MANET Link Performance Parameters using Ant Colony Optimization Approach

Sarbjeet Kaur
Department of Electronics Technology, Guru Nanak Dev University,Amritsar, Punjab, India

Ravinder Singh Sawhney
Department of Electronics Technology, Guru Nanak Dev University,Amritsar, Punjab, India

Rajan Vohra
Department of Electronics Technology, Guru Nanak Dev University,Amritsar, Punjab, India

ABSTRACT
End-to-end delay of routing packets from the source to destination in a Mobile ad-hoc network is a major challenge for a large number of network scientists. This paper presents a new routing approach for mobile ad hoc networks (MANETs) which takes into consideration the Ant Colony Optimization (ACO) technique in conjunction with linear programming approach for minimizing the overall delay in networking environment. Ant colony optimization algorithms have all been inspired by a specific foraging behavior of colonies of the ants, which are able to explore the optimum route connecting the colony's nest with a source of food. It has been observed that the mean end-to-end delay of a link from the source to the destination is one of the most important metric for a MANET. The other metrics that have been considered and evaluated are utilization or efficiency of link, arrival rate and service rate with respect to link delay. The simulation tool used for our model is MATLAB 7.5.

General Terms
MANET,Delay,Nodes

Keywords
Ant colony optimization (ACO), MANET, End to end delay, utilization, arrival rate, service rate, Queuing Network analysis model.

1. INTRODUCTION
Mobile ad-hoc network (MANET) has emerged as one of the most innovative and challenging area of wireless networking that promises to become increasingly present in our day-to-day lives. The basic architecture of MANET consists of nodes that are freely but dynamically self-organized (i.e. nodes are autonomous) into arbitrary and temporary network topology without any infrastructure support [1]. The advantage of employing MANET is to offer a large degree of freedom at a minimal cost in comparison to other networking solutions. The ease and speed of deployment of these networks make them ideal for recovery after a natural or manmade disaster (hurricane, earthquake, flooding or nuclear explosion), business associates sharing information during a meeting or conference, and military communications in a battlefield. One of the main hurdles that a network designer normally encounters in mobile ad-hoc networks is to find the best route between the communication end-points, which may be difficult due to node mobility. Several routing algorithms have been proposed in the existing literature. Some of these like DSDV, AODV, DSR, and TORA have been extensively simulated for various scenarios [2]. The optimum goal of every routing algorithm is to direct traffic from sources to destinations, maximizing network performance while minimizing cost. In this paper, we have presented a new algorithm designed specifically for a MANET, which has taken a clue from *Ant Colony Optimization* (ACO) algorithm [3] and its combination with queuing network analysis. 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 that are not communicating with each other is called *Stigmergy*[4]. 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. 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. In this paper, I present the ACO principle in a unifying framework which includes queuing network analysis to evaluate End-to-End packet delay in MANET.

2. LITERATURE SURVEY AND PREVIOUS WORK
In the last years, a great deal of literature has been published in the field of mobile ad hoc networks. Examples might include delay of the network, link capacity, link stability or identifying low mobility nodes. There exists relatively little work with regard to biologically inspired algorithms for routing in communications networks. However, there are a number of algorithms which show that these concepts can provide a significant performance gain over traditional approaches. A brief review of these algorithms is given in this section.

2.1 Ant-Based Control Algorithm
One of the earliest work on swarm intelligent routing is done by Schoonderwoerd et al. on the *Ant Based Control* (ABC) algorithm [Schoonderwoerd et al.,1996][5] This algorithm is applied for a wired circuit switched network, such as a

telephony network. ABC was later modeled analytically in [Subramanian et al., 1997][6] The algorithm is adaptive and exhibits robustness under various network conditions. The problem with this algorithm is that it updates the table probabilities when the ants move forward. The underlying assumption is that the network is symmetric, which is not always true. This issue is discussed by the next algorithm, called *AntNet*

2.2 AntNet Algorithm
DiCaro and Dorigo introduced an adaptive routing algorithm based on the adaptive learning and ant colonies called AntNet, for routing in packet-switching networks [DiCaro and Dorigo, 1997]. [7]This algorithm explores the network with the goal of building (and rebuilding) routing tables and keeping them adapted to traffic conditions. It is presented as a distributed, scalable and responsive algorithm for routing in wired networks. The algorithm is improved upon in [Barán and Sosa, 2000].[8]

2.3 Mobile Ants Based Routing
Mobile Ants Based Routing (MABR) algorithm is inspired by social insects and introduced as the first routing algorithm for large-scale mobile ad hoc networks MANETs [Heissenbüttel and Braun 2003].[9]. The algorithm consists of three layers:Topology Abstraction Protocol (TAP).

2.4 Ant-Colony Based Routing Algorithm
The Ant-Colony Based Routing (ARA) algorithm presents a detailed routing scheme for MANETs [Günes et al., 2002] [10]. It is similarly constructed as many other routing approaches and consists of three phases: route discovery, route maintenance, route failure handling

2.5 Termite
Termite is a distributed routing algorithm for mobile wireless ad hoc networks [Roth and Wicker, 2003]. [11]It is designed using the swarm intelligence framework in order to achieve better adaptivity, lower control overhead and lower per-node computation. The algorithm is inspired by the hill building behavior of termite. In part, this is required in order to maintain current pheromone through the network, but this behavior can also generate significant resource inefficiencies. AntNet is a related algorithm, although its forward/backward ant feature is not used. Termite and ARA share nearly the same data routing mechanism, although they differ significantly with regards to route discovery and failure recovery.

2.6 Ants Routing with Routing History in Dynamic Networks
Ants Routing with routing History (ARH) algorithm is a stochastic routing algorithm that chooses a suitable route and efficiently acquires the information of a route, when the nodes communicate man-to-man in ad hoc communication environments [12] .

2.7 Probabilistic Emergent Routing Algorithm
A Probabilistic Emergent Routing Algorithm for mobile ad hoc networks (PERA) which was presented in [Baras and Mehta, 2003][13] is a routing algorithm for mobile ad hoc networks based on the swarm intelligence paradigm and similar to the swarm intelligence algorithms described in [DiCaro and Dorigo, 1997] and [Subramanian et al., 1997].[5] The algorithm uses three kinds of agents - *regular forward ants*, *uniform forward ants* and *backward ants*. The algorithm assumes bidirectional links in the network and that all the nodes in the network fully cooperate in the operation of the algorithm.

2.8 Adaptive Swarm-based Routing (ASR) algorithm
Adaptive Swarm-based Routing (ASR) algorithm increases the speed of convergence and has quite good stability by using a novel variation of reinforcement learning and a technique called *momentum* [Lü et al., 2004][14] It is a very effective mechanism that seeks to incorporate a memory in the learning process, increases the stability of the scheme and helps to increase the learning efficiency of the network.

3. SIMULATED ANT ROUTING ALGORITHM
In this section, we would describe in details the proposed framework for routing algorithm based on *ACO algorithms* and *Kleinrock's delay analysis* to find the best route with minimum End-to-End packet delay in a MANET.

The routing algorithm is based on a type of learning algorithm, similar to one described in AntNet, that provides deterministic forwarding of message packets (ant packets) from source node to destination node. Our algorithm assumes that all links in the network are bi-directional and all the nodes in the network fully cooperate in the operation of the algorithm. This algorithm has proposed many parameters which are simulated in matlab environment.

Kleinrock [Kleinrock, 1964] [Wong, 1978] first derived an expression for the mean End-to- End delay in a message-switched network. He chose to model a data network as a network of communication channels whose purpose was to move data messages from their origin to their destination.[15] The expected response time delay to send packets from a source node S to a destination node D is the sum over the response times at all links and nodes visited along the way [Haverkort 1998][16]

3.1 Parameter Initialization
We must define at the beginning of our simulation process the *arrival rate* λ, which is the average number of packets arrived each unit of time. Another parameter to be defined is the *service rate* μij (Bandwidth) between two nodes i and j in the network, which is the average number of packets serviced in each time unit. This parameter μij has be given in a matrix called service rate table

The work reported here follows the simulation of routing in ad hoc networks using AntNet algorithm [17], and is informally summarized as follows:

- Each node in the network retains a record of packet destinations as seen on data packets passing through that node. This is used to periodically, but asynchronously, launch 'forward' ants with destinations stochastically sampled from the collected set of destinations.
- Once launched, a forward ant uses the routing table information to make probabilistic decisions regarding the next hop to take at each node. While moving, each forward ant collects time stamp and node information in a stack, which is later used to update the routing tables along the path followed. The trip time to reach the desired neighbor is computed using this simple formula:

$$dij + (qij + Sa)/ Bij \qquad (1)$$

Where *dij* is the link's propagation delay (distance/signal propagation speed) between two mobile nodes *i* and *j*. Note that this value can be neglected because the distance value is very small in comparison to the value of signal propagation speed.

- The number of data packets waiting in the queue between nodes i and j is qij and is calculated by using the *M*/*M*/1 equation

$$q_{ij} = \rho_{ij}^2/1 - \rho_{ij},$$

$$\rho_{ij} = \lambda_j X \rho_{ij}/\mu_j$$

is the utilization of the link between two nodes *i* and *j*, and λj and μj are the arrival rate and service rate at node j respectively. *Pij* is the probability of routing from node *i* to *j*. *Sa* is the size of the ant packet, and *Bij* is the bandwidth of the link between two nodes *i* and *j*. Bandwidth is here defined as the amount of data that can be transmitted in a fixed amount of time, expressed in bits per second (bps).

- Following equation represents the time delay for links. We suppose that the node delay is also determined by *M*/*M*/1 formulas for all nodes;

$$E[R(S,D)]=\sum E[R_{ij}] + \sum E[R_n]$$

where

$$E[R_{ij}] = 1/\mu C_{ij} - \lambda_{ij}$$

is the expected delay at all links, μcij is the number of packets that can be transmitted over a link *ij* (packets/sec.), and λij is the arrival rate over a link *ij*.

$$E[R_n] = \frac{\rho_n E[S_n]}{1-\rho_n} + E[S_n]$$

is the expected delay for node *i* with $E[Si]=1/\mu_n$ and $\rho_n = \lambda_n/\mu_n$

• If a forward ant re-encounters a node previously visited before reaching the destination, it is killed (in other words, identification of a loop in the path).

• On successfully reaching the destination node, the total link delay is calculated, and the forward ant is converted into a backward ant;

• The backward ant returns to the source using exactly the same route recorded by the forward ant. Instead of using the data packet queues, the backward ant uses priority queues;
• At each node visited by the backward ant, the corresponding entries are recorded to reflect the relative performance of the path.

Given a starting topology, then its delay is calculated. After applying the ant algorithm on the current topology, the values of delay are altered a little bit and then, the delay is recalculated. If the new delay is better, it definitely becomes the starting point for the next ant and each forward ant will move on the new path. If the delay is worse each forward ant will move on the old path. This has the effect that, nearly every new solution is adopted, while over time it becomes more and more likely that only better solutions are accepted.

4. SIMULATION APPROACH AND PARAMETERS

In this paper, the simulation tool used is MATLAB 7.5. Matlab's built in functions provide excellent tools for linear algebra computations, data analysis, signal processing, optimization, numerical solution of ordinary differential equations, quadrature and many other types of scientific computations.

Table1. Simulation Parameters

PARAMETER	VALUE
No of nodes	10,20,30,………………..100
Arrival rate	100 kbps
Link Bandwidth	15mbps
Routing Protocol	Ant algorithm + delay analysis by Kleinrock

5. RESULTS AND DISCUSSIONS

In this section we evaluate the End-to-End delay for all the parameters one by one such as for

- no of packets and nodes ,
- arrival rate,
- service rate (bandwidth),
- utilization or efficiency of link ,
- Throughput through the MANET using the Matlab framework

It has been observed in figure 1, the link delay varies linearly up to the number of packets to be 70 but as the number of packets increase beyond 70, the link delay rises exponentially because more the number of packets, more will be the traffic which will introduce more congestion and gives rise to the expected link delay.

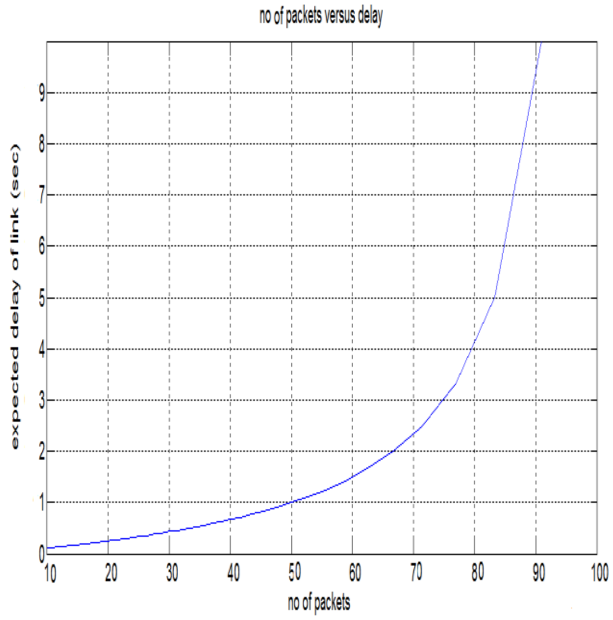

Figure 1: Delay of link against no of packets

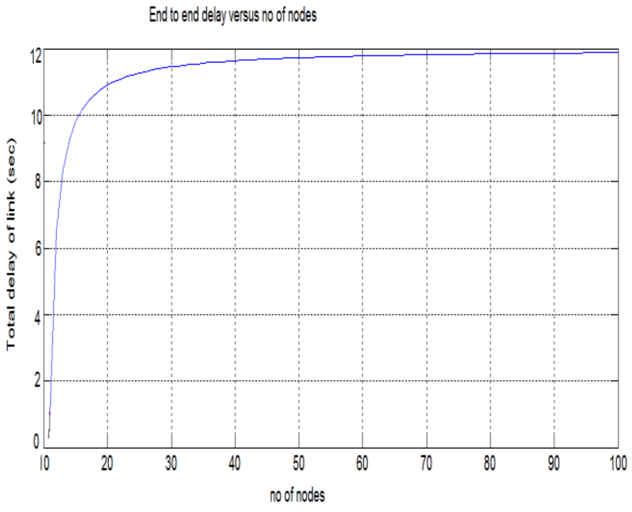

Figure 1: Total delay of link against no of node using ant routing algorithm

In figure 2, the total end to end delay is measured against number of nodes using simulated ant routing algorithm. In most of other systems, delay rises as number of nodes increases, but using our algorithm the link delay rises initially exponentially and once the nodes start learning using ant routing algorithm and become able to find the shortest path with the help of pheromone deposited, the delay become constant. Therefore, with the implementation of this algorithm in the real world networks, we would be able to saturate link delay for huge networks consisting of very large number of nodes.

In figure 3, the service time delay is measured and plotted against the service rate also known as bandwidth (mbps), which in fact is the average number of packets serviced in each time unit. It is observed that as the service rate improves, service time delay initially reduces exponentially and finally reduces linearly as the service rate improves further.

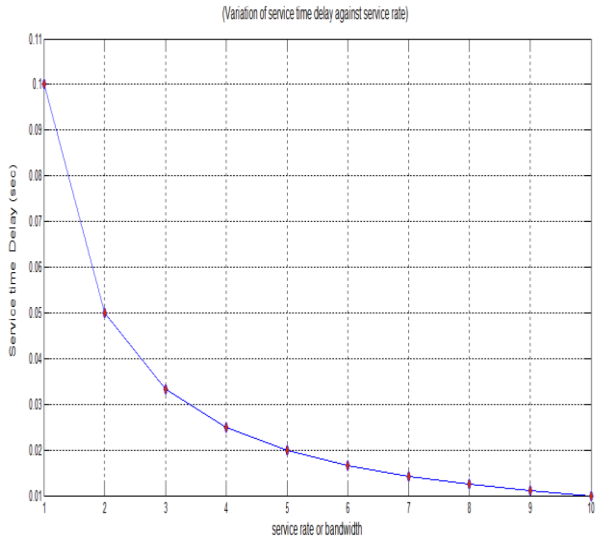

Figure 2: Variation of service time delay against service rate or bandwidth

In figure 4, the characteristics have been plotted between the arrival rate (which is the average number of packets arrived in each unit of time) and the link delay for the different values of µ; It has been observed that as the value of service rate i.e. the number of packets successfully transmitted increases, the value of delay decreases with the increase in arrival rate.

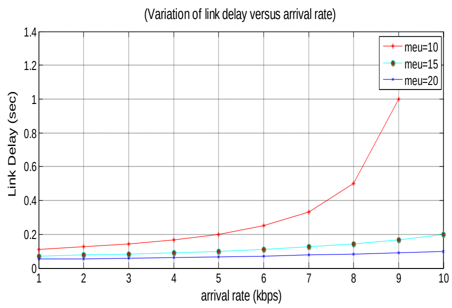

Figure 3: Link delay measured against arrival rate for different values of service rate

Figure 5 shows the utilization i.e. how efficiently the link is being used, against delay for varying number of nodes. For smaller number of nodes i.e. of the order of 10, link delay almost remains constant even if the utilization of the link improves. As we increase the number of nodes in a link, the utilization is much improved which is on account of reducing the link delay from higher values to very small values. This shows the strength of the proposed algorithm for huge to gigantic networks, where link delay can be minimal when compared with other routing algorithms.

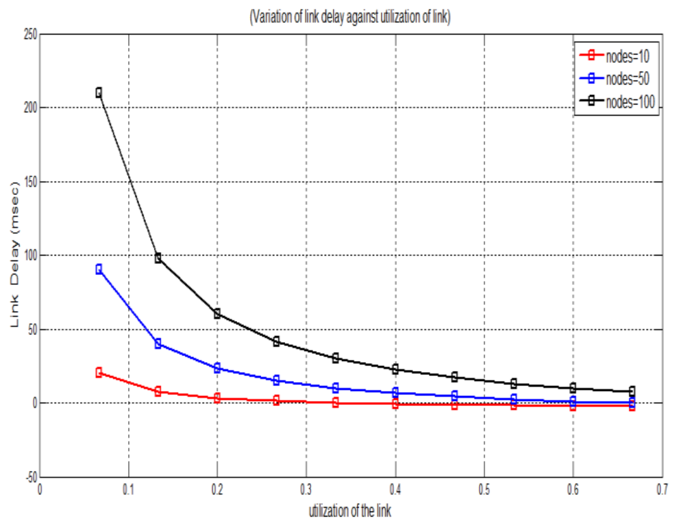

Figure 4: Delay of link measured against utilization of efficiency of the link for varying no of nodes

It has been observed in figure 6, that the throughput of the link improves as the bandwidth increases from 10 to 15 mbps which further correlates with significant reduction in link delay

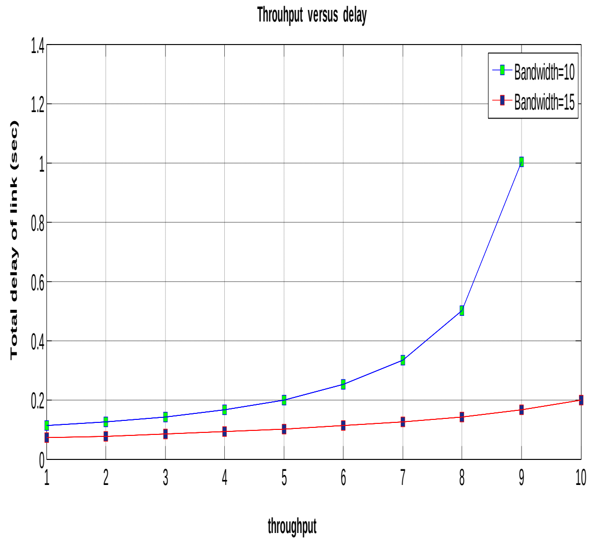

Figure 5: Total Delay of link against throughput

6. CONCLUSION

An Improved ant routing algorithm coupled with linear programming approach for mobile ad-hoc networks has been simulated using Matlab and results have been presented together with simulation framework. The simulation results predict that the modified ant algorithm has been able to cope with all dynamic networks, in particular its ability to improve the link performance after curtailing the link delay to acceptable limits in simulation environment. The same performance can also be produced practically for huge networks containing large number of nodes. It is observed and presented that the total end to end delay of the link is significantly reduced with regard to utilization of the link, bandwidth availability and arrival rate. The proposed algorithm has been able to achieve optimum route performance characteristics with reduced link delay in MANETs and certainly holds an edge over the existing algorithms. The efficiency of the link also improves as the reduction of delay happens and therefore, various link performance parameters of MANET gets improved. In particular the comparison with similar routing algorithms will be our main focus for future work.

7. REFERENCES

[1] I. Chlamtac, M. Conti, and J. Liu, “Mobile ad hoc networking: imperatives and challenges”, Ad Hoc Networks, No. 1, 2003.

[2] M. Elizabeth, and T. Chai-Keong, “A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks”, IEEE Personal Communications, 1999.

[3] Deepak Bansal, Ravinder Singh Sawhney and Ankur Bansal, “Routing Metrics Improvisation in Wireless Mobile Networks Using Ant Colony Optimization” International Conference on Recent Advances and Future Trends in Information Technology (iRAFIT2012) Proceedings published in International Journal of Computer Applications® (IJCA)

[4] D. Corne, M. Dorigo, and F. Glover (Eds.), “New ideas in optimization”, Maidenhead, UK: McGraw-Hill, 1999.

[5] R. Schoonderwoerd, O. Holland, J. Bruten, and L. Rothkrantz, “Ants for Load Balancing in Telecommunications Networks”, Adaptive Behavior, Vol. 5, No. 2, 1997, pp. 169-207.

[6] D. Subramanian, P. Druschel, J. Chen.” Ants and Reinforcement Learning: A Case Study in Routing in Dynamic Networks”. *Proceedings of the International Joint Conference on Artificial Intelligence*, 1997.

[7] G. Di Caro, and M. Dorigo, “AntNet: A mobile agents approach to adaptive routing”, Technical report TR-97-12. Université Libre de Bruxelles, IRIDIA, 1997.

[8] B. Barán, R. Sosa. “A New Approach for AntNet Routing, Ninth International Conference on Computer Communications and Networks IEEE ICCCN-2000,” Las Vegas - Estados Unidos, 2000.

[9] M. Heissenbüttel, and T. Braun, “Ants-Based Routing in Large Scale Mobile Ad-Hoc Networks”, Kommunikation in Verteilten Systemen (KiVS), 2003.

[10] 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, 2002, pp.79-85.

[11] M. Roth, S. Wicker. Termite: “Emergent Ad-Hoc Networking”, The Second Mediterranean Workshop on Ad-Hoc Networks, 2003.

[12] K. Fujita, A. Saito, T. Matsui, “An Adaptive Ant-Based Routing Algorithm used Routing History in Dynamic Networks.” *4th Asia-Pacific Conference on Simulated*

Evolution And Learning (SEAL'02). *IPSJ SIGNotes MoBiLe computing and wireless communications*, Abstract No.020 – 040. 2002.

[13] J.S. Baras, H. Mehta. "A Probabilistic Emergent Routing Algorithm for Mobile Ad Hoc Networks." *WiOpt03 Proc. of Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, 2003.

[14] Y. Lü, G. Zhao, F. Su, and X. Li. "Adaptive Swarm-Based Routing in Communication Networks". *Journal of Zhejiang University Science JZUS* 2004, vol. 5(7), pages 867-872, 2004

[15] L. Kleinrock. Communication Nets: Stochastic Message Flow and Delays. McGraw-Hill, New York, 1964.

[16] B. Haverkort. Performance of Computer Communication Systems, "A Model-Based Approach", John Wiley & Sons, Ltd., 1998.

[17] Tarek H. Ahmed, "Simulation of Mobility and Routing in Ad Hoc Networks using Ant Colony Algorithms" Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'05) IEEE