Improved Ant Colony Optimization Technique for Mobile Adhoc Networks

Mano Yadav[1], K.V. Arya[2], and Vinay Rishiwal[3]

[1] ITS Engineering College, Greater Noida, UP, India
[2] ABV-IIITM, Gwalior, India
[3] MJP Rohilkhand University, Bareilly, UP, India
mano425@gmail.com, kvarya@iiitm.ac.in, vrishiwal@mjpru.ac.in

Abstract. Efficient routing is a crucial issue in Mobile Adhoc Networks. This paper proposes an improved algorithm for routing in mobile adhoc networks based on the Ant Colony Optimization (ACO) technique. The proposed improved ACO (I-ACO) performs routing by making use of transition probability among nodes along with available pheromone update information in ACO principle. This approach reduces the cost of ant agents. I-ACO has two phases *viz* route discovery and route maintenance and also utilizes the concept of backtracking when the packets reaches destination node.

Simulation results show that I-ACO achieves better packet delivery ratio and reduces the average end-to-end-delay as compare to its counterpart AODV. This scheme can be incorporated in any version of the on demand routing protocol. In this paper it has been implanted in the basic AODV protocol.

Keywords: Ant Colony Algorithm, AODV, Mobile Adhoc Networks.

1 Introduction

Mobile Ad-hoc network (MANET) [1] [2] is an infrastructure less network consisting of mobile nodes. These nodes constantly change the topologies and communicate via a wireless medium. The transmission range of the nodes in MANETs is usually limited [3], so nodes within a network need to reply data packets over several intermediate nodes to communicate. Thus nodes in MANETs act as both hosts and routers. Since the nodes in MANETs are mobile, designing an appropriate routing technique for MANETs needs to be flexible enough to adapt to arbitrarily changing network topologies, and to support efficient bandwidth and energy management, since low powered batteries operate the nodes [4].

ACO [6] [7] make use of computational intelligence [5] and simulates the behaviour of ant colonies in nature as they forage for food and find the most efficient routes from their nests to food sources. The ACO technique for routing in MANET uses stigmergic [11] process to determine the best possible routes from a source node to a destination node. Artificial ants are placed at each node and they mark their trails with pheromone as they move within the network. The level of concentration of pheromone on a trail is a reflection of its quality [8] [9]. For multiple objectives [10] more than one colony can be maintained which interact to share the information gained and hence, give those paths which lie in optimal range.

N. Meghanathan et al. (Eds.): CCSIT 2011, Part I, CCIS 131, pp. 1–13, 2011.

ACO research has not been used much in MANETs and a good review of ACO application is given in [6]. In Early applications ACO was applied for routing in Telecommunication networks and a successful algorithm in this domain is Ant Net [11]. Existing literature [1] [2] [12] [13] [14] presents the idea of traditional and ant colony based routing in MANETs.

In this work we present an improved optimization technique for mobile adhoc networks using ACO concept. Rest of the paper is organized as follows: Section 2 presents the related work. In section 3 application of Ant Colony Optimization algorithm on MANETs is discussed. The Model Formulation is discussed in section 4. Simulation results are presented in section 5 and in section 6 conclusion and direction for future work are given.

2 Related Work

The concept of ant colony optimization is explained by Dorigo and Stutzle [6] and a hypercube framework for ACO is presented by Blum et al. [7]. Zuan et al. [15] have presented an improved adaptive ant colony routing algorithm (ARAAI) for MANETs. Their routing algorithms are based on ant colony optimization to find routing paths that are close to the shortest paths even if the nodes in the network have different transmission ranges. The drawback of these algorithms is the long delay due to large number of messages needs to be sent for path establishment. Kamali and Opatrny [16] have proposed POSANT, a reactive routing algorithm for mobile ad hoc networks which combines the idea of ant colony optimization with nodes position information. They claimed that POSANT has a shorter route establishment time while using a smaller number of control messages than other ant colony routing algorithms. Wang et al. [17] have proposed a hybrid routing algorithm (HOPNET) for MANETs based on the framework of ACO and zone routing [18]. HOPNET is based on ants hopping from one zone to the next, consists of the local proactive route discovery within a node's neighborhood and reactive communication between the neighborhoods. The algorithm has features extracted from Zone Routing [18] and DSR [1] protocols and this algorithm has been compared to AODV [20] and AntHocNet [21] routing protocols. HOPNET is highly scalable for large networks compared to AntHocNet. Cauvery and Vishwanatha [8] have given proposal is to improve the performance of the Ant colony algorithm [22]. This algorithm is on-demand source initiated routing algorithm. This is based on the principles of swarm intelligence. The algorithm is adaptive, scalable and favors load balancing. The improvement suggested in [8] handles loss of ants and resource reservation.

3 Applying ACO on MANETs

The principles of ACO routing can be used to develop a suitable solution routing in MANETs. Here an agent is used which is referred as Ant. Initially Ant is forwarded to explore all possible paths of the network using flooding and then the same Ant backtracks to establish the path information acquired leaving a pheromone amount on each node. Data packets are stochastically transmitted towards nodes with higher pheromone concentration along the path to the destination.

The network under consideration is represented as a connected graph *G= (V, E)*, with |V|= *N* nodes. The metric of optimization can be a parameter defined by the user which may depends upon factors like number of hops, bandwidth, delay and other network conditions. The network consists of mainly two data structures (i) routing table and (ii) visited array. Each node in the network has a routing table whose size is the degree of the node times number of nodes in the network. Each pair (row, column) in the routing table has two values *viz* the pheromone concentration and the desirability of the neighbor.

Each ant maintains an array which contains the node id, of visited nodes in searching the path to the destination. The goal is to avoid the loops in the path traced by the ant while forwarding and if a loop is found during path searching then the corresponding ant is discarded.

In this paper the routing algorithm carries out three main activities namely route discovery, route maintenance and packet forwarding. The route discovery phase in the network consists of two phases forwarding and backtracking. A source node sends ants to all its neighbors in search of the destination node and the number of ants is equal to the out-degree of the source node. Each ant maintains its own visited array of nodes (say visit array) to remember its path. On an intermediate node an Ant divides itself into the number of Ants which is equal to out-degree. If an intermediate node is in the visit array of the ant then that path is discarded and further propagation of ant is stopped, otherwise that node is added to the visit array of that ant.

After reaching the destination, the ant backtracks the path according to its visit array. Pheromone value is updated in the routing table of the node according to the quality of that path. In backtracking, if a link breakage is found ant retraces the path to destination to decrease the pheromone concentration at the nodes in visit array with the same amount and that path is discarded at destination. The complete process is detailed in algorithm 1 and algorithm 2.

Algorithm 1: FORWARDING

```
1. At source node do the following
      1.1 create an ant
      1.2 add node to visit array
      1.3 Broadcast to all neighbors.
2. At intermediate node do the following
      2.1 IF node exists in visited array Then
            discard the path and stop that ant.
          Else add the node in visit array.
      2.2 Maintain the information about quality of path.
      2.3 Forward ant to all neighbours.
```

Algorithm 2: BACKTRACKING

```
1. At destination node
      1.1 Backtrack the path according to the entries in the visit
          array.
      1.2 IF a link breakage found Then
```

```
1.2.1 Decrease pheromone concentration with the same
  amount on each backtracked node.
1.2.2 Decrement desirability counter of that node
  Else
1.2.3 IF desirability > threshold Then
        set desirability equal to initial value
      Else increment desirability counter by one.
Update pheromone concentration at each node backtrack according
  to quality of the path.
```

Forwarding of packets

Established routes are maintained in the routing table. When a packet is to be delivered to a destination node, source node computes transition probability for each neighbor using desirability and pheromone concentration. The node with maximum probability is chosen as the next node for forwarding the packet. The process is described in Algorithm 3.

Algorithm 3: PACKET FORWARDING

```
1. Identify source node and destination node.
2. At each node
     2.1 Compute transition probability for all neighbors (see
   3.1).
     2.2 Find the neighbor with maximum transition probability,
   set it as next node.
     2.3 Repeat steps 2.1 to 2.2 till packet reaches to destination.
```

Route Maintenance

Hello message to detect and monitor the links to neighbours is periodically broadcast to its neighbours by each node. If a node fails to receive Hello messages from a neighbor a link breakage is detected and its routing table is updated by deleting the information about that neighbor. The procedure is given in Algorithm 4.

Algorithm 4: ROUTE MAINTENANCE

```
1. Each node broadcasts hello message to all its neighbors.
   If a link breakage is detected
   Then update routing table with deleting entries for that
   neighbor.
   Else update link parameter information.
2. Evaporate pheromone concentration periodically.
```

4 The Model Formulation

The model formulation on different parameters used in the development of the proposed algorithm is formulated as follows:

4.1 Transition Probability

Transition probability [ ] primarily helps in finding location of the new pixel in the movement of ants and is defined as

$$p_{i,j} = \frac{(\tau_{ij}^{\alpha})(\eta_{ij}^{\beta})}{\sum(\tau_{ij}^{\alpha})(\eta_{ij}^{\beta})}$$

Where,

τ_1, τ_2 represents the pheromone matrix element (x,y) for constraint 1 and 2 respectively, similarly η_1, η_2, respectively presents the desirability of node (x,y) for constraint 1 and 2.

α and β are the parameters which control the influence of τ & η respectively, Here constraints are factors which restrict the movement on predefined area. In this simulation value of α is taken as 7 and β as 3.

4.2 Directional Probability

The directional probability [ ] increases the chances to choose the next node that is in the direction of destination. The movement in east-west direction is decided by,

East-west (ew) direction,

$$ew = (x2-x)*constt\,/\,penalty[k][x][y]$$

Similarly north-south direction is given by,

$$ns = (y2-y)*constt\,/\,penalty[k][x][y]$$

$$xlim[k] = \begin{cases} xlim[k] - ns, & x = s, se, sw;\ \ where\ ns < 0 \\ xlim[k] + ns, & x = n, ne, nw;\ where\ ns < 0 \\ xlim[k] - ew, & x = n, nw, sw;\ where\ ew < 0 \\ xlim[k] + ew, & x = n, ne, se;\ where\ ew \geq 0 \end{cases}$$

Where directions are represented by letters as, n: north, *e*: east, *s*: south, *w*: west, *ne*: north-east, *se*: south-east, *nw*: north-west, *sw*: south-west; and *k is* particular constraint; *x:* x-coordinate, *y*: y-coordinate, x_2: x-coordinate of destination, y_2: y-coordinate of destination; and *xlim[k] d*etermine directional probability in direction x for constraint k.

4.3 Pheromone Update

$$\tau_{,x,y} = \rho\tau_{x,y} + \Delta\tau_{x,y}$$

where,

$\tau_{x,y}$, amount of pheromone on a node (x,y) for each constraint

ρ, rate of pheromone evaporation

$\Delta\tau_{,x,y}$, amount of pheromone deposited, typically given by

$$\Delta\tau_{x,y}^{k} = \begin{cases} \frac{1}{L_k}, & if\ pixel\ (x,y) is\ visited \\ 0, & otherwise \end{cases}$$

where, L_k is the cost of the kth ant's tour.

5 Simulation Framework and Results

Proposed algorithm I-ACO has been evaluated using Network Simulator (NS-2) [19]. Network test beds have been setup for 25, 50, 75 and 100 nodes in an area of 1000 * 1000 m^2. The performance of the proposed work (I-ACO) has been measured for packet delivery ratio and average end-to-end-delay (in seconds) by varying the values pause time from 0 to 150 (seconds) and speed of node movement from 5 to 20 (meter/second) for different network scenarios. Details of the simulation framework have been given in Table 1.

5.1 Average End to End Delay

Average end-to-end delay is the delay experienced by the successfully delivered packets in reaching their destinations. It denotes the efficiency of the underlying routing technique as delay primarily depends on optimality of path chosen.

Scenario 1: pause time 0-100 seconds, number of nodes: 50, Speed: 15 m/sec

I-ACO has been compared to AODV in terms of average end-to-end-delay by varying pause time from 0 seconds to 150 seconds while keeping the speed of the nodes constant at 15 m/sec. The variation of End-to-End-Delay with respect to pause time is shown in Figure 2. It is clear from Figure 2 that initially (for 0 seconds pause time) AODV takes less time as compare to I-ACO. This is due to the initial calculations involved for setting up a route in I-ACO. As the pause time increases the average delay in both the schemes (I-ACO and AODV) is decreasing, but for all the pause times greater than 0 (zero), I-ACO exhibits less average delay as compare to its counterpart AODV.

Scenario 2: Number of Nodes: 25, 50, 75 and 100, pause time: 50 seconds, speed: 15 m/sec

The comparison of average end-to-end-delay for I-ACO and AODV with varying number of nodes (25 to 100) while keeping the value of pause time (50 seconds) and speed (15 m/sec) constant is shown in Figure 3. It is observed from Figure 3 that for any size of network from 25 nodes to 100 nodes, I-ACO bear very less delay as compare to AODV.

Table 1. Simulation Scenarios and Parameter Values for I-ACO and AODV

Parameter	**Value**
Simulation Time	900 seconds
Mobility Model	Random Way Point Model, Pause Time (0-150 seconds) Speed (5-20 m/sec)
MAC Protocol	802.11
Routing protocol	AODV
Network Size (Nodes)	25-100
Propagation Model	Two Ray Ground
Time Between Retransmitted requests	500 ms
Size of Source Header Carrying an Address	4n + 4 Bytes
Timeout for Non Propagating Search	30 ms
Maximum Rate for Sending Replies from a Route	1/s
Node Transmission Range (TRange)	150-300 m.
Terrain Area	1000 x 1000 m^2
A to control the influence of τ	7
B to control the influence of η	3
P (rate of evaporation of pheromone content)	√pheromone content

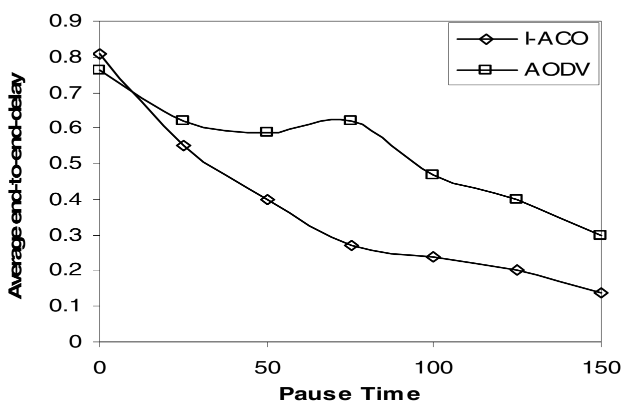

Fig. 2. Pause Time Vs End-to-end-delay (for 50 nodes)

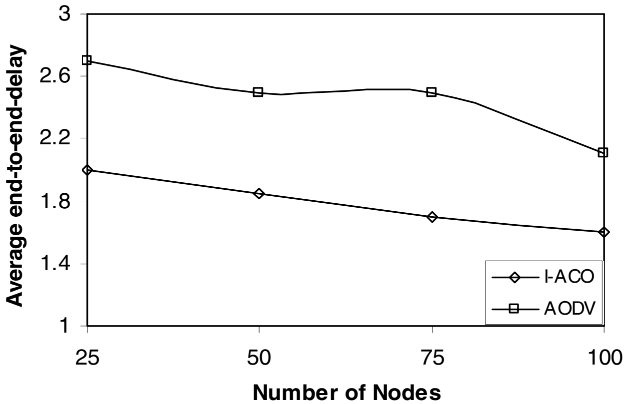

Fig. 3. Number of Nodes Vs end-to-end-delay

Scenario 3: Number of nodes: 50, speed: 15 m/sec, pause time: 50 seconds

In scenario 3, I-ACO has been compared to AODV keeping pause time and number of nodes constant while varying the speed from 5 m/sec to 20 m/sec and the effect has been shown in Figure 4. It can be observed in Figure 4 that for all the values of speed from 5 m/sec to 20 m/sec both the scheme bears more delay as the speed increases. But for all the values of speed the average end-to-end-delay for I-ACO is much less as compared to AODV.

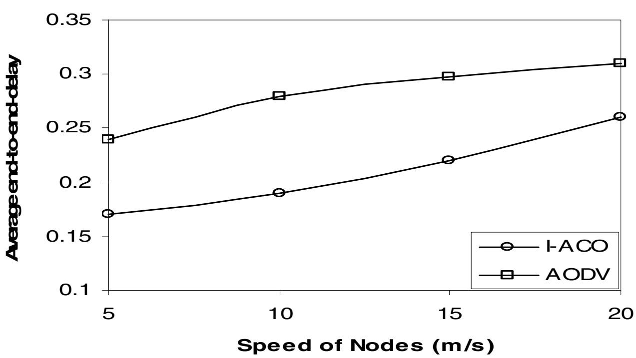

Fig. 4. Speed Vs end-to-end-delay

Scenario 4: Number of nodes: 50, speed: 15 m/sec, pause time: 50 seconds, TRange: 150m-300 m

Scenario 4 is a special case of three previous scenarios. In this scenario the value of pause time, speed and number of nodes is kept constant and the transmission range of the participating nodes has been varied from 150 meters to 300 metes and average end-to-end-delay in case of both the schemes *viz* I-ACO and AODV has been observed.

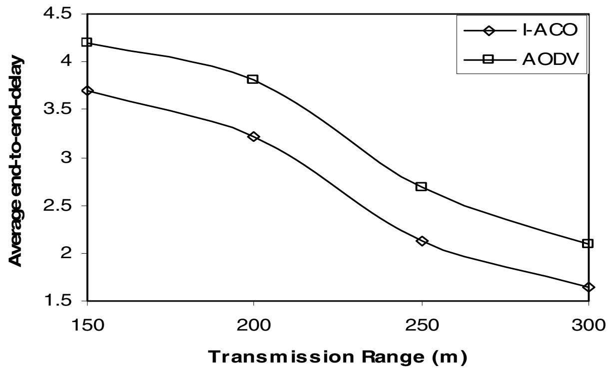

Fig. 5. Transmission range Vs end-to-end-delay

It can be seen in Figure 5 that as the transmission range of the nodes increases the value of the average end-to-end-delay decreases for both the schemes. But once again I-ACO is a better option.

5.2 Packet Delivery Ratio

PDF is defined as the fraction of successfully received packets, which survive while finding their destination. PDF determines the efficiency of the protocol to predict a link breakage and also the efficacy of the local repair process to find an alternate path. The completeness and correctness of the routing protocol is also determined.

Scenario 1: pause time 0-150 seconds, number of nodes: 50, Speed: 15 m/sec

As shown in Figure 6, I-ACO has been compared to AODV in terms of packet delivery ratio for varying pause time from 0 seconds to 150 seconds while keeping the speed of the nodes constant i.e. 15 m/sec. It can be seen in the figure 6 that, for all the values of pause time from 0 seconds to 150 seconds, packet delivery ratio increases as the value of the pause time increases. But for all the pause times greater than zero I-ACO is able to deliver more number of packets as compared to AODV. Even for lager value of pause time (150 seconds), I-ACO is able to deliver almost 99% packets successfully while AODV is delivering only 97.32 % of packets for the same value of pause time.

Scenario 2: Number of Nodes: 25, 50, 75 and 100, pause time: 50 seconds, speed: 15 m/sec

Figure 7 shows a comparison of I-ACO and AODV for packet delivery ratio for varying number of nodes (25 to 100) keeping the value of pause time (50 seconds) and speed (15 m/sec) constant. Figure 3 witnesses that for any size of network from 25 nodes to 100 nodes I-ACO deliver more number of packets successfully as compare to AODV. For the largest network of 100 nodes I-ACO is delivering more than 98% of the packets successfully while AODV is delivering only 94% of the packets for the same size of the network.

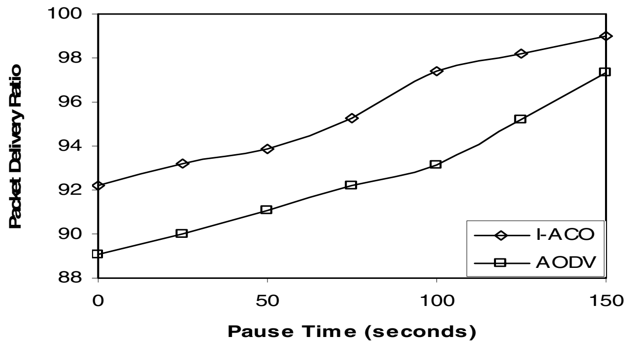

Fig. 6. Packet delivery ration Vs. pause Time

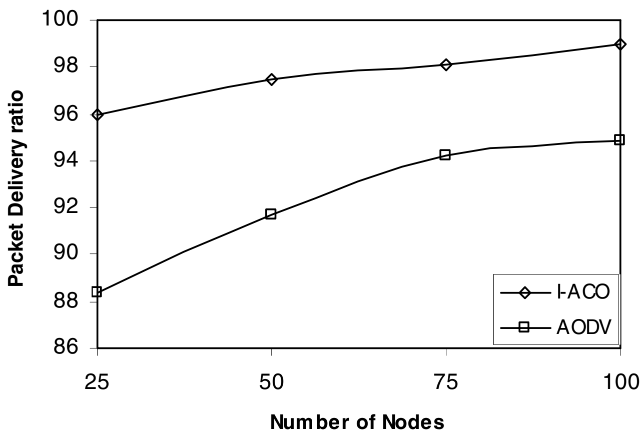

Fig. 7. Number of Nodes vs. packet delivery ratio

Scenario 3: Number of nodes: 50, speed: 15 m/sec, pause time: 50 seconds

In scenario 3, I-ACO has been compared to AODV keeping pause time and number of nodes constant while varying the value of speed from 5 m/sec to 20 m/sec and the effect has been shown in Figure 8. It can be observed in Figure 8 that for all the values of speed, I-ACO delivers more packets successfully than AODV. Even at a very high speed of 20 m/sec I-ACO delivers approximately 94% packets successfully while AODV can deliver only 82% packets.

Scenario 4: Number of nodes: 50, speed: 15 m/sec, pause time: 50 seconds, TRange: 150m-300 m

In Scenario 4, the value for pause time, speed and number of nodes is kept constant. Only the transmission range of the participating nodes has been varied from 150 meters to 300 metes and the effect of this factor has been observed on packet delivery ratio in case of both the schemes *viz* I-ACO and AODV.

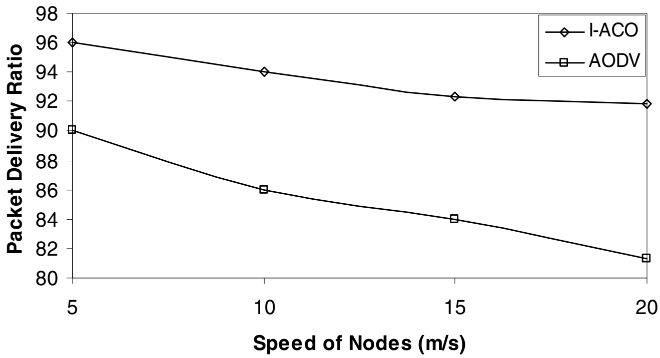

Fig. 8. Speed Vs. Packet delivery ratio

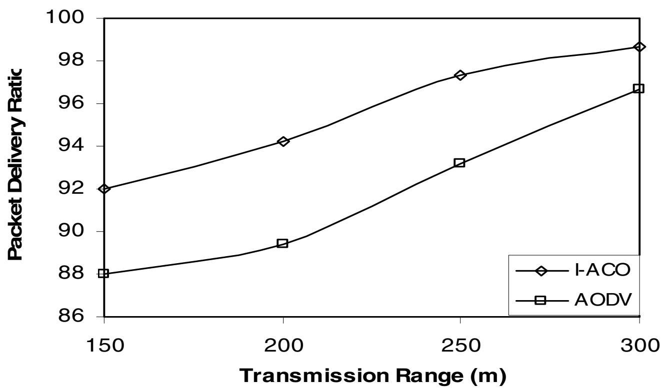

Fig. 9. Transmission range Vs. packet delivery ratio

It can be seen in Figure 9 that, increased transmission range of the nodes increases the value of the packet delivery ratio for both the schemes. But once again I-ACO is a better option in terms of number of packets successfully for different values of transmission range of the nodes from 150 meters to 300 meters. At a very high transmission range of 300 meters I-ACO is able to deliver more than 98% of the packets successfully while AODV can deliver approximately 96%.

6 Conclusion and Future Work

In this paper an improved Ant Colony Algorithm (I-ACO) has been proposed to search optimal paths in MANETs. In the proposed work an approach for routing in MANETs has been given and evaluated. The algorithm is very dynamic and robust. It incorporates mobility which is a key feature to making it adaptable to an Ad-hoc environment. The algorithm is also able to perform route maintenance and handle link failure in network.The simulation results show that the algorithm is able to achieve better packet delivery ratio and bears less average end-to-end-delay as compare to AODV for the different values of pause time, speed , network size and transmission range. The scheme analysis is still on for checking the values of control overheads occurred for different network scenarios as well as I-ACO needs a better approach for energy management in the network.

References

1. Abolhasan, M., Wysocki, T., Dutkiewicz, E.: A Review of Routing Protocols for Mobile Adhoc Networks. Adhoc Networks 2(1), 1–22 (2004)
2. Chenna Reddy, P., Chandrasekhar Reddy, P.: Performance Analysis of Adhoc Network Routing Protocols. In: International Symposium on Ad Hoc and Ubiquitous Computing (ISAUHC 2006), pp. 186–187 (2006)
3. Chen, Y., Siren, E.G., Wicker Stephan, B.: On selection of Optimal Transmission Power for Adhoc Network. In: Proceedings HICCS 2003, vol. 9, pp. 300–309 (2003)
4. Doshi, S., Brown, T.X.: Minimum Energy Routing Schemes for a Wireless Adhoc Networks. In: Proceedings of IEEE INFOCOM, pp. 103–111 (2002)
5. Engelbrecht, A.P.: Computational Intelligence-An Introduction, 2nd edn. John Wiley & Sons, Chichester (2007)
6. Dorigo, M., Stutzle, T.: Ant Colony Optimization. MIT Press, Cambridge (2004)
7. Blum, C., Roli, A., Dorigo, M.: The hyper-cube framework for Ant Colony Optimization. IEEE Trans. on Systems, Man, and Cybernetics – Part B 34(2), 1161–1172 (2004)
8. Cauvery, N.K., Vishwanatha, K.V.: Enhanced Ant Colony Based Algorithm for Routing in Mobile Ad-hoc Network. World Academy of Science, Engineering and Technology, 30–35 (2008)
9. Osagie, E., Thulasiraman, P., Thulasiram, R.K.: PACONET: Improved Ant Colony Optimization Routing Algorithm for mobile ad-hoc networks. In: International Conference on Advance Information Networking and Apllications (AINA 2008), pp. 204–211 (2008)
10. Dorigo, M., Maniozzo, V., Colorni, A.: The Ant System: Optimization by a colony of cooperating agents. IEEE Trans. on System, Man, and Cybernetics, Part-B 26(1), 29–41 (1996)
11. Di Caro, G., Dorigo, M.: AntNET: Distributed Stigmergetic Control Communications Network. Journal of Artificial Research, 317–365 (1997)
12. Gutjahr, W.J.: A Graph Based Ant System and its Convergence. Future Generation Computer Systems 16(9), 873–888 (2000)
13. Dorigo, M., et al.: Review of Ant colony Optimization. Journal of Artificial Intelligence 165(2), 261–264 (2005)
14. Fu, Y., Wang, X., Shi, W., Li, S.: Connectivity Based Greedy Algorithm with Multipoint Relaying for Mobile Ad Hoc Networks. In: The 4th International Conference on Mobile Ad-hoc and Sensor Networks,MSN 2008, pp. 72–76 (2008)
15. Yuan-yuan, Z., Yan-xiang, H.: Ant routing algorithm for mobile ad-hoc networks based on adaptive improvement. In: Proceedings, International Conference on Wireless Communications, Networking and Mobile Computing, vol. 2, pp. 678–681 (2005)
16. Kamali, S., Opatrny, J.: POSANT: A Position Based Ant Colony Routing Algorithm for Mobile Ad-hoc Networks. In: Third International Conference on Wireless and Mobile Communications, ICWMC, p. 21. (2007)

17. Wang, J., Osagie, E., Thulasiraman, P., Thulasiram, R.K.: HOPNET: A hybrid ant colony optimization routing algorithm for mobile ad hoc network. Journal of Adhoc Networks 7(4), 690–705 (2009)
18. Haas, Z.J., Pearlman, M.R., Samar, P.: The Zone Routing Protocol (ZRP) for Ad Hoc Networks, IETF Internet Draft, draft-ietf-manet-zone-zrp-04.txt (July 2002)
19. Fall, K., Varadhan, K. (eds.): The ns Manual, The VINT Project. Collaboration between researchers at UC Berkeley, LBL, USC/ISI, and Xerox PARC (November 2007)
20. Perkins, C.E., Royer, E.M.: Ad Hoc On-Demand Distance Vector Routing. In: Proceedings of IEEE Workshop on Mobile Computing Systems and Applications, pp. 90–100 (1999)
21. De Gianni, C., Frederick, D., Maria Gambardella, L.: AntHocNet: An ant-based hybrid routing algorithm for mobile ad hoc networks. In: Proc. International conference on parallel problem solving from nature, pp. 461–470 (2004)
22. Cauvery, N.K., Viswanatha, K.V.: Ant Algorithm for Mobile Ad Hoc network. In: Proceedings of the International Conference on Advanced Computing and Communication Technologies for High Performance Applications (2008)