New Adaptive Routing Protocol for Mobile Ad hoc Networks

Tarek M Mahmoud
Dept. of Computer Science
Faculty of Science
Minia University
El Minia - Egypt
d.tarek@mu.edu.eg

Essam H Houssein
Dept. of Computer Science
Faculty of Computers and Informatics
Benha University
Benha - Egypt
esam.halim@fci.bu.edu.eg

Alaa A K Ismaeel
Dept. of Computer Science
Faculty of Science
Minia University
El Minia - Egypt
alaa.ismaeel@mu.edu.eg

Abstract- An ad hoc mobile network is a collection of mobile nodes that are dynamically and arbitrarily located in such a manner that the interconnections between nodes are capable of changing on a continual basis. Routing in MANET is extremely challenging because of MANETs dynamic features, its limited bandwidth and power energy. The routing protocol is used to discover routes between nodes. Routing is a challenging task in ad hoc network due to mobility of nodes that frequently changes network topology. Nature-inspired algorithms (swarm intelligence) such as ant colony optimization (ACO) algorithms have shown to be a good technique for developing routing algorithms for MANETs. In this paper, we propose a new routing algorithm for MANETs called Ant-HRP which based on ACO, proactive and reactive routing protocol capability and is simulated on NS2. Results indicate that Ant-HRP effectively improve the connectivity, packet delivery ratio and reduce the end-to-end delay as compared with the AntNet, AODV and DSDV routing protocols.

Keywords: *MANETs, AntNet, ACO, AODV, Ant-HRP and WSWAN.*

1. INTRODUCTION

A mobile ad hoc networks (MANETs) consisting of a collection of mobile nodes (MNs) sharing a wireless channel without any centralized control or established communication backbone. MANETs have no fixed routers; all nodes are capable of movement and can be connected dynamically in an arbitrary manner. Usually, these nodes act as both end systems and routers at the same time. Nodes of these networks, which function as routers, discover and maintain routes to other nodes in the network. The topology of the MANET depends on the transmission power of the nodes and the location of the MNs, which may change with time. A working group namely "MANET" has been formed by the Internet Engineering Task Force (IETF) to study the related issues and stimulated research in MANETs [28].

A fundamental problem in MANET is how to deliver packets among MNs efficiently without predetermined topology or centralized control, which is the main objective of routing protocols. Since MANETs change their topology frequently, routing in such networks is a challenging task. So far, much work has been done on routing in MANETs and can be divided into: proactive protocols and reactive protocols [11].

Proactive routing protocol includes: Destination Sequenced Distance Vector (DSDV) and Fisheye State Routing (FSR) etc. They attempt to maintain a correct view of the network topology add the time and build routes from each node to every other node before they are needed, hence they are also called table-driven protocols. Any changes in topology are propagated through the network, so that all nodes know of the changes in topology. Thereby, proactive protocols maintain routing information about the available paths in the network even if these paths are not currently used. The major drawback of these approaches is that the maintenance of unused paths may occupy an important part of the available bandwidth if the topology of the network changes frequently [28].

Reactive routing protocols includes: Ad hoc On-demand Distance Vector (AODV) Routing [3] and Dynamic Source Routing Protocol (DSR) [3] etc. Reactive routing protocols maintain only the routes that are currently in use, thereby trying to maintain low control overhead, reducing the load on the network when only a small subset of all available routes is in use at any time. However, they still have some inherent limitations. First, since routes are only maintained while in use, it is usually required to perform a route discovery before packets can be exchanged between communication peers. This leads to a delay for the first packet to be transmitted. Second, even though route maintenance for reactive algorithms is restricted to the routes currently in use, it may still generate an important amount of network traffic when the topology of the network changes frequently. Finally, packets to the destination are likely to be lost if the route to the destination changes.

Several performance studies [1, 5, 10, 16, 18, 24 and 27] of MANETs have built and utilize only one single route for each pair of source and destination nodes. Due to node mobility, node failures, and the dynamic characteristics of the radio channel, links in a route may become temporarily unavailable, making the route invalid. The overhead of finding alternative routes may be high and extra delay in packet delivery may be introduced. Multipath routing addresses this problem by providing more than one route to a destination node. Source and intermediate nodes can use these routes as primary and backup routes. High route discovery latency together with frequency route discovery attempts in dynamic networks can affect the performance adversely. Multipath protocols try to alleviate these problems by computing multiple paths in a single route discovery attempt. Multiple paths could be formed at both traffic sources as well as at intermediate nodes. New route discovery is needed only when all paths fail. This reduces both route discovery latency and routing overhead. Multiple paths can also be used to balance load by forwarding data packets on multiple paths at the same time, though we will investigate this aspect in our work.

In general, reactive protocols are more efficient than proactive routing protocols in terms of control overhead and power consumption since routes are only established when required. By contrast, proactive protocols require periodic route updates to keep information current and consistent. In addition, many routes maintained might never be needed, which significantly adds to routing overhead in the bandwidth-constrained network. As routing overhead grows exponentially with network size, it prevents the application of these protocols in large-scaled networks. Proactive protocols generally provide better quality of service than reactive protocols. As in proactive protocols, routing information is constantly updated, routes to every destination are always available and up-to-date, and, hence, end-to-end delay can be minimized. For reactive protocols, the source node has to wait for the route to be discovered before communication can happen. This latency in route discovery might be intolerable for real-time communications. However, we will investigate this aspect in our work. In [18], the authors present a set of tables that summarize the main differences between these protocols in terms of their complexity, route update patterns, and capabilities.

Recently, there is an increasing interest in the use of swarm intelligence (SI) [6, 8, 14 and 19] or nature inspired algorithms for routing in MANET. SI is a computational intelligence technique that involves collective behavior of autonomous agents that locally interact with each other in a distributed environment to solve a given problem in the hope of finding a global solution to the problem. Ant colonies, bird flocking, animal herding and fish schooling are examples in nature that use swarm intelligence. The foraging behavior of ants [8], bees [12] and the hill building behavior of termites [21] has inspired researchers in developing an efficient routing algorithm for MANETs. There are lots of similarities between MANETs and ants. MANET environment is unstructured, dynamic and distributed like the ants environment. In MANETs, the route request packet interact with each node locally to get routing information similar to ants that use pheromones to get local information. The traditional protocols for MANETs and ant based algorithms provide multiple paths. They are both self configuring and self organizing systems. The foraging behavior of ants and the interaction behavior of MANETs to deliver packets from source to destination are similar. The goal for both systems is to find the shortest path.

ACO [13 and 25] is based on the behavior of a group of artificial ants in search of a shortest path from the source to the destination. These artificial ants mimic real ants in nature in search of food from the nest to the destination. The ants deposit a chemical substance called pheromone that other ants can sense on their journey to the destination. The ants interact with each other and the environment using the pheromone concentration. As with any perfume, if not reapplied, the scent evaporates. As the ants travel, the longer paths lose their pheromone concentration making the ants to choose the shortest path.

ACO has been applied to many network optimization problems, ant based routing has been applied to static telecommunication network. Existing ant based routing protocols for MANET [7, 9, 17, 20, 22, 23 and 26] is very promising when compared to conventional routing algorithms. They are more efficient, more robust and are able to discover multiple paths. However, all of these protocols have scalability and control packet overhead problems due to the fact that each node has to keep in its routing table, the pheromone amount from all its neighbors to all other nodes in the network or to desired destinations. If the number of nodes in the network is small, the table size is not a concern. However, when the network size grows, the routing table size of each node increases dramatically, which not only consumes a large portion of the mobile nodes' memory, but also costs a lot of computation power to retrieve, modify or insert a new record in the routing table. Consequently, end to end delay becomes large. In the literature, AntHocNet [9], based on ACO, has been proposed as a hybrid routing algorithm for MANETs this algorithm reactively finds a path between source and destination reactively and proactively maintains the existing routes. When a node desires a route to a destination, the reactive part of the algorithm is initiated. The proactive part of the algorithm serves to maintain and reinforce the existing routes by updating the routing table. This is a large overhead of network resources in AntHocNet. In our approach, the proactive component is limited to create the network paths at the first time (predefined) and proactively maintains the existing routes using mobile ant. This dramatically reduces the control overhead packets and increases the packet delivery ration when compared with DSDV. Furthermore, the route maintenance is expensive with the ants broadcasting update information over the network. In our approach, ants will not flood the entire network these ants flood to maintain the broken routs only (like reactive behavior). Also, link failure in AntHocNet involves broadcasting the entire network which again is another overhead. In our approach, the handling of link failures processes reactively. If an alternative path can be found, the buffered packets will be sent to the destination and will inform the source of the alternative route. Otherwise, an error message will be sent to the source.

ACO can be used for efficient routing in a network and discover the topology, to provide high connectivity at the nodes. The nodes depend solely on the ant agents to provide them routes to various destinations in the network. This may not perform well when the network topology is very dynamic and the route lifetime is small. In ACO nodes have to wait to start a communication, till the ants provide them with routes. In some situations it may also happen that the nodes carrying ants suddenly get disconnected with the rest of the network. This may be due to their movement away from all other nodes in the network or they might go into sleep mode or simply turned off. In such situations, the amount of ants left for routing are reduced in the network which leads to ineffective routing.

This paper tries to overcome these shortcomings of ACO and proactive routing protocol by combining them to develop a hybrid routing scheme called HOPPRP this algorithm proactively creates routes and reactively maintenance of used paths only. The HOPPRP is able to reduce the overhead, the delay and packet dropped by providing high connectivity also, increase the packet delivery ratio as compared to DSDV and AODV routing protocols.

The rest of this paper is organized as follows. In Section 2 provides an overview of the related work. Proposed routing protocol is introduced in Section 3. Section 4 shows the Simulation model and performance metrics. Simulation results and performance analysis is introduced in Section 5. Performance summary is demonstrated in Section 6. Finally Section 7 concludes the paper.

2. RELATED WORK

The IETF MANET Working Group has developed a number of protocols for MANETs.

2.1 Ad hoc On-demand Distance Vector (AODV)

The AODV routing protocol [3, 9, 18 and 27] is a reactive protocol. Like most reactive routing protocols, route finding is based on a route discovery cycle involving a broadcast network search and a unicast reply containing discovered paths. AODV protocol relies on per-node sequence numbers for loop freedom and for ensuring selection of the most recent routing path. AODV nodes maintain a route table in which next-hop routing information for destination nodes is stored. Each routing table entry has an associated lifetime value. If a route is not utilized within the lifetime period, the route is expired. Otherwise, each time the route is used, the lifetime period is updated so that the route is not prematurely deleted.

When a source node has data packets to send to some destination, it first checks its route table to determine whether it already has a route to the destination. If such a route exists, it can use that route for data packet transmissions. Otherwise, it must initiate a route discovery procedure to find a route. To start route discovery, the source node creates a route request (RREQ) packet. It places in this packet the destination node’s IP address, the last known sequence number for that destination, and the source’s IP address and current sequence number. The RREQ also contains a hop count, initialized to zero, and a RREQ ID. The RREQ ID is a per-node, monotonically increasing counter that is incremented each time the node initiates a new RREQ. In this way, the source IP address, together with the RREQ ID, uniquely identifies a RREQ and can be used to detect duplicates.

After creating this message, the source broadcasts the RREQ to its neighbors. When a neighboring node receives a RREQ, it first creates a reverse route to the source node. The node from which it received the RREQ is the next hop to the source node, and the hop count in the RREQ is incremented by one to get the hop distance from the source.

The node then checks whether it has an unexpired route to the destination. If it does not have a valid route to the destination, it simply rebroadcasts the RREQ, with the incremented hop count value, to its neighbors. In this manner, the RREQ floods the network in search of a route to the destination. When a node receives a RREQ, it checks whether it has an unexpired route to the destination. If it does have such a route, then one other condition must hold for the node to generate a reply message indicating the route. The node's route table entry for the destination must have a corresponding sequence number that is at least as great as the indicated destination sequence number in the route request. Once this condition is met, the node can create a route reply (RREP) message. The RREP contains the source node's IP address, the destination node's IP address, and the destination's sequence number as given by the node's route table entry for the destination. In addition, the hop count field in the RREP is set equal to the node's distance from the destination. If the destination itself is creating the RREP, the hop count is set equal to zero. After creating the reply, the node uncast the message to its next hop toward the source node. Thus, the reverse route that was created as the RREQ was forwarded is utilized to route the RREP back to the source node.

When the next hop receives the RREP, it first creates a forward route entry for the destination node. It uses the node from which it received the RREP as the next hop toward the destination. The hop count for that route is the hop count in the RREP, incremented by one. This forward route entry for the destination will be utilized if the source selects this path for data packet transmissions to the destination. Once the node creates the forward route entry, it forwards the RREP to the destination node. The RREP is thus forwarded hop by hop to the source node. Once the source receives the RREP, it can utilize the path for the transmission of data packets. If the source receives more than one RREP, it selects the route with the greatest sequence number and smallest hop count.

Once a route is established, it must be maintained as long as it is needed. A route that has been recently utilized for the transmission of data packets is called an active route. Because of the mobility of the nodes, links along paths are likely to break. Breaks on links that are not being utilized for the transmission of data packets do not require any repair; however, breaks in active routes must be quickly repaired so that data packets are not dropped. When a link break along an active path occurs, the node upstream of the break invalidates the routes to each of those destinations in its route table. It then creates a route error (RERR) message. In this message it lists all of the destinations that are now unreachable due to the loss of the link. After creating the RERR message, it sends this message to its upstream neighbors that were also utilizing the link. These nodes, in turn, invalidate the broken routes and send their own RERR messages to their upstream neighbors that were utilizing the link. The RERR message thus traverses the reverse path to the source node. Once the source node receives the RERR, it can repair the route if the route is still needed.

AODV contains a number of optimizations and optional features. To improve the protocol performance and reduce overhead, source nodes can utilize an expanding ring search to search for routes to the destination. The propagation of the RREQ is controlled by modifying the time to live (TTL) value of the packet. Incrementally larger areas of the network are searched until a route to the destination is discovered. If a route to the destination can be found in the local area, a network-wide flood can be avoided. Another optimization is the local repair of link breaks in active routes. When a link break occurs, instead of sending a RERR to the source, the node upstream of the break can try to repair the link locally itself. If successful, fewer data packets are dropped because the route is repaired more quickly. If the local repair attempt is unsuccessful, a RERR message is sent to the source node as previously described.

2.2 Destination-Sequenced Distance Vector (DSDV)

The Destination-Sequenced Distance Vector (DSDV) routing protocol [18, 27 and 28] is a distance vector protocol that implements a number of customizations to make its operation more suitable for MANETs. DSDV utilizes per-node sequence numbers to avoid the counting to infinity problem common in many distance vector protocols. Nodes increment its sequence number whenever there is a change in its local neighborhood (i.e., a link addition or removal). When given a choice between two routes to a destination, a node always selects the route with the greatest destination sequence number. This ensures utilization of the most recent information.

Because DSDV is a proactive protocol, each node maintains a route to every other node in the network. The routing table contains the following information for each entry: destination IP address, destination sequence number, next-hop IP address, hop count, and install time. DSDV utilizes both periodic and event-triggered routing table updates. Every time interval, each node broadcasts to its neighbors its current sequence number, along with any routing table updates. The routing table updates are of the form: < destination IP address, destination sequence number, hop count >, after receiving an update message, the neighboring nodes utilize this information to compute their routing table entries using an iterative distance vector approach. In addition to periodic updates, DSDV also utilizes event-triggered updates to announce important link changes, such as link removals. Such event-triggered updates ensure timely discovery of routing path changes.

As stated previously, if a node learns two distinct paths to a destination, the node selects the path with the greatest associated destination sequence number. This ensures the utilization of the most recent routing information for that destination. When given the choice between two paths with equal destination sequence numbers, the node selects the path with the shortest hop count. On the other hand, if all metrics are equivalent, then the choice between routes is arbitrary.

2.3 Ant based routing protocols Overview

Ant-based routing algorithm for MANETs has been previously explored by [2, 13 and 25]. AntNet and ABC [7 and 26] are among the earliest algorithms that use ant colony optimization methods to solve routing problems for wired network. Both are proactive algorithms. AntNet uses two ants, forward and backward ants to find the shortest route from the source to the destination. The forward ant searches for the route from the source to destination, while the backward ant changes the status of the links by increasing the pheromone concentration on the links along the path. ABC, which is also similar to AntNet, is considered to be more efficient since the algorithm uses less control overhead packets.

Ant-based routing algorithm for MANETs has been previously explored by [12]. Ants in network routing applications are simple agents embodying intelligence and moving around in the network from one node to the other, updating the routing tables of the nodes that they visit with what they have learned in their traversal as shown on Fig. 1.

Routing ants keep a history of the nodes previously visited by them. When an ant arrives at a node, it uses the information in its history to update the routing table at that node with the best routes that it has for the other nodes in the network. The higher the history size the larger the overhead, hence a careful decision on the history size of the ants has to be made. All the nodes in the network rely on the ants for providing them the routing information, as they themselves do not run any program (protocol) for finding routes. The ant-based routing algorithm implemented in this paper does not consider any kind of communication among the ants and each ant works independently. The population size of the ants is another important parameter, which affects the routing overhead.

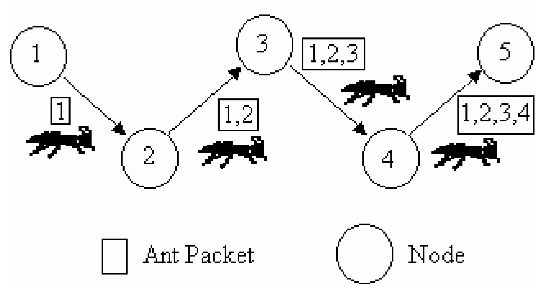

Fig. 1: ant traversing and providing routing information to nodes.

In the conventional ant algorithms the next hop is selected randomly. This is because if the next hop selected is the same as the previous node then this route would not be optimal. Data packets sent on such routes would just be visiting a node and going back to the previous node in order to reach the destination. Every node frequent broadcasts HELLO messages to its neighbors so that every node can maintain a neighbor list, which is used for selecting the next hop by the ants.

ARA [20] proposed by Gunes et al. is one of the first ACO algorithms for finding routes in MANETs. It is a reactive algorithm which consists of three phases: route discovery, route maintenance and route failure handling. Forward and backward ants are used in the route discovery phase. The route maintenance phase does not use any specialized ants, but rather uses data packets to maintain the route between the source and destination. If the source receives a route failure notification, the source re-initiates a route discovery phase. ARA paved the way for many more ant colony optimization algorithms for MANETs since it is not scalable and does not detect cycles.

ARAMA [23] is a proactive routing algorithm. The main task of the forward ant as in other ACO algorithms for MANETs is to collect path information. However, in ARAMA, the forward ant takes into account not only the hop count factor, as most protocols do, but also the links local heuristic along the route such as the node's battery power and queue delay. ARAMA defines a value called grade. This value is calculated by each backward ant, which is a function of the path information stored in the forward ant. At each node, the backward ant updates the pheromone amount of the node's routing table, using the grade value. The protocol uses the same grade to update pheromone value of all links. The authors claim that the route discovery and maintenance overheads are reduced by controlling the forward ant's generation rate. However, they do not clarify how to control the generation rate in a dynamic environment.

Islam et al. [22] propose an on demand routing algorithm called source update algorithm. The routing update scheme in this algorithm solves the all pair shortest path problem unlike other protocols which solves the single source shortest path problem. The algorithm uses an exploration technique instead of an exploitation technique and detects cycles. The disadvantage of this algorithm is that it does not scale well for large networks.

AntHocNet is a hybrid ant based routing protocol proposed by Di Caro [9] in the effort to combine the advantages from both AntNet and ARA. AntHocNet reactively finds a route to the destination on demand, and proactively maintains and improves the existing routes or explore better paths. In AntHocNet, ant maintains a list of nodes it has visited to detect cycles. The source node sends out forward ants and when it receives all the backward ants, one generation is completed. Each node *i* keeps the identity of the forward ants, the path computation, number of hops, of the ant from the source to node *i*, and the time the ant visited node *i*. Note that more than one ant may have reached node *i* and therefore the identity of the ant is important. When an ant arrives at a node, the node checks the ant's path computation and the time it reached node *i*. If the path computation and time are within a certain limit of those produced by another ant of the same generation then the ant is forwarded. Otherwise, the ant is discarded. In case of a link failure at a node and no alternative paths are available, the node sends a reactive forward ant to repair the route locally and to determine an alternative path. If a backward ant is received for the reactive forward ant, the data packets are sent along the newly found path and all its neighbors are notified about the change in route. Otherwise, the node sends a notification to all its neighbors of the lost destination paths which in turn initiate forward ants from the neighbors. In the literature, AntHocNet is considered a well known hybrid algorithm. However, route maintenance is expensive with the reactive forward ants flooding the network. Each node keeps a table with all possible destination routes which lead to scalability problems.

3. PROPOSED ROUTING PROTOCOL

Ant-HRP is a hybrid ant based routing protocol proposed in the effort to combine the advantages from both of AntNet, proactive and reactive routing protocol. Ant-HRP reactively finds a route to the destination on demand, and proactively maintains and improves the existing routes or explores better paths. In Ant-HRP, ant maintains a list of nodes it has visited to detect cycles. The source node sends out forward ants and when it receives all the backward ants, one generation is completed. Each node I keeps the identity of the forward ants, the path computation, number of hops, of the ant from the source to node I, and the time the ant visited node I. Note that more than one ant may have reached node I and therefore the identity of the ant is important. When an ant arrives at a node, the node checks the ant's path computation and the time it reached node I. If the path computation and time are within a certain limit of those produced by another ant of the same generation then the ant is forwarded. Otherwise, the ant is discarded. In case of a link failure at a node and no alternative paths are available, the node sends a reactive forward ant to repair the route locally and to determine an alternative path. If a backward ant is received for the reactive forward ant, the data packets are sent along the newly found path and all its neighbors are notified about the change in route. Otherwise, the node sends a notification to all its neighbors of the lost destination paths which in turn initiate forward ants from the neighbors.

To overcome some of the inherent drawbacks of ant-based, proactive and reactive routing protocols the proposed technique forms a hybrid of them. The hybrid routing protocol enhances the packet delivery ratio and the node connectivity, decreases the delay. In conventional ant-based routing techniques route establishment is dependant on the ants visiting the node and providing it with routes. If a node wishes to send data packets to a destination for which it does not have a fresh enough route, it will have to keep the data packets in its send buffer till an ant arrives and provides it with a route to that destination. Hence when a route breaks the source still keeps on sending data packets unaware of the link breakage. This leads to a large number of data packets being dropped. In conventional proactive routing protocol on the other hand is that the maintenance of unused paths may occupy an important part of the available bandwidth if the topology of the network changes frequently.

Ant-HRP adopts the behavior of a real ant colony to maintain a broken path efficiently and quickly. Ant-HRP consists of two processes, namely routing discovery and route maintenance. In the routing discovery process, a mobile node launches QUERY packets (forward ants) to find multiple paths to destination (food sources). If a destination is found, a REPLY packet (backward ant) is returned from the destination to the mobile node to set up the paths (lay pheromone). Destination packets are then routed stochastically according to pheromone intensity once the destination routing paths are established. In the routing maintenance process periodically sends EXPLORE message to destinations to monitor the quality of existing routes, and to explore new routes to destinations.

Ant-HRP utilizes ants working independently and providing routes to the nodes. Once a route is established, it must be maintained as long as it is needed. Because of the mobility of the nodes, links along paths are likely to break. However, breaks in routes must be quickly repaired so that data packets are not dropped. When the link break, the node upstream sends EXPLORE message. In this message it lists all of the destinations that are now unreachable due to the loss of the link. After creating the EXPLORE message, it sends this message to its upstream neighbors that were also utilizing the link. These nodes, in turn, invalidate the broken routes and send their own EXPLORE messages to their upstream neighbors that were utilizing the link. The EXPLORE message thus traverses the reverse path to the source node. Once the source node receives the EXPLORE, it can repair the route if the route is still needed.

Local connectivity in Ant-HRP is maintained by using EXPLORE message. The routing table in Ant-HRP routing scheme is common to both ants and proactive routing protocol. Frequent HELLO broadcasts are used to maintain the neighbor table like the technique used in reactive protocols. This table is used to select a randomly chosen next hop from the list of neighbors by the ant.

The use of ants with proactive routing protocol increases the node connectivity, which in turn reduces the maintenance of unused paths. Lastly, as ant agents update the routes continuously, a source node can switch from a longer (and stale) route to a newer and shorter route provided by the ants. This leads to a considerable decrease in the end-to-end delay as compared to traditional routing protocols.

4. SIMULATION MODEL AND METRICS

4.1 Simulation Model

The Ant-HRP protocol proposed in this paper is compared with the AODV, DSDV and AntNet routing protocols. Network Simulator (NS-2) [11] is a discrete event simulator used to simulate these protocols which can model and simulate multi-hop MANETs was used for the simulations. The physical layer for the simulation uses two-ray ground reflection as the radio propagation model. The link layer is implemented using IEEE 802.11 Distributed Coordination Function (DCF), Media Access Control protocol (MAC). Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) is used to transmit these packets. Packets sent by routing layer are queued at the interface queue till MAC layer can transmit them. Simulations were run for 100 simulated seconds. Our simulation models the network with variable size 10, 20, 40, 50 and 100 nodes migrating within an area of 1500x1000m^2 with a speed of 20 m/s. Radio propagation range for each node is 250 m and channel capacity is 2 Mbps. Pause time can be defined as time for which nodes waits on a destination before moving to other destination. The pause time is 0 s which means the node is always moving in the entire simulation period. The simulations have been carried using the parameters mentioned in Table 1.

TABLE 1: SCENARIO FOR NS-2 TOPOLOGY

Parameter	Value
Number of simulated nodes	10-20-40-50-100
Area size of topography	1500x1000m2
Wireless range	150 m
Packet size	512 byte
Send rate of traffic	1 packets / s
Traffic type	CBR
Number of traffic sources	5-10-20-25-50
Speed Mobility Model	RWP
Pause Time (s) at simulation	0s
Simulation Time	100 s
Simulated Routing Protocols	Ant-HRP, AntNet, AODV and DSDV

4.2 Mobility Model

A mobility model should attempt to mimic the movements of real Mobile Networks. Changes in speed and direction must occur and they must occur in reasonable time slots. There is several mobility models supported. The mobility model uses the random waypoint (RWP) model in the rectangular field. Rectangular area is used to force the nodes to create long routes and this help in studying the effect of the proposed modifications. In this model, at every instant, each node is randomly placed in the simulated area and remains stationary for a specified pause time. It then randomly chooses a destination and moves there at a velocity chosen uniformly between a minimum velocity and a maximum velocity. Each node independently repeats this movement pattern through the simulation.

- Effect of Unvarying Pause Time: Pause time can be defined as time for which nodes waits on a destination before moving to other destination. We used a constant pause time as a parameter as it is measure of mobility of nodes. Low pause time means node will wait for less time thus giving rise to high mobility scenario.
- Effect of Varying Number of Nodes: Number of nodes plays important role in performance. Our simulations show various performance parameters versus number of nodes, we tested the considered routing protocols by varying the number of nodes to account for system scalability.

4.3 Traffic Model

A traffic generator named Cbrgen [15] was developed to simulate constant bit rate sources in NS-2, act as the important parameter of our simulation to compare the performance of each routing protocol; we chose our traffic sources to be constant bit rate (CBR). When defining the parameters of the communication model the number of source/destination pairs and the packet sending rate in each pair is varied to change the offered load in the network. We can use shell command Cbrgen to generate 5 pair of, 10 pair of, 20 pair of, 25 pair of and 50 pair of UDP stream stochastically thus, the network connectivity is 0.5. Each CBR package size is 512 bytes and one second transmits one package which used varying the number of CBR source was approximately equivalent to varying the sending rate. We have chosen this value because smaller payload sizes penalize protocols that append source routes to each data packet.

4.4. Performance Metrics

In this subsection, we present performance metrics that have been proposed for the performance evaluation of MANET routing protocols. The following metrics are applied to comparing the routing protocols performance. Some of these metrics are suggested by the MANET working group for routing protocol evaluation [4].

- Packet Delivery Ratio: The ratio between the number of data packets originated by the CBR sources and the number of data packets received by the CBR sink at the final destination.
- End to End Delay: This includes all possible delays caused by buffering during routing discovery latency, queuing at the interface queue, and retransmission delays at the MAC, propagation and transfer times.
- Routing Packet Overhead: The total number of transmissions routing packets transmitted during the simulation. For packets sent over multiple hops, each transmission of the packet counts as one transmission.
- Connectivity: is the average number of nodes in the network for which a node has un-expired routes.

5. SIMULATION RESULTS AND PERFORMANCE ANALYSIS

This section presents an attempt was to compare all of the considered routing protocols under the same simulation environment, we will try to discuss the behavior of the considered routing protocols depend on a constant pausing time and variable network size.

5.1 Packet Delivery Ratio

Figure 2 shows the packet delivery ratio for Ant-HRP, AODV, DSDV and AntNet routing protocols. There are two reasons an intermediate node will not be able to deliver packets: the pheromone concentration along the neighboring links is zero or the node has moved away. In the latter case, the upstream node of the broken link will conduct a local repair procedure, trying to find an alternative path to the destination while buffering all the packets it receives for Ant-HRP and AODV. If the node successfully finds a new path to the destination, it will send all the buffered packets to the destination via the newly found route, meanwhile, a notification ant will be sent to the source to let the source node know the change of route. In the former case, the ants cannot select any links to travel if all their links, upstream and downstream are zero and the data packet is dropped at that node, hence higher delivery ratio in AODV than other routing protocols. Note that as the network size increases and with more neighbors for a node, the delivery ratio for Ant-HRP is better than AODV. This is because the ants can choose from multiple paths rather than a single path as in AODV. The reason for high packet delivery ratio is that they make use of link failure detection and route error messages. Whereas in case of DSDV there is no such feature and so the source nodes keep on sending packets unaware of the link failures. This leads to a large amount of data packets being dropped which reduces the packet delivery ratio.

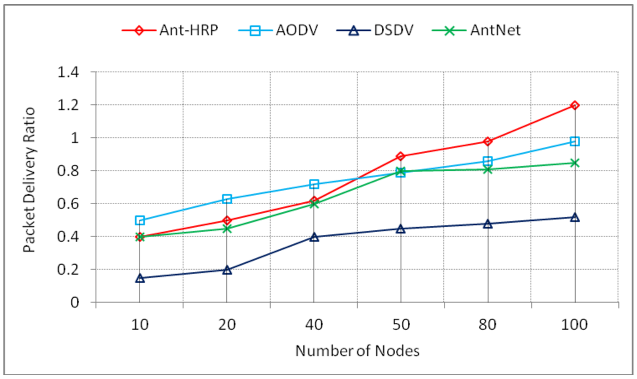

Fig. 2: Packet Delivery Ratio vs. Number of Nodes.

5.2 End to End Delay

Figure 3 shows the Ant-HRP produces better end to end delay results than compared routing protocols.

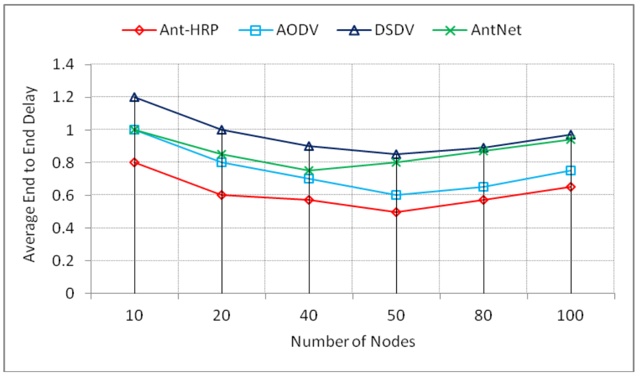

Fig. 3: End to End Delay vs. Number of Nodes.

Comparing Ant-HRP with AODV, DSDV and AntNet routing protocols it can be observed that the end-to-end delay is considerably reduced in Ant-HRP. In Ant-HRP, ants help in maintaining high connectivity hence the packets need not wait in the send buffer till the routes are discovered. Even if the source node does not have a ready route to the destination, due to the increased connectivity at all the nodes the probability of its receiving replies quickly from nearby nodes is high resulting in reduced route discovery latency. Lastly, the dynamic natures in which routes are kept updated by the ants leads to the source node switching from a longer route to shorter ones hence reducing end-to-end delay. For the AODV, the source node has to wait for the route to be discovered before communication can happen. But for the Ant-HRP, routing information is constantly updated, routes to every destination are always available and up-to-date, and, hence, end-to-end delay can be minimized.

5.3 Routing Packet Overhead

Figure 4 shows the routing packet overhead of Ant-HRP in comparison to AODV, DSDV and AntNet routing protocols. The control packets are periodically sent out to maintain the routes. This is a major contributing factor to the overhead in Ant-HRP. As the network size increases with more neighbors for a node, the node has more choices for paths to destination and therefore, the routes between nodes have multiple paths.

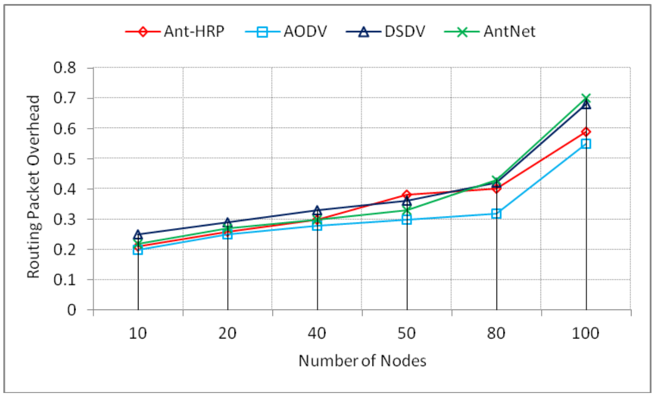

Fig. 4: Routing Packet Overhead vs. Number of Nodes.

The total routing overhead in case of Ant-HRP is independent of the traffic. Even if there is no communication the ants would still is traversing the network and update the routing tables this behavior increase the control message overhead. However in case of AODV, the overhead is dependent on the traffic and if there is no communication then there will be no control messages generated in the network. In Ant-HRP the overhead has two components: (i) the ants traversing in the network, (ii) each node maintains a route to every other node in the network and update routing tables. Every time interval, each node broadcasts to its neighbor after receiving an update message, the neighboring nodes utilize this information to compute their routing table entries using an iterative distance vector approach. From the comparison results it is seen that the overhead is too high in case of Ant-HRP because of the continuous movement of ants in the network and each node maintains a route to every other node in the network (multipath route). The continuous drop in routing packet overhead for all the three routing protocols is attributed to the increased packet delivery ratio.

5.4 Connectivity

Figure 5 shows the connectivity of Ant-HRP in comparison to AODV, DSDV and AntNet routing protocols.

In case of Ant-HRP agents continuously traverse the network and update the routing table entries. Due to this, a node has fresh enough routes to a large number of nodes in the network at any given point of time. The connectivity in Ant-HRP is more than DSDV and AODV routing protocols. Higher connectivity leads to lesser route discoveries and reduced end-to-end delay.

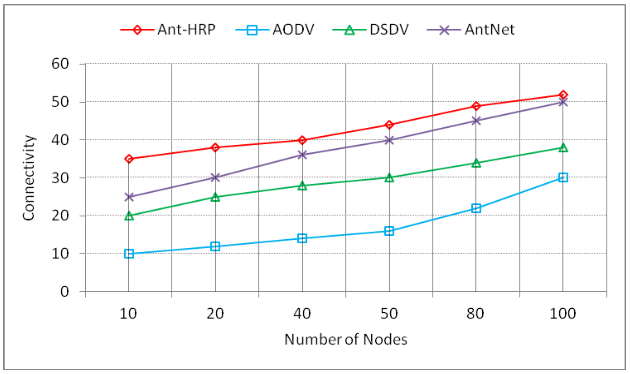

Fig. 5: Connectivity vs. Number of Nodes.

6. PERFORMANCE SUMMARY

An important characteristic of ant agents for routing in MANETs was observed during the simulations. After a certain period, the ant activity would almost subside. This could be due to various reasons such as: (i) the ant packets could be lost in wireless transmission, (ii) the next node which was to receive the ant packet moves out of the wireless range of the sending node, or (iii) the ant bearing node goes out of wireless range of every node in the network and there is no next hop node available for the ant. In such situations the number of ants actually available for routing purpose decreases.

To overcome this decrease in number of ants available for routing, a "minimum ant visit period" was set. If no ant visited a node within this period the node would generate a new ant and transmit it to one of its neighbors selected randomly. This way the ant activity would never subside and the network would not become devoid of ants. The simulations carried out used a minimum ant visit period of 5 seconds. The reduction in end-to-end delay and higher connectivity are achieved at the cost of extra processing of the ant messages and the slightly higher overhead occupying some network capacity. This however does not adversely affect the packet delivery fraction

Finally, Table 2 summarizes the performance evaluation of the considered routing protocols mentioned in this paper. It provides the performance matrices and the protocol name, where G, M and W mean Good, Medium and Worst performance respectively. As can been seen, from the above results, it is observed that, the proposed routing protocol Ant-HRP outperforms over the AODV, DSDV and AntNet routing protocols.

TABLE 2, SUMMARY OF PERFORMANCE RESULTS

Performance Matrices	**Ant-HRP**	**AODV**	**DSDV**	**AntNet**
Packet delivery ratio	M	G	W	M
End to end data delay	G	W	M	M
Routing packet overhead	W	G	M	M
Connectivity	G	W	M	M

7. CONCLUSION

This paper tries to overcome the shortcomings of the proactive, reactive and ACO by combining them to enhance their capabilities and alleviate their weaknesses. Ant-HRP hybrid protocol is able to provide reduced delay and high connectivity as compared to proposed protocols. As a result of increased connectivity the number of packets dropped is reduced. This makes Ant-HRP hybrid routing protocol suitable for real time data and multimedia communication.

References

[1] A. A. A. Radwan, T. M. Mahmoud and E. H. Houssein, "Evaluation Comparison of Some Ad Hoc Networks Routing Protocols", Elsevier, Egyptian Informatics Journal (EIJ), Vol. 1(2), Jul 2011.

[2] A. A. A. Radwan, T. M. Mahmoud and E. H. Houssein, "AntNet-RSLR: A Proposed Ant Routing Protocol for MANETs", The first Saudi International Electronics, Communications and Electronics Conference (SIECPC'11) and IEEE Explore Digital Library, Riyadh, Saudi Arabia, http://www.kacst-siecpc.org, pp. 1-6, Apr 2011.

[3] C.E. Perkins, E.M. Belding Royerand and I. D. Chakeres, "Ad Hoc On-Demand Distance Vector (AODV) Routing", draft-perkins-manet-aodvbis-01.txt, Jan., 2004.

[4] L. Layuan, L. Chunlin and Y. Peiyan, "Performance evaluation and simulations of routing protocols in ad hoc networks ", El Sevier, Computer Communications 30, pp: 1890–1898, 2007.

[5] D. Lang, "On the Evaluation and Classification of Routing Protocols for Mobile Ad Hoc Networks ", PhD thesis, Fakultät für Informatik der Technischen, Universität München, Germany, 2006.

[6] F. Ducatelle, "Adaptive Routing in Ad Hoc Wireless Multi-hop Networks", PhD thesis, Faculty of Informatics, Università della Svizzera, italiana, May, 2007.

[7] G. Di Caro and M. Dorigo, "AntNet: distributed stigmergetic control for communication networks", Journal of Artificial Intelligence Research 9, pp: 317–365, 1998.

[8] G. Di Caro, "Ant Colony Optimization and its Application to Adaptive Routing in Telecommunication Networks", PhD thesis, Faculté Des Sciences Appliqué Es, Université Libre De Bruxelles, Bruxelles, 2004.

[9] G. Di Caro, F. Ducatelle and L.M. Gambardella, "AntHocNet: an adaptive nature inspired algorithm for routing in mobile ad hoc networks", European Transactions on Telecommunications 16 (2), 2005.

[10] H. Duyen Trung, W. Benjapolakul and P. Minh Duc, "Performance evaluation and comparison of different ad hoc routing protocols", El Sevier, Computer Communications 30, 2478–2496, 2007.

[11] H. D. Trung and W. Benjapolakul, "Routing protocols in mobile ad hoc networks", Encyclopedia of Wireless and Mobile Communications", CRC Press, Book Chapter.

[12] 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", Proceedings of Genetic and Evolutionary Computation Conference, Washington, DC, pp: 153-160, 2005.

[13] H. Matsuo and K. Mori, "Accelerated Ants Routing in Dynamic Networks", Second Intl. Conf. On Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, 2001.

[14] H. Zhang and A. Arora, "Gs3: scalable, self configuration and self healing in wireless sensor networks", El Sevier, Computer Networks 43, pp: 459–480, 2003.

[15] K. Fall and K. Varadhan, "The ns Manual", VINT Project, UC Berkley, http://www.isi.edu.nsnam/ns, Dec., 2005.

[16] L. Layuan, L. Chunlin and Y. Peiyan, "Performance evaluation and simulations of routing protocols in ad hoc networks ", El Sevier, Computer Communications 30, pp: 1890–1898, 2007.

[17] L. Rosati, M. Berioli and G. Reali, "On ant routing algorithms in ad hoc networks with critical connectivity ", El Sevier, Ad Hoc Networks 6, pp: 827–859, 2008.

[18] M. Abolhasan, T. Wysocki and E. Dutkiewicz," A review of routing protocols for mobile ad hoc networks ", El Sevier, Ad Hoc Networks 2, pp: 1-22, 2004.

[19] M. Dorigo, E. Bonabeau and G. Theraulaz, "Swarm Intelligence: From Natural to Artifical Systems", Oxford University Press, New York, NY, 1999.

[20] M. Gunes, U. Sorges and I. Bouazzi, "ARA – the ant-colony based routing algorithm for MANETs", Proceedings of the International Conference on Parallel Processing Workshops (ICPPW'02), Vancouver, BC, pp: 79–85, 2002.

[21] M. Roth and S. Wicker, "Termite: ad-Hoc networking with stigmergy", Proceedings of IEEE Global Telecommunications Conference, vol. 5, San Francisco, USA, pp: 2937–2941, 2003.

[22] M. T. Islam, P. Thulasiraman and R.K. Thulasiram, "Implementation of ant colony optimization algorithm for mobile ad hoc network applications: openmp experiences", Parallel and Distributed Computing Practices 2 (5), pp: 177–191, 2004.

[23] O. Hossein and T. Saadawi, "Ant routing algorithm for mobile ad hoc networks (arama)", Proceedings of the 22nd IEEE International Performance, Computing, and Communications Conference, Phoenix, Arizona, USA, pp: 281–290, 2003.

[24] R. Kumar, V. Joshi and V. Dhir, "Performance Comparison of Routing Protocols in Mobile Ad Hoc Networks", International Journal of Engineering Science and Technology", Vol. 2(8), pp: 3494-3502, 2010.

[25] R. R. Choudhary, S. Bhandhopadhyay and K. Paul, "A Distributed Mechanism for topology discovery in Ad Hoc Wireless Networks Using Mobile Agents", Proc. of First Annual Workshop on Mobile Ad Hoc Networking Computing (MobiHOC'2000), 2000.

[26] R. Schoonderwoerd, O.E. Holland, J.L. Bruten and L.J.M. Rothkrantz "Ant based load balancing in telecommunications networks", Journal of Adaptive Behavior 5 (2), pp: 169–207, 1996.

[27] S. A. Ade and P.A.Tijare, "Performance Comparison of AODV, DSDV, OLSR and DSR Routing Protocols in Mobile Ad Hoc Networks ", International Journal of Information Technology and Knowledge Management, Vol. 2, No. 2, pp: 545-548, 2010.

[28] S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic, "Mobile Ad Hoc Networking", A JOHN WILEY & SONS, Publication, 2004.

Tarek M. Mahmoud received his Ph.D. degree in engineering (computer science) from Bremen University (Germany) in 1997. Since 1997, he has been with the Department of computer science of Minia University. Currently, he is an associate professor there. His research interests include wired and wireless networks, pattern recognition, combinatorial optimization, metaheuristics algorithms, web and semantic mining.

Essam H. Houssein received his Ph.D. degree in "Studying Some Routing Problems of Mobile Ad Hoc Networks using Artificial Ant Colony", from Dept. of CS, Faculty of Science, Minia University, Jan 2012. Currently work as Lecturer in CS Dept., Faculty of Computers and Informatics, Benha University. His research interests include wired and wireless networks, pattern recognition, combinatorial optimization, metaheuristics (Genetic, Simulated Annealing,…). http://www.bu.edu.eg/staff/esamhalim14