A Hybrid Multi-agent Routing Algorithm based on Ant Colony Optimization for MANET

V. K. SARASWAT and AMIT SINGHAL

Institute of computer & Information Science, Dr. B. R. Ambedkar University,
Agra - 282 002 (India).

ABSTRACT

There are many routing algorithms are presented for Mobile Ad hoc Network such as Destination sequenced Distance Vector (DSDV) which is known as proactive routing algorithm based on routing table, due to continuously updating routing tables, it consumes large portion of network capacity. Another is Ad hoc on Demand Distance Vector (AODV) is known as reactive algorithm. In this algorithm, when path breaks, it needs a new route discovery, this cause a route discovery latency, therefore these are not suitable for real time communication. With the inspiration by Swarm Intelligence (SI), we borrowed idea from ant colony and Ant Colony Optimization framework and proposed an Enhanced hybrid routing algorithm which can improve the performance in MANET. The hybrid quality of the algorithm makes it suitable for the entire environment in comparison with reactive and proactive protocols. The introduced routing algorithm is highly adaptive, efficient and scalable. The main aim of this algorithm to reduce the end-to-end delay in the context of pause time and mobility on packet received. We refer to this algorithm as the "Ant Colony based Multi Agent Routing Algorithm (ACMRA).

Key words: Swarm Intelligence, Mobile Ad Hoc Network, Ant Colony Optimization.

INTRODUCTION

A mobile ad-hoc network (MANET) is a set of mobile nodes which communicate over radio and do not need any established infrastructure. This kind of network is very flexible and suitable for several situations and applications due to it need a temporary communication without preinstalled infrastructure[9]. There are so many important fields where mobile ad hoc network play an important role such as disaster management, military operations, rescue management, etc. These applications need frequent use of wireless environment to communicate; this can be achieved by the mobile ad-hoc network. The main problem in mobile ad-hoc network is finding of a route between the communications end-points which is due to the mobility of the nodes. Many researchers have proposed and worked on on-demand routing algorithms[1-2] ,such as AODV (Ad Hoc On-demand Distance Vector protocol)[6,7]and DSR[8] (Dynamic Source Routing Protocol), *TORA* and several

others[10] but there is no routing algorithm which fits in all cases.

In this paper, we present a new approach for On-demand Ad-hoc Network, which is based on swarm intelligence. Nature's self-organizing systems like insect societies show precisely these desirable properties. Making use of a number of relatively simple biological agents (e.g., the ants) a variety of different organized behaviors is generated at the system-level from the local interactions among the agents and with the environment. The robustness and effectiveness of such collective behaviors with respect to variations of environment conditions are key-aspects of their biological success. This kind of systems is often referred to with the term Swarm Intelligence. Swarm systems have recently become a source of inspiration for the design of distributed and adaptive algorithms, and in particular of routing algorithms. Routing is the task of directing data flows from source to destination maximizing network performance.

Several successful routing algorithms have been proposed taking inspiration from ant colony behavior and the related framework of Ant Colony Optimization (ACO)[3]. Examples of ACO routing algorithms are AntNet[4] and ABC[5].Mobile Ad hoc Network is one of the most innovative and challenging area of wireless networks. The goal of every routing algorithm is to direct traffic from source to destination, maximizing network performance while minimizing cost. In this paper, we will present a new approach for an Ad hoc routing algorithm, which is based on *Ant Colony Optimization* (ACO) algorithm and its combination with proactive and reactive algorithms. The basic idea of the ACO meta-heuristic is taken from the food searching behavior of real ants. While walking, ants deposit *pheromone*, which marks the route taken as they move from a food source to their nest, and foragers follow such pheromone trails. The concentration of pheromone on a certain path is an indication of its usage. These pheromone trails are used as a simple indirect form of communication. The process of emerging global information from local actions through small, independent agents not communicating with each other is called *Stigmergy*[11]. This behavior of the ants can be used to find the shortest path in a network. Especially, the dynamic component of this method allows a high adaptation to changes in mobile ad hoc network topology, since in this network the existence of links are not guaranteed and link changes occur very often.

The simple ant colony optimization meta-heuristic illustrates why this kind of algorithms could perform well in mobile ad hoc networks by different reasons. The main reason is that ACO meta-heuristic is based on agent systems and works with individual ants.

This allows a high adaptation to the current topology of the network. One other reason is the way of taking decision about selecting the next node that is based on the pheromone concentration on the current node which is provided for each possible link. Thus, the approach supports multipath routing[12]. In this paper, we focus on AODV and propose Multi agent Ants based Routing Algorithm with some enhancement like pheromone diffusion (ACMRA), this new ACMRA hybrid routing is able to reduce end-to-end delay and route discovery latency by

providing high connectivity compared to AODV and MARA[13], and hybrid algorithm also does not overload. We present our result, which is based on simulation made with the current implementation in NS-2[14]. The remainder of this paper is organized as follows. In section 2 we present the basics of ant colony optimization meta-heuristic techniques. The proposed Routing Algorithm ACMRA is described in section 3. Subsequently, simulated results and performance analysis is discussed in section 4. Finally, a conclusion is given in section 5.

Basics and Background of Ant Colony Optimization

The ACO meta-heuristic is based on generic problem representation and the definition of the ant's behavior. ACO adopts the foraging behavior of real ants. When multiple paths are available from nest to food, ants do random walk initially. During their trip to food as well as their return trip to nest, they lay a chemical substance called pheromone, which serves as a route mark that the ants have taken. Subsequently, the newer ants will take a path which has higher pheromone concentration and also will reinforce the path they have taken. As a result of this autocatalytic effect, the solution emerges rapidly. Given this formulation, the ants in ACO build solutions to the problem being solved by moving concurrently and asynchronously on an appropriately defined construction graph. It defines the way of solution construction, pheromone updation and possible interactions in the solution process. ACO meta-heuristic inspired by the foraging behavior of ant colonies targets discrete optimization problems. ACO's constructive
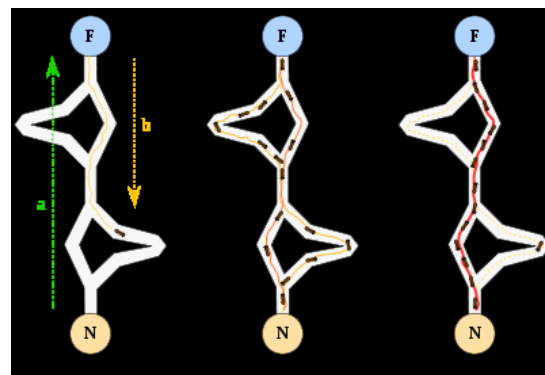

Fig. 1: All ants take the shortest path after an initial searching time

population based metaheuristic which exploits an indirect form of memory of previous performance makes it a unique approach to solve optimization problems. Figure 1 shows a scenario with two routs from the nest to the food place

An ad hoc network can be modeled as an undirected graph G = (V, E) in which the nodes set V represents the collection of wireless mobile hosts. The link set E represents the collection of any node pairs that can communicate directly [Figure 2].

Now network route finding problem is just finding a set of minimum cost path between nodes present in the corresponding graph representation which can be done easily by the ant algorithm
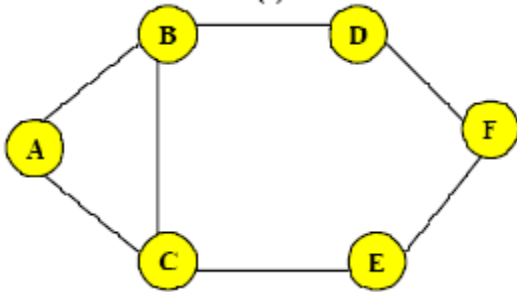

Fig. 2:

Simple ant colony optimization meta-heuristic algorithm

Let G = (V, E) be a connected graph with n =| V | nodes. The simple ant colony optimization meta-heuristic can be used to find the shortest path between a source node V_S and a destination node V_D on the graph G. The number of nodes on the path gives the path length. A variable $\varphi_{i,j}$ (artificial pheromone), which is modified by the ants when they visit the node is associated with an edge e(i, j) ε E of the graph connecting the nodes V_i and V_j. The pheromone concentration is an indication of the usage of this edge. Initially is constant for each edge e(i, j). An ant located in node V_i uses pheromone of node to compute the probability of node V_j being the next hop. N_i is the set of one-step neighbors of node V_i. The transition probability $P_{i,j}$ of a node V_i, i.e. the probability that the ant selects node V_j after it has visited V_i is defined as follows

$$P_{i,j} = \begin{cases} \frac{\varphi_{i,j}}{\sum_{j \in N_i} \varphi_{i,j}} & if\ j \in N_i \\ 0 & if\ j \in N_i \end{cases} \quad \sum_{j \in N_i} P_{i,j} = 1 \quad i \in [1,N] \qquad \ldots(1)$$

During the route finding process, ants deposit pheromone on the edges. In the simplest version of the algorithm, the ants deposit a constant amount of pheromone, i.e. the amount of pheromone of the edge e(i, j) when the ant is moving from node V_i to node V_j is changed as follows:

$$\varphi_{i,j} = \varphi_{i,j} + \Delta\varphi \qquad \ldots(2)$$

Like real pheromone the artificial pheromone concentration decreases with time which is described as follows:

$$\varphi_{i,j}(t+\tau) = (1-q)\varphi_{i,j}(t), \qquad q \in [0,1] \qquad \ldots(3)$$

General Characteristics of ACO algorithms for routing

The following set of core properties characterizes ACO instances for routing problems:

1.   Providing traffic-adaptive and multipath routing.
2.   Relying on both passive and active information monitoring and gathering.
3.   Making use of stochastic components.
4.   Not allowing local estimates to have global impact.
5.   Setting up path in a less selfish way than in pure shortest path scheme favoring load balancing.
6.   Showing limited sensitivity to parameter setting.

Proposed Routing Algorithm

This paper proposes an On-demand Hybrid Algorithm for MANET. Since the requirement for various applications may vary time to time, the approach for routing may not be proactive. The proposed approach has three phases namely route discovery phase, route maintenance phase and route failure handling. When a source node has to

pass data to a destination node it starts with the route discovery phase. Once the route is found, the data transfer will take place. While data transmission is going on, it is also required to maintain the path to the destination. This is very much desirable and required in mobile ad hoc networks and hence it is done in the route maintenance phase. If existing route fails then this route failure handling phase will generate the alternative route. Each node in the network consists of mainly two data structures such as routing table and neighbor list.

Routing table of each node is consist address of the destination, address of the adjacent node and pheromone value. Neighbor list N_i is the set all neighbors of ith node.

Route Discovery Phase

Route discovery is responsible for generating all possible routes between source and destination. It uses control packet to discover route. The control packets are mobile agents which walk through the network to establish route between nodes. Route discovery uses two mobile agents called Forward Ant (FANT) and Backward Ant (BANT) perform as acknowledgement. These two ants are similar in structure but differ in the type of work they perform. A forward ant is an agent which establishes the pheromone track to the source node and backward ant establishes pheromone track to the destination. A forward ant is broadcast by the sender and relayed by the intermediate nodes till it reaches the destination. A node receiving a forward ant for the first time creates a record in its routing table. The record includes address of the destination, address of the adjacent node and pheromone value. The node interprets the source address of the forward ant as the destination address, the address of the previous node as the next hop and computes the pheromone value depending on then number of hops the forward ant needed to reach the node. Then the node forwards the forward ant to its neighbors. A forward ant packet has unique sequence number. Duplicate forward ant is detected through sequence number. Once the duplicate ant is detected, it is dropped by the node. When the forward ant reaches the destination, its information is extracted and it is destroyed. Backward ant is created with same sequence number and sent towards the source. Backward ant

reserves the resources at along the nodes towards source. Backward ant establishes path to destination node. Once the source receives the backward ant from the destination, the path is generated and the data can be sent along with this path.

The forward ant contains (destination address, next hop, pheromone value and Total time including traveling time and delay, if any).

We assume T_{thresh} as a threshold time, this time indicates that a acknowledgement ant must return within this threshold time otherwise Source node will retransmit the duplicate forward ant with other sequence no. $t_{ij(travel)}$ is traveling time between i^{th} node to j^{th} node and there may be some situations in which an ant may be delayed, such as some time an ant may be distract from its path and after some time it get back to its original path, some time an ant may be get in sleepy mode as a natural phenomena of an ant, some more situation may encountered for the delaying an ant. So we are taking $t_{ij(delay)}$ is a delay time between i^{th} node to j^{th} node for showing delay, if any.

$$T_{(travel)} = \Sigma \; t_{ij(travel)}$$
$$T_{(delay)} = \Sigma \; t_{ij(delay)}$$
$$T = T_{(travel)} + T_{(delay)} \qquad \qquad …(4)$$

Where T is a total time taken by an ant as forward ant as well as backward ant.

Working of this phase is as follows:

Step I: (For the Source Node)

i)   Source node generates a FANT and broadcast to every node of N_i

ii)   if (Tthresh < = T)

then it will send data packet along with the path extract from BANT

else

Source node will regenerate a duplicate FANT with new sequence number and broadcast again to its neighbors.

Step II: (Any node in forward direction)

When any node receives a FANT, then it follows

if (current node = = destination node) **then**

a)   it extract the information from FANT

b)   Destroy the FANT

c)   Generates the BANT with the same sequence number as FANT

d)   Send the BANT to node from which it has received this FANT

else

a)   it increase the path length by increasing the number of hop by 1
b)   change the pheromone value using Eq. 2
c)   calculate the total time taken by an ant including traveling time and delay, if any
d)   send FANT to its neighbor.

Step III: (Any node in backward direction)
        When any node receives BANT, then it follows
                if(current node ! = source node)
a)   reserve resource at current node
b)   calculate the total time taken by an ant including traveling time and delay, if any
c)   send BANT to node from which it has received FANT
else
a)   get pheromone value of all links using N_i
b)   compute the probability for all nodes in N_i using Eq. 1
c)   send data packets to the link which has highest probability value

Route Maintenance Phase
        Route maintenance phase is responsible for the maintenance of the path generated during the discovery phase. This phase basically helps in maintaining the route which has already been established during route discovery phase. As the topology of the network changes, it is required to refresh the route between the nodes. Once the path between source and destination is set up, it is upto the data packets to maintain the route. When a node V_i forwards the data packet to node V_j to reach the destination V_D, it increments the pheromone value along the path V_j and V_D by Δ, thus strengthening the path. An acknowledgement is sent to all the packets received. If acknowledgement is not received within the threshold time $T_{thresh,}$ then the route error message is transmitted to previous node. This module is work as follows:
(Any node)
i)   if(current node = = destination node) then
a)   it extract data
b)   send packet (error acknowledge packet)

to previous node
else
a)   get pheromone values of all links using N_i.
b)   compute probability for all nodes in neighbor list using Eq. 1
c)   send packet to that link which has highest probability value.
d)   increment pheromone value for the highest probability link.
ii)   Pheromone value decreases by x naturally.
            if (pheromone value = = 0) then
                Load route discovery phase
iii)   if acknowledgement is not received at current node before threshold time, send route error to previous node (handling by error handling phase)
iv)   refresh route after time out.

Route Failure Handling Phase
        This phase is responsible for generating alternative routes in case the existing route fails. Node mobility in ad hoc network may cause certain links to fail. Every packet is associated with acknowledgement; hence if a node does not receive an acknowledgement, it indicates that the link is failed. On detecting a link failure the node sends a route error message to the previous node and deactivates this path by setting the pheromone value to zero. The previous node then tries to find an alternate path to the destination. If the alternate path exists, the packet is forwarded on that path else the node informs its neighbors to relay the packet towards source. This continues till the source is reached. On reaching the source, the source initiates a new route discovery phase. Ant algorithm provides multiple paths. If the optimal path fails, it leads to choosing next best path. Next best path will be that path with links having next highest pheromone value (second best path). Hence ant algorithm does not break down on failure of optimal path. This will help in load balancing. That is, if the optimal path is heavily loaded, the data packet can follow the next best path. This phase is implement as follows:
(Any Node)
i)   if any node received a error message then
if alternate path exist then
send packets through other route
else
a) pheromone = 0
b) Send route error message to previous node.

ii)   if route error message reaches source then it load route discovery phase

Simulation and Performance Analysis
Simulation Environment

The simulation of the algorithm was carried out using NS-2 simulator. We considered 50 nodes in a 1500M * 1000 M. Simulation is carried out for 300 seconds. The sources in the network are 20 CBR (constant bit rate) traffic generators. The data packets used for simulations are 512 bytes. The MAC layer uses the IEEE 802.11 DCF protocol for the wireless standard. The radios use two ways ground propagation model and have a receiving range of 250 meters. All the nodes in the network are constantly moving and have a random speed selected in the range of [1..20] m/s. The key performance metrics evaluated in the simulations are

i)   Effect of Mobility on Packet Received
ii)   Effect of Pause time on end-to-end Delay

Performance Analysis

The performance of the proposed routing scheme is compared with another on demand state of the ant routing protocol AODV. Fig.3 shows the effect of packets received for both the protocols in terms of mobility. In both cases higher the node mobility, lesser is the packet delivery. Because in such cases there will be frequent path breaks; hence frequently route establishment phase will be invoked in case of AODV. But due to the availability of alternate routes during the data transmission, in all
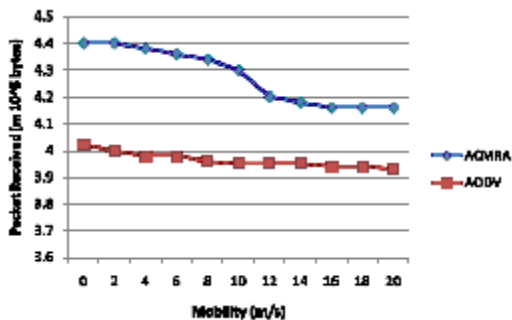

Fig. 3: Effect of Mobility on Packet Received

cases of mobility, proposed scheme dominates AODV in terms of packets received.

Fig. 4 shows the effect of pause time on end-to-end delay of AODV and proposed protocols. As the pause time is very less, there will be frequent mobility leading to frequent path breaks and path reestablishment. Hence with less pause time, the end to end delay will be higher. Since the proposed
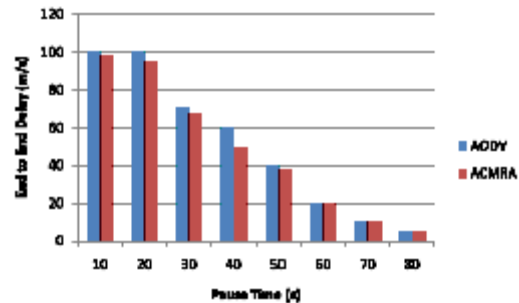

Fig. 3: Effect of Mobility on Packet Received

algorithm proactively maintains the alternate route, path changing is done in a short time compared to AODV. At higher pause time, the node stay longtime in each waypoint and hence path break will be lesser. So with higher pause time, both algorithms provide almost same and low end-to-end delay

CONCLUSIONS

In this paper, we have proposed an ant based routing protocol (ACMRA) for mobile ad hoc networks which can effectively fine the globally best solution in terms of routing for a ad hoc network that can further satisfy users need for higher bandwidth, shorter delay and also shorter distance path in terms of hop count. The simulation results indicate that proposed scheme can perform better than other algorithms under high mobility because of alternate route maintenance scheme. In future, we have planned to investigate the performance of the algorithm for real-time multimedia data using various mobility models.

REFERENCES

1. C. E. P (Editor): "Ad Hoc Networking", Addison-Wesley, 2001. ISBN 0-201-30976-9

2. C.K Toh: "Ad hoc mobile wireless networks: protocol & systems", Prentice Hall, 2002. ISBN: 0-13-007817-4

3. M. Dorigo, G. Di Caro, and L. M. Gambardella: "Ant algorithms for discrete optimization", Artificial Life, **5**(2): 137-172 (1999).

4. G. Di Caro and M. Dorigo: "AntNet Distributed stigmergetic control for communications networks", *J. of Artificial Intelligence Research,* **9**: 317–365 (1998).

5. R. Schoonderwoerd, O. Holland, J. Bruten, and L. Rothkrantz: "Ant-based load balancing in telecommunications networks, *Adaptive Behavior*", **5**(2): 169-207 (1996).

6. S. Das, C. Perkins, E. Royer: "Ad hoc on demand distance vector (AODV) routing", Internet Draft, draft-ietf-manetaodv-11.txt, works in progress, (2002).

7. Marco Dorigo and Gianni Di Caro: "The ant colony optimization meta-heuristic", In David Corne, Marco Dorigo, and Fred Glover, editors, New Ideas in optimization, McGraw-Hill, London, pages 11-32 (1999).

8. G. Di. Caro snd M. Dorigo: "Antnet: distributed stigmergetic control for communications networks", *Journal on Artificial Intelligence Research*, **9**: 317-365 (1998).

9. I. Chlamtac, M. Conti, and J. Liu: "Mobile ad hoc networking: imperatives and challenges", Ad Hoc Networks, 1 (2003).

10. M. Elizabeth, and T. Chai-Keong,: "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks", IEEE Personal Communications (1999).

11. D. Corne, M. Dorigo, and F. Glover (Eds.): "New ideas in optimization", Maidenhead, UK: McGraw-Hill (1999).

12. M. Günes, U. Sorges, and I. Bouazizi: "ARA-The Ant-Colony Based Routing Algorithm for MANETs": International Conference on Parallel Processing Workshops (ICPPW'02), IEEE Computer Society Press, pp.79-85 (2002).

13. D. Siva Kumar and R.S. Bbuvaneswaran: "Proposal on Multi agent Ants based Routing Algoritm for Mobile Ad- Hoc Networks", JCSNS International Journal of Computer science and Network Security, **6**: 260-267 (2007).

14. K. Fall and K. Varadhan: "The ns manual", (2000).