Efficient Ant Colony Optimization (ACO) based Routing Algorithm for Manets

By P. Kranthi Kumar & Dr. V. V. Rama Prasad

Sree Vidyakethan Engineering college, India

Abstract- The mobile ad-hoc networks (MANET) are infrastructure less and decentralized group of mobile nodes where the routers and mobile hosts are connected over wireless links. In MANET, nodes communicate with each other using multi-hop communication system. The nodes and routers are free to move within the network and it is a difficult task to maintain the network topology as it changes every time. So, efficient algorithms are needed for routing. Due to the suddenly changing of topology every time routing of packets from source to destination using shortest path is difficult. In present work, it is proposed by adding a factor called orientation to Ant Colony Optimization algorithm for routing in MANETs in which destination nodes are selected and exchanges the packets (agents) between the source and the destination. This algorithm helps in finding the shortest path between source and destination. The results are compared with AODV by using throughput as performance metric.

Keywords: mobile ad hoc networks, routing, ant colony optimization, dynamic topology.

GJCST-E Classification : C.2.2 C.2.3
Efficient Ant Colony Optimization (ACO) based Routing Algorithm for Manets

P. Kranthi Kumar & Dr. V. V. Rama Prasad

Abstract - The mobile ad-hoc networks (MANET) are infrastructure less and decentralized group of mobile nodes where the routers and mobile hosts are connected over wireless links. In MANET, nodes communicate with each other using multi-hop communication system. The nodes and routers are free to move within the network and it is a difficult task to maintain the network topology as it changes every time. So, efficient algorithms are needed for routing. Due to the suddenly changing of topology every time routing of packets from source to destination using shortest path is difficult. In present work, it is proposed by adding a factor called orientation to Ant Colony Optimization algorithm for routing in MANETs in which destination nodes are selected and exchanges the packets (agents) between the source and the destination. This algorithm helps in finding the shortest path between source and destination. The results are compared with AODV by using throughput as performance metric.

Keywords: mobile ad hoc networks, routing, ant colony optimization, dynamic topology.

I. Introduction

A mobile ad hoc network (MANET) is a decentralized group of mobile nodes it exchange information temporarily by means of wireless links. Since the nodes are mobile, the network topology may change dynamically and unpredictably every time. A node can communicate to other nodes which are within its transmission range. This kind of network promises low level of cost when compared with infrastructures network.

As routing is the central theme of many of the issues in MANETs thus, lots of research proposals have been reported in the literature in this direction. Basically, routing protocols unit area classed into different categories such as Proactive, Reactive and Proactive routing protocols work by exchanging topological information among the network nodes. The proactive protocols endlessly listens the topology of the network. On demand (Query-reply) are the basis of the reactive routing protocols and these establish routes to the destination only when there is a requirement of the same. Hybrid routing protocols use the combination of reactive and proactive. By using Ant Colony Optimization technique many several routing problems have been solved.

The ant colony optimization is intended by the traveling activity of ants. While ants inspects for food, they move in arbitrarily and upon finding food, they come back to their camp leaves a substance named pheromone. A lot of ants may follow different directions to the same food source. Similarly as the ants move, the longer distances lose their concentration of pheromone which guides ants to settle on shortest path. This action finds the shortest route and the main catalyst in this action of ants for indentifying the shortest route is the use of chemical volatile element called as pheromone. Ants are autonomous and self-coordinated agents that work together via indirect communication known as stigmergy.

Motivated by the on top of facts in this work, have a tendency by adding a factor called orientation ant algorithm for routing in MANETs in which destination nodes are selected and exchanges the packets (agents) between the source and the destination. The results obtained show that the proposed scheme is quite effective in comparison with various algorithms in respect of the throughput.

II. Related Work

The contributions of this work let us review some of the most relevant proposals reported in the literature.

Gunes and Spaniol (2003) proposed an Ant-Colony-Based Routing Algorithm (ARA) which is highly scalable, adaptive and efficient. The algorithm is based on Ant algorithms which are a class of SI. The main goal of algorithm was to reduce the overhead in routing (G. Baras and Mehta (2003) proposed Probabilistic Emergent Routing Algorithm (PERA) which exploits the inherent broadcast capability of wireless networks for better performance.

Caro et al. (2004) proposed Ant Hoc Net which is a multipath hybrid routing algorithm for MANETs. It does not maintain routes to all possible destinations at all times, but only for the open data sessions. Yang et al. (2006) proposed An on Demand multi-path Algorithm (ADAA). ADAA can reduce the frequency of the congestion effectively by selecting the path by its bottleneck width and distribute the traffic by multiple link-disjoint routes. The congested node will retrace the
path of FANT to inform the source to change route by reducing the pheromone value.

Wang et al. [18] introduced multicast routing for delay variation bound using a Modified ant colony algorithm (MACA). Authors have overcome the problem of slow convergence and have speed up the process of finding shortest paths in ant based algorithms for MANETs.

### III. Proposed System

Ant based algorithms are used to obtain efficient routing to develop the network performance. In the present work, it is proposed to design reinforcement learning as the framework of routing by adding a new factor called orientation, i.e., considering the direction of the destination as coordinates. In these, by employing the nature of reinforcement learning, it had been promised that the chances of a packet getting trapped in a loop are lesser and the factor orientation will help the packets to move in the correct direction. The goal of routing is to determine the shortest path between the source and the destination. Based upon the discussion, the objectives of this work are as follows:

Selection of the end nodes is done by exploring portion of topology to reduce network layer overheads and hence large number of packets can flow through the network.

### IV. Network Model

![Network Model Diagram](https://example.com/network_model.png)

*Figure 1: Generalized topology of network in the proposed scheme*

Fig. 1 show the network topology used in MANETs which is represented by $G=(V,E)$ showing wireless node a finite non empty set $V(V_1, V_2, \ldots, V_j)$ of nodes and non-empty set of links (E) representing two end points that can communicate with each other because direct communication range. Fig. 1 represents 2D view of the network topology constructed with x-y coordinates that have the orientation attribute. Suppose A, B, C, D, E, F, G & H represent eight wireless nodes in a MANET environment. Let us assume that node A wants to communicate with node H. There exist three different outgoing edges through which packets can be routed.

### V. Algorithm

**Input:** Ant Set $\{a_1, a_2, a_3, \ldots, a_m\}$, Number of nodes: $\{n_1, n_2, \ldots, n_0\}$

**Output:** Edges forming path by selecting nodes with highest probability.

1. Start
2. Procedure Sel_node()
3. while (SN = sel_node)
4. For each source node
5. Select randomly from set of nodes
6. End for
7. End Procedure
8. Procedure Initialize_pheromone()
9. Select the pheromone value as $\rho_{ij} = 1.0 / |N_i|$
10. End Procedure
11. Procedure Send_FANT()
12. Update the stack contents as stack $F[\text{top}]=PN$
13. Increment top of the stack as top = top + 1
14. END if
15. If (NN = = TN) then
16. Create the Backward ant and BANT
17. Destroy forward ant FANT
18. else if (NN == TN) then
19. Delete Ant's Memory Clear the Stack contents
20. Call Procedure Send_FANT()
21. Else
22. Assign current node as next node by assigning $PN == NN$
23. End if
24. Procedure Send_BANT()
25. While (TTL > 0) and (stack $F[\text{top}]! = \text{NULL}$)
26. Decrement top of the stack as top = top - 1
27. If (NN = = SN) then
28. Exit()
29. else if (PN = = NN) then
30. Update pheromone as follows
31. IF (node was in path of the ant) then
32. Else
33. Destroy BANT
34. END if
35. END if
36. $\rho_{ij} = \rho(i)+r*(1 + r\rho_i +r^2\rho_{\text{high}}/(c*(|N_i|-1)))$
37. END if
38. Else
39. Destroy BANT
40. END if
41. End while
42. End procedure
43. End Procedure
44. Procedure Factor(PN,TN)
45. \( x1 = \text{get\_x\_coordinate}(PN) \)
46. \( y1 = \text{get\_y\_coordinate}(PN) \)
47. \( x2 = \text{get\_x\_coordinate}(TN) \)
48. \( y2 = \text{get\_y\_coordinate}(TN) \)
49. \( \text{Return OF} \)
50. END Procedure
51. End
52. Procedure Next_selection (PN,TN)
53. For (i=1 to |Ni|)
54. Call factor (PN,TN)
55. Calculate
56. Return node\_Id with highest value \( \Omega_{ijd} \)
57. End for
58. End Procedure

VI. Workflow of Algorithm

Step 1
Select the source node and destination node randomly. Routing of packets from source to destination there exists different paths to reach the destination. Here initialization of pheromone occurs.

Step 2
FANT (Forward Ant) is created on the source side and it passes to different path s o reach the destination. During these process FANT create the stack and update by pushing the different paths data. After FANT reaches the destination from different paths from source at destination it creates BANT. BANT is created and it send to source to different paths until stack gets empty. During these process BANT update the Pheromone value \( \rho(i) \).

Step 3
By using value of pheromone Calculating Increasing of pheromone value
\[
\rho(i) = \rho(i)+r*(1-\rho(i))+r\rho_{high}/C
\]
Decreasing of pheromone value
\[
\rho(i) = \rho(i)-r\rho(i)-r^2\rho_{high}/(c*(|N_i|-1))
\]

Step 4
By dividing the whole network topology in to co ordinates\((X,Y)\) by using a factor called orientation going to update the calculation of probability \( \Omega_{ijd} \).

\[
\text{OF}(X,Y) = \frac{X_1X_2 + Y_1Y_2}{\sqrt{X_1^2 + Y_1^2} + \sqrt{X_2^2 + Y_2^2}}
\]

The orientation factor increases the probability \( \Omega_{ijd} \) of finding shortest path.

Step 5
Every time random selection of source and destination this will be done until the total network get populated. During this each and every time it affects the pheromone value. This will increase the probability \( \Omega_{ijd} \) which helps in finding the shortest path.

VII. Results

To implement the Efficient Ant Colony Optimization Routing Algorithm NS 2.35 simulator is used on Ubuntu operating system. For getting the result 20 number of nodes which placed randomly in the area of 1000x1000 and simulation time is 100sec. Analyses and implement Efficient Ant Colony Optimization Routing Algorithm and then compare the result with AODV protocol. Then compare the result on the basis of throughput. Fig. 2 shows that the throughput of Efficient Ant Colony Optimization Routing Algorithm is sharply increase after some time. Throughput of Efficient Ant Colony Optimization Routing Algorithm is more when it compared with AODV.

![Graph showing comparison between Efficient Ant Colony Optimization Algorithm vs AODV](image_url)

Figure 2: Graph shows the comparison on the basis of throughput

VIII. Conclusion

In this work, it is proposed a new algorithm by using a factor called orientation which divides the MANETs area into coordinates and help the exchange of packets between source and destination by using shortest path. The designed algorithm has highest pheromone value which avoid local maxima problem in existing solutions. Future work is needed the algorithm may adoptable to large dimension topology of MANETs and some security issues would like to integrate.
References Références Referencias
