An Ant Colony Optimization Algorithm for the Mobile Ad Hoc Network Routing Problem Based on AODV Protocol

Ahmed M. Abdel-Moniem
Dept of Computer Science
Assiut University
Assiut, Egypt
ahmedcs982@hotmail.com

Marghny H. Mohamed
Dept of Computer Science
Assiut University
Assiut, Egypt
marghny@aun.edu.eg

Abdel-Rahman Hedar, IEEE Member
Dept of Computer Science
Assiut University
Assiut, Egypt
hedar@aun.edu.eg

***Abstract*—In this paper, we present a modified on-demand routing algorithm for mobile ad-hoc networks (MANETs). The proposed algorithm is based on both the standard Ad-hoc On-demand Distance Vector (AODV) protocol and ant colony based optimization. The modified routing protocol is highly adaptive, efficient and scalable. The main goal in the design of the protocol was to reduce the routing overhead, response time, end-to-end delay and increase the performance. We refer to the new modified protocol as the Multi-Route AODV Ant routing algorithm (MRAA).**

***Keywords*-Mobile Ad-hoc network, Routing, Ant Colony Optimization.**

I. Introduction

The problem of MANET can be summarized in the answer of this question, how to find the route between the communication end-points. One of the reasons is that routing in MANETs is a particularly challenging task due to the fact that the topology of the network changes constantly and paths which were initially efficient can quickly become inefficient or even infeasible. Moreover, control information in the network is very restricted. This is because the bandwidth of the wireless medium is very limited, and the medium is shared. It is therefore important to design algorithms that are adaptive, robust and self-healing. Moreover, they should work in a localized way, due to the lack of central control or infrastructure in the network [1], [4].

In this paper, we present a modification to optimize the well-known MANET routing protocol, Ad-hoc On-demand Distance Vector (AODV), using swarm intelligence. Ant colony optimization (ACO) algorithms compose a main class in swarm intelligence. And it considers the ability of simple ants to solve complex problems by cooperation. The interesting point is that the ants do not need any direct communication for the solution process, instead they communicate by stigmergy. The notion of stigmergy means the indirect communication of individuals through modifying their environment. Several algorithms which are based on ant colony problems were introduced in recent years to solve different problems, e.g. optimization problems, including MANET [4], [5]. To show that the approach of ACO has the potential to be applied to optimize the existing ANT-Free Mobile Ad-Hoc Routing Protocols, we have taken existing implementation of AODV then applied the ACO algorithm to AODV without removing or affecting any of pre-existing AODV mechanisms. Results are based on simulations made with the current implementation in OPNET Modeler V.14 [11], which comply to the published IETF version of AODV routing protocol [8].

The remainder of this paper is organized as follows. In Section II, we present the Mobile Ad Hoc Network Routing Problem. In Section III, we introduce the ant colony optimization meta heuristic for MANETs. In Section IV, we explain the new routing algorithm (MRAA) in detail. In Section V, we show some simulation results to show the ability of the ACO to improve routing performance. Finally, a conclusion is given in Section VI.

II. Mobile Ad Hoc Network Routing Problem

Mobile Ad-Hoc Networks(MANETs) nodes are typically distinguished by their limited power, processing, and memory resources as well as high degree of mobility. In such networks,the wireless mobile nodes may dynamically enter the network as well as leave the network. Due to the limited transmission range of wireless network nodes, multiple hops are usually needed for a node to exchange information with any other node in the network. Thus routing is a crucial issue to the design of a MANET [7].

Routing algorithms in conventional wired networks are usually based algorithms that require periodic routing advertisements to be broadcast by each router. Which are clearly not efficient for the type of dynamic changes which may occur in an ad-hoc network. Also, routers in conventional networks do not generally move around and only rarely leave or join the network. In an environment with mobile nodes, the changing topology will not only trigger frequent recomputation of routes but the overall convergence to stable routes may be infeasible due to the high-level of mobility [7].

So, there are numerous issues to consider when deploying MANETs. The following are some of the main issues [7]:

1) Ad hoc networks may be deployed in unknown environments where node failures may occur frequently.
2) Communication through the wireless medium is unreliable and subject to errors.
3) Nodes in a MANET are typically battery-powered as well as limited in storage and processing capabilities.
4) The topology in an ad hoc network may change constantly due to the mobility of nodes.

Routing protocols for MANETs must deal with these issues to be effective. So our goal in this paper is to try to apply the ant-colony optimization technique, that has proved its effectiveness in wired networks routing protocols, to optimize the existing MANET routing protocols. In our case, we applied ACO to AODV routing protocol to overcome the addressed issues and to improve its performance.

A. Ad-Hoc On-Demand Distance Vector

Routing protocols for a MANET can be classified as proactive (table-driven) and reactive (on-demand), depending on how they react to topology changes for more information, see [12]. A host running a proactive protocol propagates routing-related information to its neighbors whenever a change in its link state is detected. The information may trigger other mobile hosts to recompute their routing tables and further propagate more routing-related information. The amount of information propagated each time is typically proportional to the scale of the MANET. an example of proactive protocol is destination sequenced distance vector (DSDV) [10].

Observing that a proactive protocol may pay costs to construct routes even if mobile hosts do not have such need, thus wasting the limited wireless bandwidth, many researchers have proposed using reactive-style protocols, in which routes are only constructed on-demand. Many reactive protocols have been proposed based on such on-demand philosophy, such as dynamic source routing (DSR) [2], ad hoc on-demand distance vector routing (AODV) [9], and temporally-ordered routing algorithm (TORA) [**?**]. In the following, we review AODV as one of the reactive (on-demand) routing protocols:

When a node *S* needs a route to some destination *D*, it broadcasts a ROUTE-REQUEST message to its neighbors, including the last known sequence number for that destination. The ROUTE-REQUEST is flooded in a controlled manner through the network until it reaches a node that has a route to the destination. Each node that forwards the ROUTE-REQUEST creates a reverse route for itself back to node *S*.

When the ROUTE-REQUEST reaches anode with a route to *D*, that node generates a ROUTE-REPLY that contains the number of hops necessary to reach *D* and the sequence number for *D* most recently seen by the node generating the ROUTE-REPLY. Each node that participates in forwarding this reply back toward the originator of the ROUTE-REQUEST(node *S*), creates a forward route to *D*. The state created in each node along the path from *S* to *D* is hop-by-hop state that is, each node remembers only the next hop and not the entire route, as would be done in source routing.

In order to maintain routes, AODV normally requires that each node periodically transmit a HELLO message, with a default rate of once per second Failure to receive three consecutive HELLO messages from a neighbor is taken as an indication that the fink to the neighbor in question is down. Alternatively, the AODV specification briefly suggests that a node may use physical layer or link layer methods to detect LINK-BREAKAGES to nodes that it considers neighbors as described in [9], [1].

III. Ant Colony Optimization for MANET Routing Problem

The ant colony optimization meta-heuristics is a particular class of ant algorithms. Ant algorithms are multi-agent systems, which consist of agents with the behavior of individual ants [3].

A. Real Ants Mechanism

The basic idea of the ant colony optimization meta heuristic is taken from the food searching behavior of real ants. When ants are on they way to search for food, they start from their nest and walk toward the food. When an ant reaches an intersection, it has to decide which branch to take next. While walking, ants deposit pheromone, which marks the route taken. The concentration of pheromone on a certain path is an indication of its usage. With time the concentration of pheromone decreases due to diffusion effects. This property is important because it is integrating dynamic into the path searching process. This behavior of the ants can be used to find the shortest path in networks. Especially, the dynamic component of this method allows a high adaptation to changes in mobile ad-hoc network topology, since in these networks the existence of links are not guaranteed and link changes occur very often [5].

B. Why ant colony optimization meta-heuristic suits to ad-hoc networks

The simple ant colony optimization meta-heuristic shown in the previous section illustrates different reasons why this kind of algorithms could perform well in mobile multi-hop ad-hoc networks. We will discuss various reasons by considering important properties of mobile ad-hoc networks [5].

- ACO is based on agent systems and works with individual ants. This allows a high adaptation to the current dynamic topology of the network.
- ACO is based only on local information, i.e., no routing tables or other information blocks have to be transmitted to neighbors or to all nodes of the network.

- It is possible to integrate the connection/link quality into the computation of the pheromone concentration.
- Each node has a routing table with entries for all its neighbors, which contains also the pheromone concentration. Thus, the approach supports multi-path routing.

IV. The MRAA Algorithm Elements

Hereby, we describe the different procedures used to handle the different events of the protocol. the events to be handled are neighbor connectivity, route establishment request, route establishment reply, route expiry, connection loss and local repair. More formal details of these components are shown below.

A. Initialization

Construct the pheromone array, network model and initialize other state and statistics variables.

B. Neighbor Connectivity

When a packet from a new neighbor has just arrived, perform the following procedure.

Procedure 4.1: Neighbor_Connectivity

1. *If the arriving packet is from a new neighbor then:*
   - *add a route to the new neighbor in the routing table with initial pheromone value of 0.0,*
   - *update the pheromone array to reflect the new neighbor as a candidate next hop for all other nodes in the network but with initial value of pheromone of 0.0, and*
   - *increment the number of available active neighbors for this nodes.*
2. *Else if the neighbor already exists:*
   - *update the expiry time of the connection for this neighbor.*

C. Route Establishment Request

Sending forward ant agents to find a route to the requested destination because no route exists, by applying the following procedure.

Procedure 4.2: Route_Establishment_Request

1. *If this nodes is the source node then:*
   - *initialize the memory of the forward ant,*
   - *if the route to destination does not exist and there exists no highest pheromone neighbor then, Broadcast a forward ant to all available active neighbors, and*
   - *else if the route to destination does not exist but there exists highest pheromone neighbor then, Send the forward ant highest pheromone active neighbor.*
2. *Else if it is an intermediate node then:*
   - *update the memory of the forward ant,*
   - *if the route to destination does not exist and there exists no highest pheromone neighbor then :*
     Broadcast a forward ant to all available active neighbors, and
   - *else if the route to destination does not exist but there exists highest pheromone neighbor then :*
     Send the forward ant to all highest pheromone neighbor active neighbors.
3. *Else if it is the destination node:*
   - *send a backward ant to the source to give it the feedback of the route.*

D. Route Establishment Reply

Sending backward ant agent to inform the source with a route to destination and apply pheromone as it returns back, by acting up on the following procedure.

Procedure 4.3: Route_Establishment_Reply

1. *If this nodes is the source node then:*
   - *calculate the pheromone value of the arriving backward ants using their memory using Equation,*

     $$P = \frac{N_h}{N_n} + \frac{L_C}{100} + T, \qquad (1)$$

     where, P is pheromone of a node to Dest, N_h is number of hops from Src to Dest, N_n is number of nodes in the network,L_C is link capacity of the next neighbor, and T is trip time of the ant from Src to Dest,
   - *update the pheromone array of the node to the pheromone value for each neighbor, and*
   - *find the highest pheromone valued neighbor and update the route entry for that destination to make the node forward packets to that highest valued neighbor.*
2. *Else if it is an intermediate node then:*
   - *update the reverse memory of the backward ant, and 2.1. if the route to destination does not exist then:*
     - *create a new route table entry for that destination node in routing table,*
     - *update the pheromone array of the node to the new pheromone value of that neighbor using equation ,*
     - *find the highest pheromone valued neighbor and update the route entry for that destination to make the node forward packets to that highest valued neighbor, and*

- *increase the number of active destinations by one.*

2.2. Else if the route to destination exists then:

- *increase the pheromone value of that neighbor to that destination using the following equation,*

$$Pn = .75 * Po + .25 * P \quad (2)$$

where, Pn is the new pheromone value, Po is the old pheromone value, and P is pheromone value calculated using eq(1),

- *decrease the pheromone value of all other neighbors to that destination using the following equation, and*

$$Pn = .75 * Po - .25 * P \quad (3)$$

where, Pn is the new pheromone value, Po is the old pheromone value, and P is pheromone value calculated using eq(1),

- *find the highest pheromone valued neighbor and update the route entry for that destination to make the node forward packets to that highest valued neighbor.*

3. Else if it is the destination node:

- *initialize the reverse memory of the backward ant,*
- *copy the memory of arriving forward ant to the memory of backward ant, and*
- *make the backward ant follow the same path which is stored in its memory to the source leaving behind a pheromone value accordingly on each hop in the back path to source node.*

E. Route Expiry

After some time route expiry timer timeouts, then the route entry must be removed from routing table and pheromone array, by going through the following procedure.

Procedure 4.4: Route_Expiry

1. If the route expiry timeout event has occurred:

- *remove that destination from the pheromone array,*
- *remove the routing table entry, and*
- *decrement number of active destinations.*

F. Connection Loss

After some time if no packets arrive whether data, ants or hello from a neighbor, then we do the following actions.

Procedure 4.5: Connection_Loss

1. If the connection timeout event has occurred:

- *reset the pheromone value of that neighbor to 0.0,*
- *remove the routing table entries that route though that neighbor,*
- *if any other available neighbor has a non-zero pheromone value to that destination update the routing table to use the other neighbor (Automatic Repair using Multi-Ant Routes), and*
- *decrement number of active neighbors.*

G. Local Repair

If some request packet arrive and no routing entry exists for it, then the algorithm try to do local repair as follow.

Procedure 4.6: Local_Repair

1. If local repair is enabled on this node:

- *if any other available neighbor has a non-zero pheromone value to that destination update the routing table to use the other neighbor (Automatic Repair using Multi-Ant Routes) then, Send all packets in queue to the destination using this new neighbor in the route, and*
- *Otherwise use default built-in AODV Local Repair strategy.*

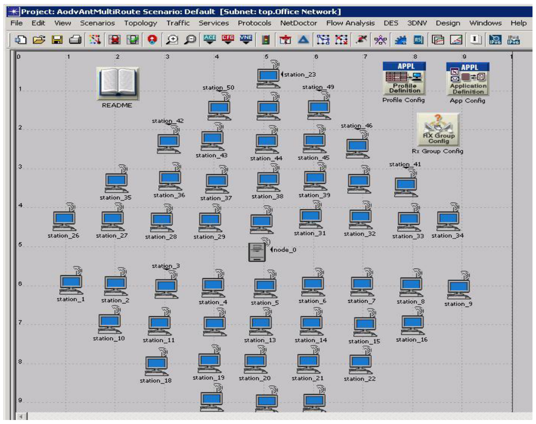
V. Simulation and Performance Evaluation

A. Simulation Environments

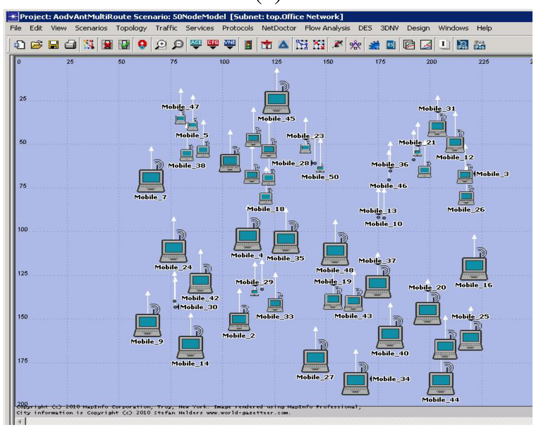
In order to evaluate performance of the proposed protocol, we use the network simulator OPNET Modeler V.14 and make modification to the AODV. We used more than one case to establish that MRAA was an improvement over AODV, the following are the explanation of the four chosen cases:

1) Number of 50 Nodes and running the same MANET Routing Protocol with default parameters but the simulation was run for 1 hour, FTP is configured to generate medium traffic loads and is allowed to make only serial connections with 3 repeats during whole simulation. For traffic source, FTP Application is used generating data starting randomly using uniform distribution from the 100th second to the end of simulation and each time the application continue to send data for 10 seconds three times during whole simulation to random chosen destinations. However, The wireless nodes are fixed rather than mobile as shown in Fig 1.[1]
2) the same model exactly as above but with Number of 50 Nodes and running the same MANET Routing Protocol with default parameters but the simulation was run for 5 hours, FTP is configured to generate high traffic loads and is allowed to make concurrent

[1] Note that default parameters differs from protocol to another but for aodv the route expiry timeout is 3 seconds, allowed hello is 2 times, hello interval is uniform(1,1.1), net diameter is 35 hop, and local repair is enabled.



(a)



(b)

Figure 1. (a) A figure showing the 50 stationary nodes network topology and their positions (b) A figure showing the 50 mobile nodes network topology and their initial positions, note that the upward arrow does not mean they all move upward but they move according to random way-point in random directions with changing speeds and making some stops during the simulation

connections with unlimited repeats during whole simulation as shown in Fig 1.

3) Number of 50 Nodes positioned randomly in are of 300*200 meters and running the same MANET Routing Protocol with default parameters but the simulation was run for 15 minutes. For traffic source, data was generated with inter-arrival time of packets is exponentially distributed with mean outcome of 10 seconds, for packet size we use exponential distribution with mean outcome of 1024 byte and the nodes starts sending from start to end of simulation to randomly chosen destinations. As a mobility model, we use the random waypoint model as shown in Fig 1. In the model each node starts its journey from a random location to another random location with a uniformly chosen speed. Once the node reaches its destination, another new random destination is targeted starting from the previous random destination after a pause time.

4) the same model exactly as above but with Number of 100 Nodes positioned randomly in are of 600*400 meters as shown in Fig 1.

B. *Performance metrics*

Five key performance metrics are evaluated in our experiments:

- Throughput : Represents the total number of bits (in bits/sec) forwarded from wireless LAN layers to higher layer which also means received throughput at the destinations of the data sources.
- WLAN network load : Represents the total number of bits (in bits/sec) submitted to wireless LAN from higher layer which also means the combined sending rate of all data sourcesNote that without any retransmission, the ratio of throughput and offered load is simply the packet delivery ratio..
- WLAN end-to-end delay : this includes all possible delays caused by buffering during route discovery latency, queuing at the interface queue, retransmission delays at the MAC, and propagation and transfer time.
- Normalized routing overhead : this metric has two variants: packet overhead is the number of routing packets transmitted per data packet delivered at the destination.
- Average loss ratio : average number of data packet dropped not routing protocol packets dropped.

C. *Simulation results*

As we can see from results of Table I shown below, that in the case of stationary stations and their no load and concurrent connections in the network and only the users allowed to make 3 FTP sessions during the simulation which lasts for only 1 hour that the traditional AODV outperforms MRAA in the routing overhand, no of dropped packets, FTP up/down-load time and simulation time but MRAA out performs the AODV in throughput, network load and the average end-end WLAN delay. But from results of simulation and not shown in table number of dropped AODV packets in case of traditional AODV was 3.5 and in case of MRAA was 11.3.

However, as we an see from results of table II shown below, in the case of stationary stations and they are making high load traffic and allowed to make unlimited number of concurrent connections in the network, in addition,the users allowed to make unlimited number of FTP sessions during the simulation which lasts for 5 hours that the traditional MRAA outperforms AODV in the routing overhand, network load, FTP up/down-load timed and simulation time but AODV out performs the MRAA in throughput,dropped packets and the average end-end WLAN delay. But note that the difference in the average delays of AODV and MRAA are not great at all and also note that from results of simulation and not shown in table number of dropped

AODV packets in case of traditional AODV was 22,488.25 and in case of MRAA was 11,001.05 which means there is more wasted AODV packets in case of traditional AODV than MRAA.

Table I
SIMULATION RESULTS FOR CASE 1. (A) A TABLE SHOWING AVERAGE VALUES FOR AVERAGE DELAY, THROUGHPUT, NETWORK LOAD, (B) A TABLE SHOWING ROUTING TRAFFIC SENT, ROUTING TRAFFIC RECEIVED AND NUMBER OF DROPPED PACKETS, (C) A TABLE SHOWING FTP DOWNLOAD RESPONSE TIME, FTP UPLOAD RESPONSE TIME AND SIMULATION DURATION.

(A)

Protocol Name -	Average Delay sec	Throughput bit/sec	Network load bit/sec
AODV	0.006848	20,354.80	18,606.08
MRAA	0.0007845	23,483.96	7,783.63

(B)

Protocol Name -	Routing traffic(S) bits	Routing traffic(R) bits	Loss no
AODV	1,693.27	4,995.96	0.0
MRAA	6,758.01	22,458.53	0.02694

(C)

Protocol Name -	FTP Download sec	FTP Upload sec	Simulation Time sec
AODV	0.7450	0.6122	5
MRAA	1.970	1.867	14

Table II
SIMULATION RESULTS FOR CASE 2. (A) A TABLE SHOWING AVERAGE VALUES FOR AVERAGE DELAY, THROUGHPUT, NETWORK LOAD, (B) A TABLE SHOWING ROUTING TRAFFIC SENT, ROUTING TRAFFIC RECEIVED AND NUMBER OF DROPPED PACKETS, (C) A TABLE SHOWING FTP DOWNLOAD RESPONSE TIME, FTP UPLOAD RESPONSE TIME AND SIMULATION DURATION.

(A)

Protocol Name -	Average Delay sec	Throughput bit/sec	Network load bit/sec
AODV	0.20809	901,054.14	1,035,794.03
MRAA	0.24189	266,315.41	263,84.13

(B)

Protocol Name -	Routing traffic(S) bits	Routing traffic(R) bits	Loss no
AODV	162,128.87	77,792.89	0.00322
MRAA	17,488.72	48,654.78	5.160

(C)

Protocol Name -	FTP Download sec	FTP Upload sec	Simulation Time sec
AODV	522.97	350.48	2700
MRAA	53.42	25.555	1320

As we can see from results of Tables III and IV shown below, that in the case of mobile stations traditional AODV outperforms MRAA in only the number of dropped data packets and throughput, but MRAA out performs the AODV in network load, routing overhand, and the average end-end WLAN delay. But from results of simulation and not shown in table number of dropped AODV packets in case of traditional AODV was 30.8 (50 node) and 65.76 (100 node) and in case of MRAA was 27.68 (50 node) and 58.70 (100 node), and the greater throughput is due to the high routing traffic that is generated, also the simulation time in case of AODV took 120 sec (50 node) and 2078 (100 node) to finish nearly twice longer than MRAA that took only 79.6 sec (50 node) and 1015 sec (100 node) sec that is due to that AODV making more processing per node each time a route not found to find a fresh new route.

Also, as we an see from results of tables III and IV, that in the case of mobile stations when traditional AODV is compared to the other MANET routing protocols (OSLR, GRP, DSR) most of them outperforms AODV. However MRAA results could show that it can compete and with respect to some performance metrics it can outperform the other MANET routing protocols.

Table III
SIMULATION RESULTS FOR CASE 3. (A) A TABLE SHOWING AVERAGE VALUES FOR AVERAGE DELAY, THROUGHPUT, NETWORK LOAD, (B) A TABLE SHOWING ROUTING TRAFFIC SENT, ROUTING TRAFFIC RECEIVED AND NUMBER OF DROPPED PACKETS.

(A)

Protocol Name -	Average Delay sec	Throughput bit/sec	Network load bit/sec
AODV	0.003082	2,451,903.38	77,287.38
MRAA	0.000669	744,958.77	64,478.37
OSLR	0.00040425	2,475,661.72	56,414.14
GRP	0.0003099	291,391.88	18,271.24
DSR	0.00347	13,387.74	8,576.05

(B)

Protocol Name -	Routing traffic(S) bits	Routing traffic(R) bits	Loss no
AODV	75,077.51	2,449,693.51	0.0
MRAA	15,359.42	695,840.87	1.3333
OSLR	44,762.75	2,194,792.82	0.0667
GRP	5,822.13	224,319.82	0.4169
DSR	2,244.12	7,056.89	0.0

Table IV
SIMULATION RESULTS FOR CASE 4. (A) A TABLE SHOWING AVERAGE VALUES FOR AVERAGE DELAY, THROUGHPUT, NETWORK LOAD, (B) A TABLE SHOWING ROUTING TRAFFIC SENT, ROUTING TRAFFIC RECEIVED AND NUMBER OF DROPPED PACKETS.

(A)

Protocol Name -	Average Delay sec	Throughput bit/sec	Network load bit/sec
AODV	0.00714	13,416,041.64	338,547.37
MRAA	0.00578	3,495,817.96	156,678.03
OSLR	0.00056417	17,893,758.44	192,775.01
GRP	0.0003479	1,091,655.67	36,655.00
DSR	0.00694	58,201.28	17,759.09

(B)

Protocol Name -	Routing traffic(S) bits	Routing traffic(R) bits	Loss no
AODV	333,985,985.74	13,411,484.55	0.0
MRAA	51,504.29	3,390,649.60	2.7578
OSLR	169,433.99	16,772,453.80	0.143
GRP	16,390.27	1,069,449.16	0.0
DSR	5,233.99	26,682.38	0.88

D. Results Discussion

In order to compare the performance of the ant-colony optimized protocol with AODV. Previous tables show the end-to-end delay, Throughput, Network Load, sent routing traffic, received routing traffic, number of dropped data packets, simulation duration. As the more mobile nodes become rather than being stationary, the performance gain by optimization algorithm becomes more significant. The proposed protocol can deliver more packets to the destination in less time and with lower overhead than AODV because MRAA uses alternate routes for data delivery in case working routes are broken. As the number of alternate paths increases, the loss ratio of data packets slightly decreases. Also the performance of MRAA is superior in terms of the end-to-end delay and throughput with comparison with the network load in most cases that is due to the routing decisions based on more intelligent decision of pheromone value that is increased for more promising routes and decreased for less promising routes. In addition, we can observer that MRAA is superior in terms of both the generated routing overhead and the processing overhead as less routing traffic is generated because of less need to broadcast unnecessary RREQ packets as ant agents when finds a path with pheromone lift, it doest not broadcast itself instead it takes the path of the previous ant agent and leave more pheromone as it returns back.

We can observe that the effectiveness of both protocols increases because of the decrease in packet mobility in comparison with other MANET Routing Protocols. but we can see from results in the tables that MRAA's performance and effectiveness does not degrade as much as AODV when compared to other routing protocols for example:

- MRAA vs OSLR : we can see that OSLR has a little bit lower delay and lower packet loss but in contrast it has a great routing overhead compared to MRAA and places more load on the network from routing overhead than MRAA.
- MRAA vs GRP : we can see that GRP has a little bit lower delay , lower packet loss , lower routing overhead and of less load on the network than MRAA that is because GRP is well suited for mobile nodes and use the GPS system to establish routes between the nodes.
- MRAA vs DSR : we can see that DSR has a little bit more delay , lower packet loss ,much lower routing overhead compared to MRAA but it has a much more lower throughput when compared to MRAA and all other routing techniques.

VI. Conclusions

In this paper we proposed an alternative ad-hoc routing protocol to use the well-known ant colony optimization technique and also to find multiple disjoint routes between a source node and a destination. We compared the performance of the proposed protocol with that of the AODV routing protocol in terms of end-to-end delay, Throughput, Network Load, sent routing traffic, received routing traffic, number of dropped data packets, simulation duration for both cases when the nodes are stationary and when they are moving. In the conventional AODV, a source node performs a route discovery procedure whenever an existing route is disconnected. In the proposed protocol, however, a source node can send data packets to its corresponding destination through one of backup routes pre-established. Our simulation results show that the proposed protocol yields better performance than the conventional AODV protocol. In addition, it was able to achieve results near to the results of other more sophisticated protocols such as (OSLR, DSR, GRP) and even outperform them in some performance metrics.

References

[1] J. Broch, D. A. Maltz, D. B. Johnson, Y-C. Hu, and J. Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. In Proceedings of the 4th International Conference on Mobile Computing and Networking (ACM MOBICOM98), pages 8597, October 1998.

[2] J. Broch, D. B. Johnson, and D. A. Maltz, The dynamic source routing protocol for mobile ad hoc networks, Internet draft, Dec. 1998.

[3] M. Dorigo and G. Di Caro, The ant colony optimization meta-heuristic. In D. Corne, M. Dorigo, and F. Glover, editors, New Ideas in Optimization, pages 1132. McGraw-Hill, London, 1999.

[4] G. Di Caro, F. Ducatelle and L. Maria Gambardella, AntHoc-Net: An Adaptive Nature-Inspired Algorithm for Routing in Mobile Ad Hoc Networks,in: Technical Report No. IDSIA-27-04-2004 September 2004.

[5] M. Gunes, U. Sorges, I. Bouazizi, ARA The Ant-Colony Based Routing Algorithm for MANETs, in: InternationalWorkshop on Ad Hoc Networking (IWAHN 2002), Vancouver, British Columbia, Canada, August 18-21, 2002.

[6] P. Johansson, T. Larsson, N. Hedman, B. Mielczarek and M. Degermark, Scenario-based performance analysis of routing protocols for mobile ad-hoc networks, in: Proc. of MOBICOM99, Seattle, WA (August 1999).

[7] S. Mueller, R. P. Tsang, and D. Ghosal, Multipath Routing in Mobile Ad Hoc Networks: Issues and Challenges.

[8] C. E. Perkins, E. M. Belding-Royer, and S. R. Das, Ad hoc On-Demand Distance Vector (AODV) Routing, http://www.ietf.org/rfc/rfc3561.txt.

[9] C. Perkins and E. M. Royer, ad hoc On demand distance vector (AODV) routing (Internet draft), August 1998.

[10] C. Perkins and P. Bhagwat, Highly dynamic destination-sequenced distance-vector (DSDV) routing for mobile computers, in ACM SIGCOMM Symposium on Communications, Architectures and Protocols, September 1994, pp. 234244.

[11] OPNET Simulator, http://www.opnet.com/.

[12] E. M. Royer and C.-K. Toh, A Review of current routing protocols for ad hoc mobile wireless networks, IEEE Personal Communications, Apr, 4655, 1999.