhuvaneswaran
[1]Department of Electrical and Electronics,
[2]Ramanujan Computing Center, Anna University, Chennai 600025, India

Abstract: The field of Mobile AdHoc Networks (MANETs) has gained an important part of the interest of researchers and has become very popular in last few years. MANETs can operate without fixed infrastructure and can survive rapid changes in the network topology. In MANETs, there are frequently disconnected routes consisting of multi-hop from a source to a destination because of the dynamic nature such as the topology change caused by node's mobility. To overcome this situation, this study proposes a Multi Ant based Routing Protocol (MARP) for mobile AdHoc networks, which combines the on-demand routing capability of AdHoc On-Demand Distance Vector (AODV) routing protocol with a ant colony optimization mechanism using ant like mobile agents. AODV requires the actual communication to be delayed until the route is determined. This may not be suitable for real time data and multimedia communication applications. The Multi Ant based Routing Protocol (MARP) provides high connectivity, reducing the amount of route discoveries before starting new connections. This eliminates the delay before starting actual communication for most new connections making a multi ant based routing protocol ideal for real time communication in highly dynamic networks such as MANETs. We study the performance of this ant protocol for dynamic network topologies. We also compare, the performance of this multi ant based routing protocol with well known existing hybrid routing protocols ZRP and ARA, for AdHoc networks. Simulation results show that the new multi ant routing protocol proposed in this study is able to achieve reduced end-to-end delay as compared to existing hybrid routing protocol ZRP and ARA. In addition, this new multi ant routing protocol also provides significantly better packet delivery ratio, route discovery frequency and multimedia communication than ARA and ZRP for high and low mobility.

Key words: AdHoc networks, multi ant based routing protocol, Adhoc on-demand Distance Vector (AODV), mobile Adhoc networks, Zone Routing Protocol (ZRP) and ant routing algorithm

INTRODUCTION

A Mobile AdHoc Network (MANET) is a decentralized network of autonomous mobile nodes able to communicate with each other over wireless links. Due to the mobility of the nodes, the topology of the network may rapidly be changing, making it impossible to use conventional routing tables maintained at fixed points (routers). Instead, each node is required to determine the best route to a given destination node by itself. Given their dynamic nature, route discovery in a MANET differs significantly from the more or less static routes in wired networks: Not all nodes in a MANET necessarily have the same capabilities. Two nodes, even if they are direct neighbors, may differ with respect to signal strength, available power, reliability etc. These differences require much more complicated and particularly more active distributed algorithms in order to maintain an accurate picture of the networks topology, while at the same time providing scalability for potentially large (and ever-growing) networks. At the same time, route discovery must not use up the majority of the often-limited bandwidth available to today mobile devices. Furthermore, it is important to point out an important difference to conventional routing approaches: In wired networks, each link is bi-directional. If a node A can send packets to a node B, we know that node B can send packets back to node A and a reverse path can be entered. This is not necessarily, the case in a wireless network, where the physical location and the individual power resources have great influence upon a nodes transmission capacity and signal strength. MANET routing protocols are IP based and may use unicast, multicast or hybrid approaches and should allow for interaction with standard wired IP services rather than being regarded as a completely separate entity. A detailed yet not overly complex overview of the various aspects of Mobile AdHoc Networking is given in Corson and Macker (1999).

Corresponding Author: D. Siva Kumar, Department of Electrical and Electronics, Anna University, Chennai 600 025, India

Ant Colony Optimization (ACO), called ant system was inspired by studies of the behavior of ants. As a multi agent approach to different combinatorial optimization problems, like the traveling salesman problem and the quadratic assignment problem. The ant-colony metaheuristic framework was enabled ACO to be applied to a range of combinational optimization problems. Ant colony algorithms have been founded on an observation of real ant colonies. By living in colonies, ants' social behavior is directed more to the survival of the colony as an entity rather than to that of an individual member of the colony. An interesting and significantly important behavior of ant colonies is the aging behavior and in particular, their ability to find the shortest route between their nest and a food source, realizing that they are almost blind. Ant colony optimization algorithms have been used to produce near-optimal solutions to the traveling salesman problem. They have an advantage over simulated annealing and genetic algorithm approaches when the graph may change dynamically; the ant colony algorithm can be run continuously and adapt to changes in real time. This is of interest in network routing and urban transportation systems (Perkins and Royer, 1999). It has been experimentally observed that ants in a colony can converge on moving over the shortest among different paths connecting their nest to a source of food. The main catalyst of this colony-level shortest path behavior is the use of a volatile chemical substance called pheromone that is the ants moving between the nest and a food source deposit pheromone and preferentially move in the direction of areas of higher pheromone intensity. Shorter paths can be completed quicker and more frequently by the ants and will therefore, be marked with higher pheromone intensity. These paths will therefore, attract more ants, which will in turn increase the pheromone level, until there is convergence of the majority of the ants onto the shortest path. The local intensity of the pheromone field, which is the overall result of the repeated and concurrent path sampling experiences of the ants, encodes a spatially distributed measure of goodness associated with each possible move.

MATERIALS AND METHODS

Related works: During recent years, new algorithms have been introduced to tackle the combinatorial optimization problems. Some examples of these algorithms are Genetic Algorithms (Hertz *et al*., 1991), Ant Systems Dorigo *et al*. (1996) and Neural Networks (GoldBerg, 1991). Those algorithms are inspired by behavior or processes present in nature. Such mechanisms have been proven to be extremely effective and evolutionary. These algorithms demonstrate adaptive, robust and effective behavior as nature does, where adaptive means that it improves its goal-achieving competence over time, robust means it is flexible and never completely breaks down while effective means it is eventually finding a satisfactory solution. The observation and study of ants societies have long since attracted the attention of the professional entomologist, but in recent years, the ant model of organization and interaction has also captured the interest of the computer scientist and engineers. Ant societies, among other societies, have many features like autonomy of individuals, fully distributed control, collective and cooperative strategies, emergence of complex behaviors with respect to the single ant and self-organization. The simultaneous presence of these unique characteristics has made ant societies an attractive and inspiring model for building new algorithms and new multi-agent systems. Since 1990, ant societies have provided the force for a growing field of scientific research, mostly in the field of robotics, operations research and telecommunications.

Researchers from all over the world, processing different scientific backgrounds, have made significant progress concerning both implementation and theoretical aspects, within this novel research framework. Their contributions have given the field a solid basis and have shown how the ant way, when carefully engineered, can result in successful applications to many real-world problems.

A particular successful research in ant algorithms, known as Ant Colony Optimization (ACO), is dedicated to their application to discrete optimization problems. Ant colony optimization has been applied successfully to a large number of difficult combinatorial problems such as the traveling salesman problem, the quadratic assignment problem and scheduling problems, as well as routing in telecommunication networks. The routing algorithm based on ants was developed by Perkins and Royer (1999) and Dorigo *et al*. (1996) and further discussed in Gunes *et al*. (2002) and Marwaha *et al*. (2002), However, none of these studies investigated, the problem of mobile networks, where nodes change their position over time. In such a mobile network, some nodes may be connected during route discovery, but are disconnected when the data should be transferred. If this happens a mechanism called Route Maintenance will start to find the node the messages should be sent to. That is how we guaranteed that the route is stable and doesn't break down. Simulation results show that the total number of messages to find the target can be reduced compared to a Zonal routing protocol as well as ARA.

Multi Ant based Routing Protocol: MARP: In this study, we present the basic concepts of proposed algorithm. We started with the following reactive approach (on demand). If a node wants to establish a connection to another node, it sends out a Forward Ant (FANT) that walks randomly in the network. In our algorithm randomly means that on each node the FANT chooses one of its neighbors with equal probability. While, hopping from one node to the next the FA leaves a track by putting routing table entries in the nodes which point towards the source. It walks around randomly in the network, until it reaches the target it was sent out for. Once, it reaches the target, a Backward Ant (BANT) is generated, which follows the routing table entries back to the source. On its way the BANT itself leaves entries in the routing table again for its source (the target of the connection).

Random walk: Before elaborating the algorithm in detail we focused on what it actually means to walk randomly in a network. The first and most important thing is certainly that the random walk on networks can be modeled as a Markov chain. This chain is ergodic and has a steady state. In Peter and Snell (2000) the authors show a way to compute the expected number of steps before absorption (hitting time). If one builds up the transition matrix in the following way:

$$P_{ij} = \begin{cases} 0 & \text{no connection between node i and j} \\ \dfrac{1}{\deg(i)} & \text{if i and j are neighbours} \end{cases}$$

Where, deg(i) is the number of adjacent edges. For the absorbing state (the target of the connection) the rule is different:

$$P_{ij} = \begin{cases} 1 & i = j \\ 0 & \text{else} \end{cases}$$

When, indexing the nodes such that the target node gets index 0, we get a transistion matrix of the following type. Note that the vector r and the matrix Q are determined by the network according to the first rule for the transition matrix.

$$P = \begin{pmatrix} 1 & 0 \\ r & Q \end{pmatrix}$$

Then one can determine the hitting time for each starting state by:

$$t = (I - Q)^{-1} \begin{bmatrix} 1 \\ 1 \\ . \\ . \\ 1 \end{bmatrix}$$

Another approach, which is explained also by Peter and Snell (2000) and David *et al.* (2006), is that a random walk on a network can be modeled by into an electrical network. The theory to understand that approach is quite broad and we think that it is not relevant to know the theory exactly to understand the behavior of our algorithm. Therefore, we only provide a few comments. As the algorithm relies on the number of edges, respectively the connectivity of the network it is clear that bad cases with long hitting time can be generated. Such a case would be, if there is a certain edge, which has to be taken. For example, the source node lies in one section of the network and the target node in other one. The only connection between these two sections is one edge. If the FANT has to reach the target it has to take this edge. This is a quite unfavorable situation for the FANT. After this short discussion about hitting times we like to have a closer look at the behavior of the FANTs and BANTs and to elaborate the whole algorithm. Let us start with the FANTs.

Forward Ants (FANT): In the beginning of this study we were talking about one FANT moving around. This was just to understand the basic concept. In real, we are sending out >1 FANT. Let us, first try to understand why and then have a look at how we are sending them out in our implementation. Just imagine we just sent out one FANT. The quality of the route we are going to find will not be very well. If the FANT has not taken any loops the BANT going back will take the same route the FANT took. This is quite probably a rather bad one. That is the reason, we send out >1 FANT and with that most nodes near the source know a quite a short route to the source and the route the BANT takes is not that bad. We defined several parameters, the number of FANTs μ that are sent out at the beginning. After some time we send out a new sequence of FANTs. Each time we sent out a new sequence of FANTs we first multiply the number of FANT with a given factor α. The number of FANTs sent out is therefore, exponentially increasing. Another parameter is the TTL of the FANTs. Now we describe what a node has to do if it receives a FANT. The FANT reaching a node X a FANT first updates the routing table. Afterwards it is tested; if the FANT reached

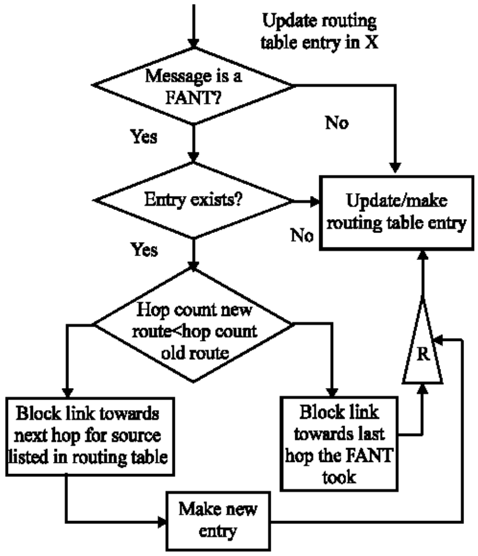

Fig. 1: Making routing table entries

the end of its lifetime in this case the ant will be destroyed. If node X is not yet the target node the FANT updates the last hop, the second last hop and the hop count in its body. If the target is reached, a BANT is sent back towards the source. In static and nice cases the algorithm research without any problem. If the network is dynamic we must face the fact that the neighbor Y could have moved out of the connection radius of the node. As next some details about how we make routing table entries.

Updating routing information: The design procedure as shown in Fig. 1, we decided that each message arriving in a node X updates or make the routing table entry for the source it came from. It does not put anything in the routing table about where it is going to. We based this decision on the following facts:

- First the message is coming from a source. Therefore, it is carrying the newest information about the path to the source the node X is able to get. That is the main reason why we use this information
- You may wonder why we are not updating the routing information for the target. We will see, that as you can imagine also our routes are not valid forever, if we have to deal with mobility. Therefore, we locally have to take a path towards a target but this information may be out of date and not valid anymore. So these two points illustrate the reason why a message makes only entries towards the source, if it enters a node X

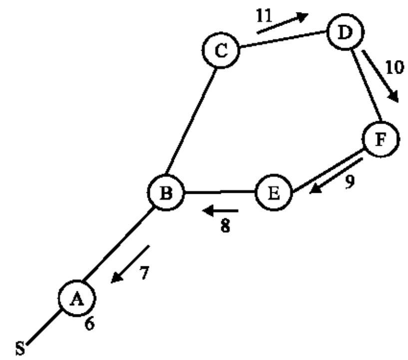

Fig. 2: Loop scenario

Preventing loops: The problem we encountered is that with the starting procedure we developed so far the algorithm produced loops. To illustrate the problem consider in Fig. 2, the FANT took the route.

ABEFDC and writes in each node the routing table entries as drawn in the figure. Arriving again in B we have the problem that when we just overwrite the existing entry we produce a loop. If a returning BANT reaches one of the nodes and follows the entries made, it ends up in a loop and will never reach the source. The idea behind the concept is the following: When a FANT arrives from a certain neighbor (Fig. 2) the particular FANT is not allowed to make a routing table entry. Now how do we detect if FANTs coming over a particular link are not allowed to make entries? And what is the profit of this procedure? When a FANT arrives in node B the node checks, if already a routing table entry exists. If the old one is better the node designates the link between C and B as blocked and it does not allow further FANTs arriving over this link to make routing table entries. In the case of Fig. 2 node B would not allow FANTs arriving from C to make routing table entries anymore. But when is a route better than another one? Comparing hop counts does not give you necessarily the correct answer. Nevertheless, we decided to base our decision on the hop counts. But additionally we introduce a lifetime for the routing table entries and for the blocked links. The algorithm may over a small period block the wrong edge, but due to the lifetime after a while when the entries expire the procedure is starting again from the beginning. To conclude the study, about making routing table entries consider the following routing table entry in node F (Fig. 2).

Target	Next hop	After next hop	Hops to target	Life tim
S	E	B	9	8

Backward Ants (BANT): As said, if a FANT reached the target, we send a BANT back to the source. The procedure is very similar to the one for the FANT. We first create a routing table entry. We then test, if the BANT reached the end of its lifetime. This is just a security check that the BANT is not walking around forever. This could for example happen, if a BANT ends up in a loop. Reaching a node X, a BANT first updates the routing table. A new routing table entry is made. Afterwards it is tested, if the BANT reached the end of its lifetime. In this case the ant will be destroyed. If the target is reached, the connection is established. If node X is not yet the target node, the BANT updates the last hop, the second last hop and the hop count in its header. Then the node looks up in his routing table for the next hop towards the source. The node is then trying to send the message to this node. If the node is due to mobility not available anymore the node starts the route recovery procedure.

RESULTS AND DISCUSSION

Performance evaluation: We evaluate the performance of Multi Ant based Routing Protocol using simulations and compare them with existing hybrid ZRP and ARA.

Simulation environment: We implemented Multi Ant based Routing Protocol (MARP) model using GLOMOSIM (refer in UCLA Parallel Computing Laboratory, Software Package, Online) and web site (http://pcl.cs.ucla.edu:).As simulation software we used GloMoSim is a scalable simulation environment for wireless and wired networks systems developed initially at UCLA Computing Laboratory. It is designed using the parallel discrete-event simulation capability provided by a C-based parallel simulation language, Parsec. GloMoSim currently supports protocols for purely wireless networks. All our simulation scenarios are derived from the base scenario used in Gunes *et al.* (2002), which is an important reference. In this base scenario, each node has a radio range of 250 m and the channel capacity is 2 Mb sec^{-1}. As a Medium Access Control (MAC) protocol, we used the IEEE 802.11 Distributed Coordination Function. The mobility model uses the random waypoint model (Broch *et al.*, 1998) in a rectangular field (2200×6000 m). Nodes move randomly within the rectangular field. A node chooses a speed between the minimum (0 m sec^{-1}) and the maximum speed. To change node mobility, we vary the maximum speed of the nodes. We considered the continuous mobility case (i.e., no pauses). The number of the nodes was varied to change node density. Simulations were run for 100 sec. Traffic sources were continuous bit rate. Source and destination pairs (sessions) were randomly selected with

Table 1: Simulation parameters of Ant based routing protocol

Parameter	Value
Packet size	512 byte
No of nodes	100-300
Simulation time	100 sec
Simulation area for base senario	2200×6000 m
Mobility model	Random waypoint
Data traffic	Constant bit rate (20)
Transmission range	300 m
Data rate	2 Mbit s^{-1}
	Ant algorithm

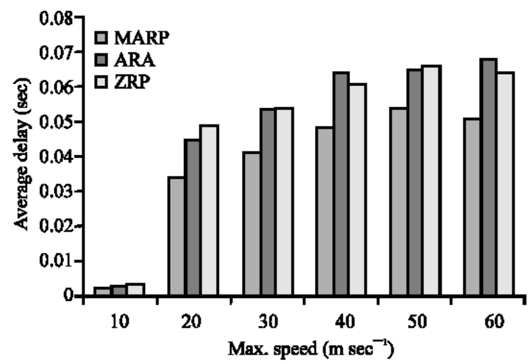

Fig. 3: Delay with varying node mobility

uniform probabilities. The number of sessions was ten and the traffic rate was 4 packets sec^{-1}. The data packets size was 512 bytes. The model parameters that have been used in the Ant routing are summarized in Table 1.

We compare our MARP protocol with ZRP and ARA. We evaluate mainly the performance according to the metrics given.

We vary the following parameters

- No. of nodes
- Node speed
- No of multimedia flows

And measure the Packet Delivery ratio, End-to-end delay, packet dropped and Route discovery frequency.

Simulation results

Varying mobility: To evaluate capability of the algorithm for different node mobility, we changed node mobility by varying the maximum speed. The number of nodes and sessions was fixed at 100 nodes and 10 sessions, respectively. The proposed protocol (MARP) exhibits the smallest end-to-end delay (Fig. 3). ZRP incurs the largest delay. For example, when the maximum speed is 20 m sec^{-1}, the delay of proposed algorithm, ARA and ZRP is 0.034, 0.045 and 0.049 sec, respectively. The proposed algorithm decreases the delay of ZRP by

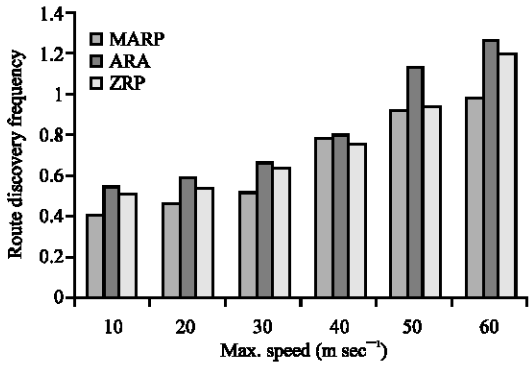

Fig. 4: Route discovery frequencies

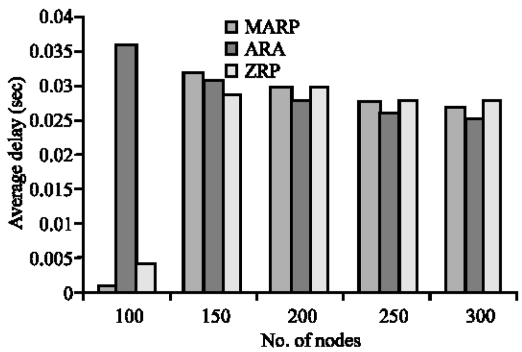

Fig. 5: Delay with varying node density

30.61% (=(0.049-0.034)/0.049×100) and decreases delay of ARA by 24.4% (=(0.045-0.034)/0.045×100) with the increase of the maximum speed, the proposed algorithm further decreases the delay, compared with the ZRP and ARA protocol. This result shows that the proposed algorithm is effective especially in situations, where nodes move frequently.

The proposed routing algorithm has smaller route discovery frequency than the ZRP and ARA (Fig. 4). In the MARP, ARA and ZRP, a new route discovery is invoked only when all paths fail. Therefore, the result of a smaller route discovery frequency of the proposed routing algorithm implies that it can increase opportunities for setting up alternative paths.

This suggests that the increase of available alternative paths decreases data packets that are affected by the route discovery latency.

Varying node density: Compared with existing hybrid routing namely ZRP and ARA, the Proposed Routing Protocol (MARP) need the additional existence of nodes that provide alternate paths. To evaluate the availability of multiple paths, we make the density of a node low by reducing the number of nodes in the rectangular field. Other parameters such as node mobility

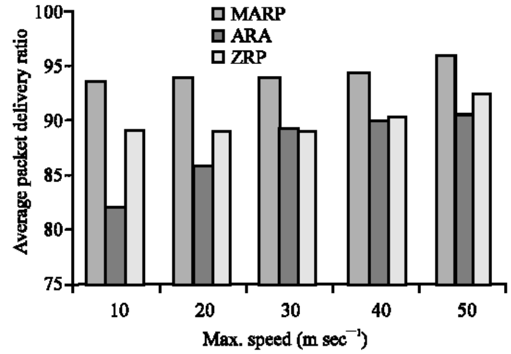

Fig. 6: Average packet delivery ratio with varying node mobility

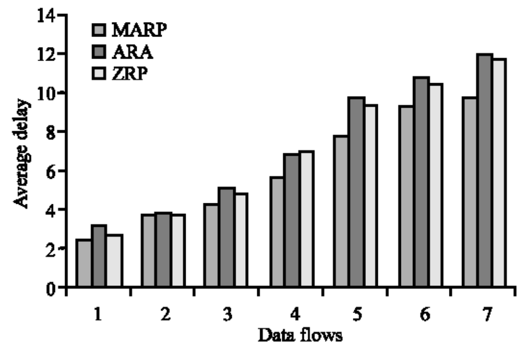

Fig. 7: Average delay with multimedia data flows

(max. speed 20 m sec^{-1}) and offered traffic load (20% traffic sources) are kept constant. Figure 5 shows the average end-to-end delay when changing the node density. When the number of nodes is 150, the end-to-end delay of both MARP and ZRP protocols is similar. But delay of ARA protocol is high compared with MARP and ZRP. This result shows that it is difficult to demonstrate availability of multiple paths, since there are few nodes that offer alternative routes. However, by increasing the node density, the proposed routing protocol reduces the delay compared with ZRP as well as ARA. This means that the proposed routing algorithm efficiently uses the nodes that can offer alternate paths, compared with zone routing protocol ZRP and ARA.

Packet delivery ratio: We will discuss the robustness of the routing protocols. Figure 6 shows the delivery rate with varying node mobility i.e., the part of packets a certain routing protocol was able to deliver properly. This value is important, since it describes the performance, which transport protocols will see, i.e., the throughput is restricted by this value.

In the case with low speed, i.e., high topology changes, only proposed protocol (MARP) are able to

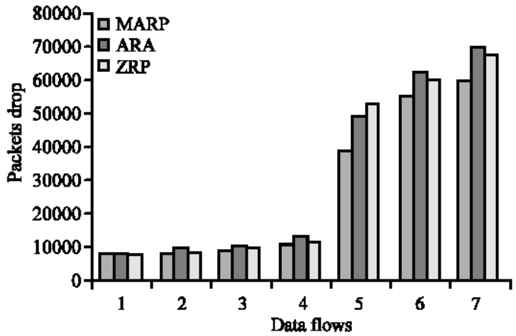

Fig. 8: Packets drop with multimedia data flows

deliver >94% of the data packets. Whereas, ARA and ZRP are able to deliver only 88-90% of the data packets. In situations with very high dynamics, >40 mill sec of speed, the proposed protocol are able to deliver shows the best performance than ARA and ZRP. With less dynamic, up to 30 mill sec of speed, the ARA and ZRP is very close to proposed protocol. They are able to deliver >90% of the packets in situations with >40 mill sec of speed.

Multimedia data flows: Finally, we vary the number of multimedia data flows from 1-7 and studied the performance of average delay and drop. Figure 7 and 8 give the results for delay and drop, respectively. We can observe that MARP clearly outperforms hybrid protocol ZRP and ARA in 2 cases.

CONCLUSION

The proposed protocol named as Multi Ant based Routing Protocol (MARP) allows sending data packets by alternative paths, without executing a new route discovery, if the primary path breaks. Using simulation models, we evaluated the multi ant based routing protocol and compared them with hybrid ZRP and ARA. The results, in a variety of mobility and node densities, show the proposed protocol achieves better performance in terms of average end-to-end delay and route discovery latency with high connectivity. Such low end-to-end delay cannot be achieved from both ZRP and ARA protocol, because of their inherent shortcomings and proposed protocol gives better delivery ratio compared with ZRP. Finally, the multimedia data are sending over the stable, failure-free paths than other 2 protocols. This makes Multi Ant based Routing Protocol (MARP) is suitable for real-time data and multimedia communication.

As future research, we will plan to investigate the utility of the proposed protocol for other networks such as the sensor network that requires consideration of constrained memory and power resources of nodes.

REFERENCES

Broch, J., D. Maltz, D. Johnson, Y. Hu and J. Jetcheva, 1998. A Performance Comparison of Multi Hop Wireless AdHoc Network Routing Protocols, Carnegie Mellon MONARCH Project, http://www.monarch.cs.cmnu.edu/.

Corson, S. and J. Macker, 1999. Mobile AdHoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations Corson Network Working Group.

David, A. Levin, Yuval Peres and Elezabeth L. Wilmer, 2006. Marcov chains and Mixing Times. University of Organ. URL: http://www.uoregon.edu/dlevin.

Dorigo, M., V. Maniezzo and A. Colorni, 1996. The Ant System: Optimization by a Colony of Cooperating Agents, IEEE Trans. Syst. Man and Cybernetics Part B, 26 (1): 29-41.

GoldBerg, D.E., 1991. Genetic Algorithms in Search, Optimization and Machine Learning, Addison Wesley. Ant colony optimization from http://en.wikipedia.org/wiki/Ant_colony_optimization Courtallam.

Gunes, M., U. Sorges and I. Bouazizi, 2002. ARA The Ant Colony Based Routing Algorithm for MANETs. In: Stephan Olariu (Ed.). Proc. 2002 ICPP Workshop on AdHoc Networks (IWAHN 2002), IEEE Computer Society Press, pp: 79-85.

Hertz, J., A. Krogh and R.G. Palmer, 1991. Introduction to the Theory of Neural Computation. Addison-Weseley.

Marwaha, S., C.K. Tham and D. Srinivasan, 2002. Mobile Agents Based Routing Protocol for Mobile AdHoc Networks. In: Proceedings of IEEE Globecom, Department of Electric and Computer Engineering, Nat University of Singapore, Singapore, 1: 163-167-167. ISBN: 0-7803-7632-3.

Perkins, C.E. and E.M. Royer, 1999. AdHoc on demand distance vector routing. Proc. IEEE Workshop on Mobile Comput. Syst. Applications WMCSA., pp: 234-242.

Peter, G. Doyle and J.L. Snell, 2000. Random walks and electric networks. Arxiv.math/0001057v1.