AN EMERGING ANT COLONY OPTIMIZATION ROUTING ALGORITHM (ACORA) FOR MANETS

M. Subha,
Lecturer,
Dept. of MCA,
Vellalar College For Women, Erode-12,
Email: subha_sangeeth@rediffmail.com

Dr. R. Anitha,
Director,
Dept. of MCA,
K.S.Rangasamy College of Technology,
Tiruchengode,
Email:aniraniraj@rediffmail.com

Abstract - Ad hoc networks are characterized by multi-hop wireless connectivity, frequently changing network topology and the need for efficient dynamic routing protocols. A mobile ad hoc network (Manet) is a collection of mobile nodes which communicate over radio. These kinds of networks are very flexible, thus they do not require any existing infrastructure or central administration. Therefore, mobile ad hoc networks are suitable for temporary communication links. The biggest challenge in this kind of networks is to find a path between the communication end points, what is aggravated through the node mobility.

In this paper we present a new ad-hoc routing algorithm ACORA, which is based on swarm intelligence. We refer to the protocol as the Ant Colony Optimization Routing Algorithm (ACORA). Ant colony algorithms are a subset of swarm intelligence and consider the ability of simple ants to solve complex problems by cooperation. Several algorithms which are based on ant colony problems were introduced in recent years to solve different problems, e.g. optimization problems. We aim to show that the approach has the potential to become an appropriate algorithm for mobile multi-hop ad-hoc networks, which are based on simulations made with the implementation in ns-2.

Keywords - Manet, Swarm Intelligence, ACO, ACORA

INTRODUCTION

An ad-hoc network consists of a set of nodes that communicate using a wireless medium over single or multiple hops and do not need any preexisting infrastructure such as access points or base stations. Therefore, mobile ad-hoc networks are suitable for temporary communication links. The biggest challenge in this kind of networks is to find a path between the communication end points, what is aggravated through the node mobility.

The routing scheme in a MANET can be classified into two major categories-Proactive and Reactive. The proactive or table driven routing protocols (DSDV) maintain routes between all node pairs all the time. It uses periodic broadcast advertisements to keep routing table up-to-date. This approach suffers from problems like increased overhead, reduced scalability and lack of flexibility to respond to dynamic changes. The reactive or on-demand (DSR, AODV) approach is event driven and the routing information is exchanged only when the demand arises. The route discovery is initiated by the source. Hybrid approaches combines the features of both the approaches [2].

In this paper we present a new routing algorithm ACORA for mobile, multi-hop ad-hoc networks to improve the performance of the existing protocol of mobile ad hoc network. The protocol is based on swarm intelligence and especially on the ant colony based metaheuristic. The proposed algorithm is implemented in ns-2 [10, 11 12 and 13].

Basics of Swarm Intelligence Systems

The emergent behavior of self-organization in a group of social insects is known as swarm intelligence. There are two popular swarm-inspired methods in computational intelligence areas: Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO) which is out off focus in this paper. ACO was inspired by the behavior of ants [1, 3, 4, and 5].

The basic idea of the ant colony optimization metaheuristic 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*1, 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.

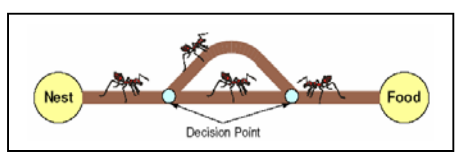

Fig.1 All ants take the shortest path after an initial searching time

Figure 1 shows a scenario with two routes from the nest to the food place. At the intersection, the first ants randomly select the next branch. Since the below route is shorter than the upper one, the ants which take this path will reach the food place first. On their way back to the nest, the ants again have to select a path. After a short time the pheromone concentration on the shorter path will be higher than on the longer path, because the ants using the shorter path will increase the pheromone concentration faster. The shortest path will thus be identified and eventually all ants will only use this one.

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. Also ACO mainly suits for ad hoc networks due to link quality, local work and support for multipath [4 and 5].

Why ACO suits to ad hoc networks ?

We discuss various reasons by considering important properties of mobile ad hoc networks.

Dynamic topology is responsible for the bad performance of several routing algorithms in mobile multi-hop ad hoc networks. The ant Colony optimization meta-heuristic is based on agent systems and works with individual ants. This allows a high adaptation to the current topology of the network.

Local work - In contrast to other routing approaches, ant Colony optimization meta-heuristic 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.

Link quality is possible to integrate the connection/link quality into the computation of the pheromone concentration, especially into the evaporation process. This will improve the decision process with respect to the link quality. It is here important to notice, that the approach has to be modified so that nodes can also manipulate the pheromone concentration independent of the ants, i.e. data packets, for this a node has to monitor the link quality.

Support for multi-path - Each node has a routing table with entries for all its neighbors, which contains also the pheromone concentration. The decision rule, to select the next node, is based on the pheromone concentration on the current node, which is provided for each possible link. Thus, the approach supports multi-path routing.

ANT COLONY OPTIMIZATION ROUTING ALGORITHM (ACORA)

The ant colony optimization algorithm (ACORA), is a probabilistic technique for solving computational problems which can be reduced to finding good paths through graphs. This algorithm is a member of ant colony algorithms family, in swarm intelligence methods, and it constitutes some meta heuristic optimizations.

ACORA has two phases. They are: Route Discovery phase and Route Maintenance phase. Both the phases use these FAnt (figure 2a), BAnt (Figure 3a) and merging FAnt and BAnt (Figure 4a). The FAnt is for the collection of information and BAnt is for feedback to the forwarding mode.

Route Discovery

In the route discovery phase new routes are created. The creation of new routes requires the use of a *Forward Ant* (FAnt) and a *Backward Ant* (BAnt). A FAnt is an agent which establishes the pheromone track to the source node. In contrast, a BAnt establishes the pheromone track to the destination node. The FAnt is a small packet with a unique sequence number. Nodes are able to distinguish duplicate packets on the basis of the sequence number and the source address of the FAnt.

A Forward Ant is broadcasted by the sender and will be relayed by the neighbors of the sender (figure 2b). A node receiving a FAnt for the first time creates a record in its routing table. A record in the routing table is a triple and consists of (destination address, next hop, pheromone value). The node interprets the source address of the Forward Ant as destination address, the address of the previous node as the next hop, and computes the pheromone value depending on the number of hops the FAnt needed to reach the node.

Then the node relays the FAnt to the neighbors. Duplicate FAnt are identified through the unique sequence number and destroyed by the nodes. When the FAnt reaches the destination node, it is processed in a special way.

The destination node extracts the information of the FAnt and destroys it. Subsequently, it creates a BAnt and sends to the source node (Figure 3b). The BAnt has the same task as the FAnt, i.e. establishing a track to this node. When the sender receives the BAnt from the destination node, the path is established and data packets can be sent.

```
FOR EACH periods DO
Get timer (); Set prone ACORA ();
IF (pheromone ACORA ant return) THEN
Get timer (); bandwidthi = Get bandwidth (pathi);
delayi =Get delay (pathi); packet-lossi = Get packet-loss (pathi);
Throughput =Get Destination (packeti)-Get Source-send (packet i)*100;
END IF
```

Fig.2a. FAnt algorithm for ACORA.

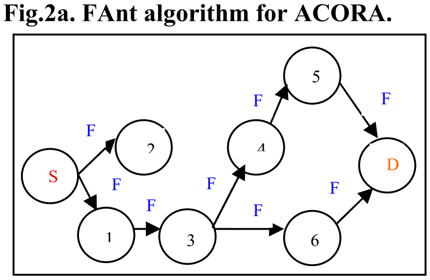

Fig.2b. A FAnt (F) is send from the sender (S) towards the destination node (D). The FAnt is relayed by other nodes, which initialize their routing table and the pheromone values

```
IF ACORA = TRUE THEN
DAi = Get Destination addr (packeti); SAi = Get Source addr (packeti);
IF transport layer packet = TRUE THEN
DPAi = Get Destination port (packeti); SPAi = Get Sourceport (packeti);
next-hop address lookup (DAi, SAi, DPAi, SPAi);
ELSE next-hop address lookup (DAi, SAi);
END IF
ELSE next-hop address lookup (DAi)

DA → Destination Address
SA→ Source Address
SPA→ Source Port Address
i→ is the number of data transmitted in path
DPA→ Destination Port Address
```

Fig.3a. BAnt algorithm for ACORA

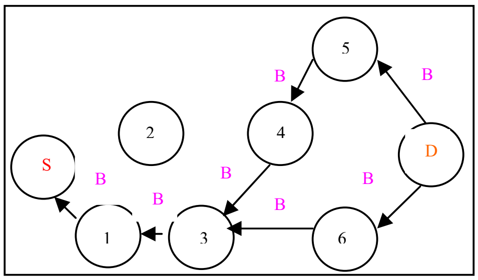

Figure 3b. The BAnt (B) has the same task as the FAnt. It is send by the destination node toward the source node

```
WHILE NOT (all packets ACORA grouped)
FIND DAi = Get Destination addr (packeti); SAi = Get Source addr (packeti) pair have the largest value in the pheromone
IF (DAi or SAi) already grouped means change the index value
GROUP (DAi or SAi)
COMPUTE (distance) VALUE of DAi and SAi
DELETE DAi, SAi (repeated key value)
END
```

Fig.4a. Merging FAnt and BAnt algorithm for ACORA

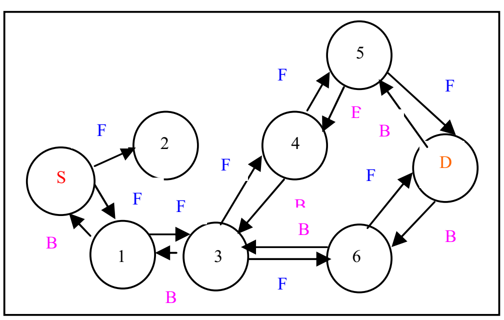

Fig.4b. Route Establishment between source node (S) and destination node (D) to send data packets by FAnt and BAnt.

Route Maintenance

The second phase of the routing algorithm is called route maintenance, which is responsible for the improvement of the routes during the communication. ACORA does not need any special packets for route maintenance. Once the FAnt and BAnt have established the pheromone tracks for the source and destination nodes, subsequent data packets are used to maintain the path. Similar to the nature, established paths do not keep their initial pheromone values forever. When a node (relay node) relays a data packet toward the destination (destination address*)* to a neighbor node (next hop), it increases the pheromone value of the entry (destination address, next hop, pheromone value*)* by pheromone function, i.e., the path to the destination is strengthened by the data packets. In contrast, the next hop (next hop) increases the pheromone value of the entry (source address, relay node, pheromone value) by pheromone function, i.e. the path to the source node is also strengthened. The evaporation process of the real pheromone is simulated by regular decreasing of the pheromone values.

The above method for route maintenance could lead to undesired loops. ACORA prevents loops by a very simple method, which is also used during the route discovery phase. Nodes can recognize duplicate receptions of data packets, based on the source address and the sequence number. If a node receives a duplicate packet, it sets the DUPLICATE ERROR flag and sends the packet back to the previous node. The previous node deactivates the link to this node, so that data packets cannot be sending to this direction any more.

ACORA handles routing failures, which are caused especially through node mobility and thus very common in mobile ad-hoc networks. ACORA recognizes a route failure through a missing acknowledgement. If a node gets a ROUTE ERROR message for a certain link, it first deactivates this link by setting the pheromone value to 0. Then the node searches for an alternative link in its routing table. If there is a second link it sends the packet via this path. Otherwise the node informs its neighbors, hoping that they can relay the packet. Either the packet can be transported to the destination node or the backtracking continues to the source node. If the packet does not reach the destination, the source has to initiate a new route discovery phase.

SIMULATION RESULTS

The ACORA is simulated under Linux Fedora-8, using the network simulator NS2 version ns-allinone-2.33. The network surface used is 1000m*1000m. The mobility scenarios are generated by the automatic generator *setdest* provided by NS2. The maximal speed of members is defined at 5km/h. The pause time is 20 seconds. The simulation duration is 300 seconds. Physical/Mac layer used is IEEE 802.11. The Mobility model used is random waypoint model with pause time equal to 20 sec and with maximum nodes movement speed equal to 3 m/s for the ACORA protocol.

The ACORA is examined the performance metrics of packet delivery ratio in nodes are in static as well as in mobility.

Packet Delivery Ratio

The number of packets originated by the MAC layer to the number of packets received by the destination is packet delivery ratio. Figure 5a shows the number of packet delivered, delay and the energy consumption of the ACORA.

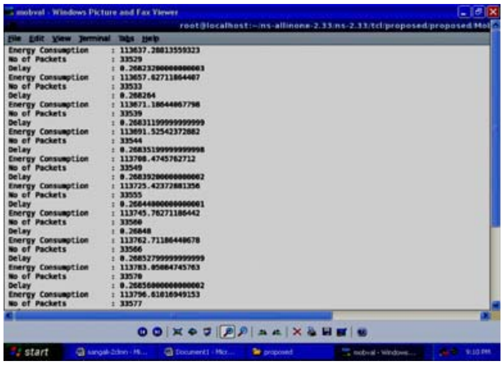

Fig.5a. Packet delivery ratio in ACORA during mobility

Packet Delivery Ratio of ACORA in Static and Mobility Nodes

Figure 6a and 6b shows the performance of the ACORA in static i.e. the packets delivered when the nodes are in static. In Figure 6a only three hops are taken, in static. In Figure 6b five hops are taken in static way for packet delivery.

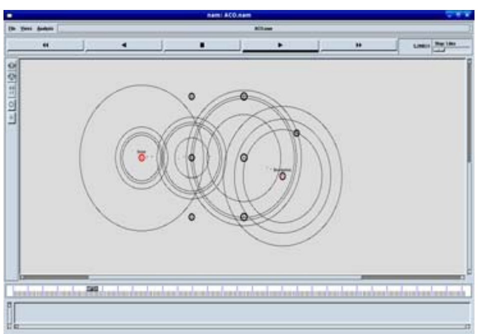

Fig.6a. ACORA for packet delivery in static (Three hops)

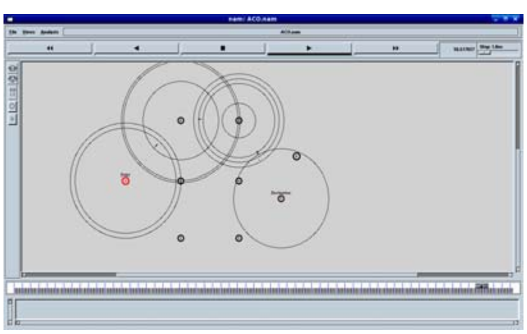

Fig.6b. ACORA for packet delivery in static (Five hops)

Also ACORA is applied for the mobility nodes. In Figure 6c the nodes are in mobility.

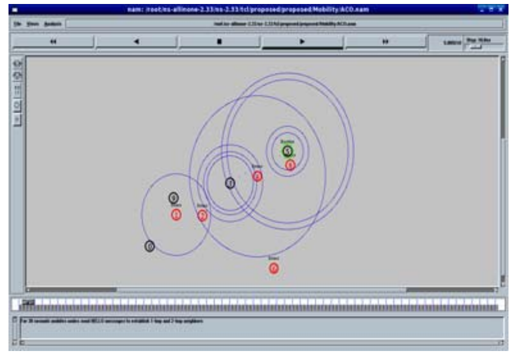

Fig.6c. ACORA for packet delivery when nodes are in mobility

The graph of the ACORA in static and mobility for the packet delivery ratio is given. The figure 6d shows the graph of the packet delivered in destination by ACORA in static.

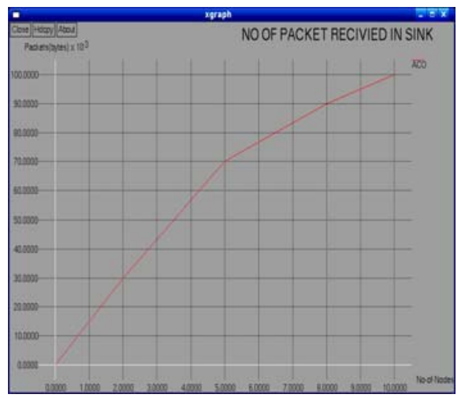

Fig.6d. ACORA for packet delivery in static

The figure 6e shows the graph of the packet delivered in destination by ACORA when nodes are in mobility. The paths vary due to mobility of the nodes even though the proposed protocol maintains the path and delivered the packets.

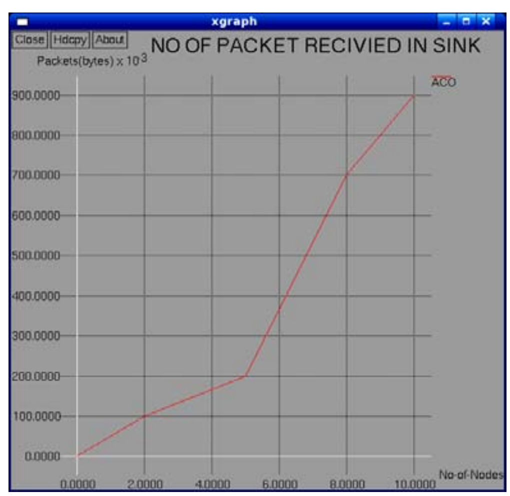

Fig.6d.ACORA for packet delivery in mobility

Conclusions And Future Work

In this paper we have implemented the proposed ACORA protocol in C++ and integrated the module in the ns Simulator. The performance of the proposed protocol was measured with respect to metrics like Packet delivery ratio and also the packet delivery ration is examined under the static nodes and the nodes are in mobility.

The results of the simulation indicate that performance of ACORA remains stable, in both the cases, static and as well as in the mobility.

In future, the performance comparison can be made between the proposed protocol and the existing protocol for performance metrics such as end-to-end delay, routing overhead, etc. of ad hoc routing protocols with different simulation parameters.

References

[1] Marco Dorigo and Thomas Stutzle, Ant Colony Optimization, 3rd ed. Harlow, England : Addison-Wesley, 1999.

[2] C.-K. Toh. Ad hoc mobile wireless networks: protocols and systems. Prentice Hall, 2002. ISBN: 0-13-007817-4.

[3] Frederick Ducatelle, Gianni Di Caro and Luca Maria Gambardella, "Using ANT Agents to combine reactive and proactive strategies for routing in mobile ad-hoc networks"

[4] Mesut Gunes, Udo Sorges, Imed Bouazizi, "ARA – The Ant-colony based routing algorithm for MANETs", International workshop on Ad-hoc Networking (IWAHN 2002).

[5] S. Prasad1, Y.P.Singh2, and C.S.Rai2, "Swarm Based Intelligent Routing for MANETs", International Journal of Recent Trends in Engineering, Vol 1, No. 1, May 2009.

[6] C. E. Perkins and P. Bhagvat. Highly dynamic destination sequenced distance-vector routing (DSDV) for mobile computers. Computer Communications Rev., pages 234 - 244, October 1994.

[7] A. Boukerche, Performance Evaluation of Routing Protocols for Ad Hoc Wireless Networks, Mobile Networks and Applications 9, Netherlands, 2004, pp. 333-342.

[8] http //en.wikipedia.org/ wiki/ Destination Sequenced_ Distance_Vector_routing

[9] A.E. Mahmoud, R. Khalaf & A, Kayssi, Performance Comparison of the AODV and DSDV Routing Protocols in Mobile Ad-Hoc Networks, Lebanon, 2007.

[10] The VINT Project, The ns Manual, http://www.isi.edu/nsnam/ns/nsdocumentation.html

[11] Marc Gries, Tutorial for Network Simulator-ns, http://www.isi.edu/nsnam/ns/tutorial

[12] Eitan Altman and Tania Jimenez, "NS Simulator for beginners"

[13] K. Fall and K. Varadhan. The ns Manual, Nov 2000.