

HAQR: THE HIERARCHICAL ANT BASED QoS AWARE ON-DEMAND ROUTING FOR MANET'S

Debajit Sensarma¹ and Koushik Majumder¹

¹ Department of Computer Science & Engineering, West Bengal University of Technology, Kolkata, INDIA

debajit.sensarma2008@gmail.com koushik@ieee.org

ABSTRACT

A Mobile Ad Hoc Network (MANET) is a collection of wireless mobile devices with no pre existing infrastructure or centralized control. Supporting QoS during routing is a very challenging task. Clustering is an effective method for resource management regarding network performance, routing protocol design, QoS etc. In real time various types of nodes with different computing and transmission power, different rolls and different mobility pattern may exist. Hierarchical routing provides routing through this kind of heterogeneous nodes. In this paper, HAQR, a novel ant based QoS aware routing is proposed on a three level hierarchical cluster based topology in MANET which will be more scalable and efficient compared to flat architecture and will give better throughput.

KEYWORDS

MANET, Ant Colony Optimization, Hierarchical Routing, QoS Routing.

1. INTRODUCTION

A Mobile Ad Hoc network (MANET) [1] is a dynamically formed wireless network by the mobile nodes. In this network nodes can move randomly. Due to the node mobility network is not stable and topology is not fixed. So, stable routing is a very essential part in MANET. It is possible to form a cluster of nodes which produces communication hierarchy. There are some advantages of this type of hierarchical routing. Many of the contemporary ad hoc networks are heterogeneous in nature. Mobile devices of this kind of network are equipped with different communication capabilities with respect to frequency band, battery power, radio range, data rate etc. E.g. in military networks, soldiers, tanks and command posts works in different interfaces. Hierarchical routing makes the protocol more scalable. Furthermore, control overhead, routing overhead decreases with the hierarchical routing scheme in comparison with flat routing and this type of routing keeps the routing table size smaller in comparison with flat routing scheme. Failure is isolated in hierarchical network topology. So, overhead of route maintenance also decreases.

In this paper, clustering technique is used to organize mobile nodes in small cluster to provide a hierarchical network structure of heterogeneous mobile nodes in MANET. This algorithm takes advantages of Ant Colony Optimization (ACO) [3]. This proposed routing scheme decreases the overhead and improves the overall performance of MANET.

The paper is organized as follows: Section 2 describes the related works. In section 3 assumptions of this algorithm is given. Section 4 describes the design of the proposed protocol. Section 5 explains the proposed HAQR protocol. Finally section 6 concludes the paper.

2. RELATED WORKS

Several cluster based hierarchical routing protocols has been proposed. In [4] a two layer heterogeneous network has been proposed. It provides the internet access to the lower layer nodes but it does not provide communication between lower layer nodes. The upper layer nodes are gateways to provide the internet access to the lower layer nodes. In [5], an end-to-end QoS aware routing in physically hierarchical Ad hoc network has been proposed. It considered the QoS metrics bandwidth and node speed but other QoS metrics does not taken care of. In [6], HAODV is proposed which is an extension of AODV. Here nodes are heterogeneous in nature and operates on Wi-Fi or Bluetooth. It is a reactive routing and the advantage it helps to reduce the control overhead. But the disadvantage there can be lacks of scalability as every time number of nodes increases, new route discovery initiated. In [7], a dynamic adaptive routing protocol (DARP) has been proposed. It is same as [6] and suffers from lacks of scalability. In [8], a weight based adaptive clustering has been proposed. It tries to improve routing in heterogeneous MANET using Global Positioning System (GPS). So, by knowing the user mobility pattern accurately, the routing efficiency is increased. But when no GPS is available it is not very useful. In [9], a heterogeneous routing protocol based stable routing has been proposed. It is a hybrid cluster based routing protocol and combines AODV and DSDV for intra cluster and inters cluster routing. But disadvantage is there can be a delay in AODV route discovery process.

3. ASSUMPTIONS

The following assumptions are taken into consideration to construct the protocol:

1. Here, a three level hierarchical cluster model is considered.
2. Ant Colony Optimization is used here for cluster head selection. Cluster heads are selected based upon the weight value of the nodes. In each layer same procedure is followed. Each cluster head can only have one hop neighbours and neighbours can communicate with each other.
3. A node can communicate maximum with three levels based on the transmission power and battery power. Level-0 nodes can communicate with only level-0 nodes. Level-1 nodes can communicate with level-1 and level-0 nodes and level-2 nodes can communicate with all three level nodes.
4. Here, hierarchical addressing is used for each node and based on the transmission power nodes are represented with different shapes. Details of this presentation are depicted in section 5.1.
5. In each level QoS constrains are taken care of and Ant Colony Optimization is used for both intra and inter cluster routing.

4. DESIGN OF PROPOSED PROTOCOL

4.1 LOGICAL TOPOLOGY LEVELS

Fig. 2 illustrates the network architecture of the proposed routing protocol. The architecture is designed as follows- Each clusters are leveled with 'C' followed by a topology level at which the cluster is formed, followed by cluster head id of that cluster. For example, C0.A1 denotes level-0 cluster having A1 as cluster head. The nodes having single interface are denoted by white circles

and represented by single digit (1, 2, 3 etc.). These nodes are formed only in bottom level. Multiple interface nodes are represented by node's name (e.g. A1, A2 etc.) followed by a digit denoting the node's interface where it operates. Nodes with interfaces indicated by triangles can operate in three levels (e.g. A1.2, A1.1, and A1.0) and nodes denoted by dark circles can communicate with lower levels (e.g. A3.1, A3.0).

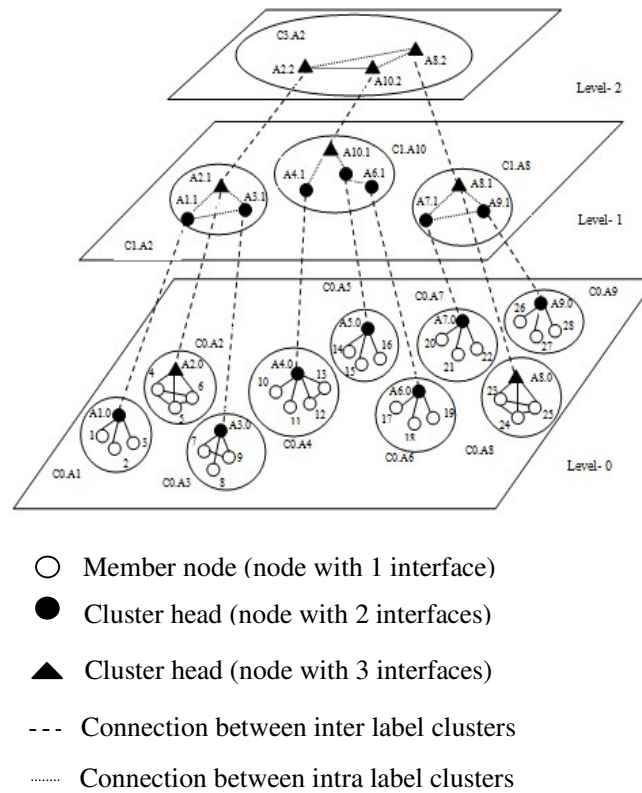


Fig.2. Architecture of proposed three layer hierarchical structure

4.2 CLUSTER HEAD SELECTION AND CLUSTER FORMATION

In cluster head election process for each cluster, Cluster head is selected based on the weight function, pheromone value, probability. According to [10] cluster head can be selected based on combined weight metric of the node. Cluster head is selected with weight factors: Battery power, Node Connectivity, Node Mobility and Distance. The algorithm is described as follows:

Algorithm 1: Weight Calculation Algorithm.

Begin

1. Find connectivity 'c' for each node which is the number of neighbours of each node. Find the remaining energy, 'e' for each node.
2. Compute the mobility m for each node which is the running.
3. Compute the sum of distances d with all its neighbours for each node.
4. Calculate the combined weights $weight_i$ as

$$\text{weight}_i = w1*c_i + w2*e_i - w3*m_i + w4*d_i$$

Here $w1+w2+w3+w4=1$.

c_i = Connectivity of node i. i.e. number of nodes that can directly communicate with the given node within the transmission range of each other.

e_i = Battery power of node i. i.e. remaining power of that node.

m_i = Mobility of node i.

d_i = Distance of each node from i. i.e. sum of distance of the node from all its neighbours.

End

Algorithm 2: Cluster Head Selection Algorithm.

Begin

1. Each node finds its neighbours and builds its neighbourhood table.
2. A set of clusters are formed from nodes and its neighbours.
3. Each node calculates its weight by calling the weight calculation algorithm given above.
4. At first, in a cluster, a node is selected as a cluster head randomly.
5. In each iteration, a node is selected as a cluster head based on probability of the neighbour nodes. The probability of each node to be selected as cluster head is:

$$P_i = \frac{\tau_i}{\sum_{k=1}^n \tau_k} \quad (1)$$

6. Each time a node is selected as the cluster head, the pheromone value is updated according to the following formula:

$$\tau_i = (1 - \rho) \tau_i + \rho (\text{weight}_i) \quad (2)$$

Here ρ is the pheromone evaporation factor ($0 < \rho < 1$).

7. Continue step 5-6 for each node in the cluster until all the nodes in the network are covered (i.e. a node is a cluster head or falls within the range of an existing cluster head).
8. The node, whose weight value and the pheromone value is greater than its neighbour and greater than threshold, is selected as cluster head and it sends message to all its neighbours.
9. On receiving a message, all neighbour nodes unicast an acknowledgment message to the selected cluster head.
10. For each cluster continue the steps 3-9.

End

5. PROPOSED HAQR ALGORITHM

This is a three level hierarchical cluster based routing scheme which utilizes the Ant Colony Optimization. This is also a QoS aware routing with parameters: delay, bandwidth, energy and link expiration time. It takes care of QoS constraints in each layer. Here 5 kinds of packets are used. Route_Ant is used by a node when it wants to know if the destination is a member of its cluster head or not. Request_Knave_Ant and Reply_Knave_Ant are used for intra cluster routing. Request_King_Ant and Reply_King_Ant are used for inter cluster routing. This algorithm has two phases: Route discovery phase and route maintenance phase.

5.1. MATHEMATICAL MODEL

For mathematical analysis MANET is represented by a connected undirected graph. Let $G(V, E)$ represents the mobile ad hoc network. Here V denotes the set of network nodes and E denotes the set of bidirectional links. QoS metrics with respect to each link $e \in E$ is delay (e), bandwidth (e), link expiration time (e). With respected to node $n \in V$, it is delay (n), energy (n). Another QoS metric considered here is hop count. It is important because multiple hops are used for data transmission in MANET. So, it is necessary to find paths with minimum hops. The main motivation of this proposed algorithm is to find path from source to destination which will satisfy the QoS requirements such as delay, bandwidth, energy, link expiration time.

Let, path (i, j) or R is entire path from node i to j where QoS constraints have to satisfied. From an arbitrary node i to an arbitrary node j , delay, bandwidth, energy, link expiration time and hop count is calculated as-

$$\text{delay (path (i, j)) or } D(R) = \sum_{e \in P(i, j)} \text{delay}(e) + \sum_{n \in P(i, j)} \text{delay}(n)$$

where, delay (e) is the transmission and propagation delay of the path(i, j) and delay (n) is the processing and queuing delay of node 'n' on path(i, j).

$$\text{bandwidth(path(i,j)) or } B(R) = \min_{e \in P(i, j)} \{ \text{bandwidth}(e) \}$$

where, bandwidth (e) is the available bandwidth of that link on path(i, j).

$$\text{link expiration time or } T(R) = \min_{e \in P(i, j)} \{ \text{link expiration time (e)} \}$$

where, link expiration time (e) is the expiration time of a link in route R .

$$\text{energy (path (i, j)) or } E(R) = \min_{n \in P(i, j)} \{ \text{energy (n)} \}$$

where, energy (n) is the residual energy of node 'n' on path(i, j).

hop count (path (i, j)) or $HC(R) =$ Number of nodes in the path.

5.2. CALCULATION OF PHEROMONE

Ant deposits pheromone during traversal of the link for finding a route. The quantity of pheromone it deposited on each link (i, j) along the route R is noted by $\Delta\tau_{i, j}$ and it is a function of global quality of route R . It is expressed by the following equation-

$$\Delta\tau_{i, j} = \frac{B(R)^{\lambda_B} + E(R)^{\lambda_E} + T(R)^{\lambda_T}}{D(R)^{\lambda_D} + HC(R)^{\lambda_{HC}}} \quad (3)$$

Here $\lambda_B, \lambda_E, \lambda_D, \lambda_{HC}$ and λ_T are the weight factors which indicate the relative significance of the QoS parameters during pheromone update on path (i, j). The quantity of the deposited pheromone is defined only after finding the route.

5.3. CALCULATION OF PATH PREFERENCE PROBABILITY

Path Preference Probability is calculated in each intermediate node as well as source node upon receiving of Reply Knave_Ant or Reply King_Ant.

Suppose current node i receives Reply Knave_Ant or Reply King_Ant from node j for destination d , then the Path Preference Probability is calculated as-

$$P_{ijd} = \frac{[\tau_{ij}]^{\alpha_1} \cdot [D_{ijd}]^{\alpha_2} \cdot [\eta_{ijd}]^{\alpha_3} \cdot [B_{ijd}]^{\alpha_4} \cdot [E_{ijd}]^{\alpha_5} [T_{ijd}]^{\alpha_6}}{\sum_{k \in N_i} [\tau_{ik}]^{\alpha_1} \cdot [D_{ikd}]^{\alpha_2} \cdot [\eta_{ikd}]^{\alpha_3} \cdot [B_{ikd}]^{\alpha_4} \cdot [E_{ikd}]^{\alpha_5} [T_{ikd}]^{\alpha_6}} \quad (4)$$

Here α_1 , α_2 , α_3 , α_4 , α_5 and α_6 are the tunable parameters which control the relative weights of pheromone trails, hop count, bandwidth, energy and link expiration time respectively.

N_i is the set of neighbors of i and k is the neighbor node of i through which a path to destination is known.

The relative metrics are calculated from source i to destination d via j as-

$$D_{ijd} = \frac{1}{\text{delay}(\text{path}(i, d))}$$

$$\eta_{ijd} = \frac{1}{\text{hopcount}(\text{path}(i, d))}$$

$$B_{ijd} = \text{bandwidth}(\text{path}(i, d))$$

$$E_{ijd} = \text{energy}(\text{path}(i, d))$$

$$T_{ijd} = \text{link expiration time}(\text{path}(i, d))$$

Now, source as well as neighbors has multiple paths from source to destination. The path with higher Path Preference Probability is selected for the data transmission.

5.4. ROUTING IN PROPOSED HIERARCHICAL NETWORK

Algorithm 3: Route Discovery Phase

BEGIN

Suppose Source S wants to communicate with destination D .

/* Intra-cluster routing */

Step 1: S searches in its neighbour table to see if D is a neighbour of S .

Step 2: If S finds that D is its own neighbour, then it directly sends data packets to D .

Step 3: If S finds that D is not its neighbour, it unicast the Route_Ant packet to its cluster head (i.e. $CH(S)$).

Step 4: $CH(S)$ searches in its member table to see if D is a member of it.

Step 5: If $CH(S)$ finds that D is its own member, then it unicast the Route_Ant to S by setting flag=1.

Step 6: After receiving Route_Ant from cluster head, S starts finding route to D which will satisfy the required QoS constrains delay, bandwidth, energy, link expiration time and hop count.

Step 7: It first consults with its route cache, if an unexpired route exists to D , and then the route with higher path preference probability is used for transmitting data.

Step 8: If cache contains no unexpired route, S initiates a Request Knave_Ant to the destination through all its neighbours.

Step 9: While travelling to the destination the Request Knave_Ant collects delay, bandwidth of each link and energy, link expiration time of each node.

Step 10: When the Request Knave_Ant reaches the destination it will converted to Reply Knave_Ant and forwarded towards the original source. It will take same path as the Request Knave_Ant but in reverse direction.

Step 11: For every Reply Knave_Ant reaching an intermediate node or source node, node can find the delay, bandwidth, energy and link expiration time to the destination and node calculate Path Preference Probability. If it is better than the requirements, and then the path is accepted and stored in the memory.

Step 12: The path with higher Path Preference Probability will be considered as best path and will be selected for transmitting data.

/ Inter-cluster routing in the same network or region */*

Step 13: IF D is not a member of CH(S), it will unicast Request_Ant to its cluster head (CH (CH(S))) in level-1 hierarchy.

Step 14: If CH (CH(S)) finds D in its own member table but not in its cluster, as well as D is in same network or region, then it will unicast the Route_Ant to the CH(S) and will set flag=1.

Step 15: After receiving Route_Ant from cluster head, CH(S) starts finding route to cluster head of D (i.e. CH (D)), which will also satisfies the QoS requirements.

Step 16: At first, it consults with its route cache, if unexpired paths exists to CH (D)), then the path with better Path Preference Probability is selected for data transmission.

Step 17: If cache contains no unexpired route, then CH(S) initiates a Request King_Ant to destination through all its neighbour.

Step 18: The Request King_Ant also collects the delay, bandwidth, energy, link expiration time and hop count of each links and each nodes respectively.

Step 19: When Request King_Ant reaches the destination cluster head, it will be converted to Reply King_Ant and also forwarded to the CH(S) by the same path as Request King_Ant but in reverse direction.

Step 20: Intermediate cluster nodes calculate the Path Preference Probability and if the Path Preference Probability is better than requirements, then it is stored in the node's cache.

Step 21: The path with higher Path Preference Probability is selected for data transmission and the data is transmitted directly to the destination through its cluster head (CH (D)).

Step 22: If CH (CH(S)) finds D in its own member tables the member of its cluster, then the communication starts directly through that cluster head [i.e. CH CH(S) have at least 2 interfaces].

/ Inter-cluster routing in the different network or region*/*

Step 23: If CH (CH(S)) finds D in its member table but D is in another network, then it will forward the Request Knave_Ant packet to the corresponding cluster member node with required QoS constraints in level-2. Then the member node unicast the request to the desired destination's cluster head and finally when the Request King_Ant packet reaches the destination, it sends the Reply King_Ant packet to the original source by the same path but in reverse order.

Step 24: If CH (CH(S)) does not find D in its member table, it unicast Request packet to cluster head (i.e. CH (CH (CH(S)))) and if D is in its member list, it sets the flag to 1 and sends to CH(CH(S)) and route discovery continues like previous process.

END

Algorithm 4: Route Maintenance Phase

There are three cases in maintenance: single node leaving the cluster, single node joining the cluster, the whole cluster moving together.

BEGIN

Maintenance of level-0 cluster.

Case 1: Leaving of a node in level-0 cluster.

Cluster head periodically will send the beacon to the member node and the member nodes send acknowledgement to the cluster head. If a cluster head does not receive any reply from its member within a specific period of time, then it can detect leaving of member from the cluster. Again, if the cluster member cannot receive any beacon from the cluster head, then it can detect leaving of its cluster head from the cluster.

Case 1.1: If an internal node moves out of a cluster, then the cluster head deletes the member node entry from the member table.

Case 1.2: If a cluster head moves out of a cluster, nodes that do not belong to any cluster started the cluster head selection procedure.

Case 2: Joining of a node in level-0 cluster.

When a cluster head starts receiving an acknowledgement from a new node, then it adds this new member node to the member table.

Case 3: The whole cluster moving together.

When a cluster head comes in the transmission range of another cluster head at level-0, then a new cluster head selection procedure started.

Maintenance of level-1 cluster.

Case 4: Level-1 change caused by one node leaving level-0 cluster.

Case 4.1: Leaving node is the level-1 cluster head.

When level-0 cluster head cannot get any beacon from its cluster head, then a new cluster head selection procedure started in level-1 as well as in level-0 cluster.

Case 4.2: Leaving node is not the level-1 cluster head.

When level-1 cluster head cannot find any beacon from a member node which is a cluster head of level-0 cluster, it deletes its corresponding node entry from its member table and a new cluster selection procedure starts at level-0.

Case 5: Level-1 change caused by new cluster head joining in level-0 cluster.

When a new cluster head selected in level-0, it is added to the member table of the cluster head of level-1 which is in the same network or region.

Maintenance of level-2 cluster.

Case 6: Level-2 change caused by one node leaving level-1 cluster.

Case 6.1: Leaving node is the level-2 cluster head.

In this case a new cluster head selection procedure is invoked. Here, if the leaving node has two interfaces (i.e. level-2 and level-1), then new cluster head selection procedure is occur in level-2 cluster and level-1. But if it has 3 interfaces, then new cluster head selection procedure occurs in three levels.

Case 6.2: Leaving node is not the level-2 cluster head.

In this case, the corresponding node entry is removed from the cluster head member table. Here, if the leaving node has two interfaces (i.e. level-2 and level-1), then new cluster head selection procedure is occur only in level-1 cluster and if it has 3 interfaces, then new cluster head selection procedure occurs in level-2 and level-1 cluster.

Case 7: Level-2 change caused by new cluster head joining in level-1 cluster.

When a new cluster head selected in level-1, it is added to the member table of the cluster head of level-2 cluster.

END

6. CONCLUSION

As MANET is dynamic in nature, so QoS provisioning is very difficult in this network. In this paper we proposed an Ant based hierarchical on-demand routing. It is a three level cluster based routing algorithm. It takes the advantages of both the ant colony optimization and cluster architecture. It is also a power efficient routing. Node's transmission power plays a very crucial role for increasing routing stability. Unlike other routings, QoS constrains are taken care of each layer. So, it is very efficient for real time communication with heterogeneous nature of nodes. Scalability also increases in this routing scheme. Here, an efficient cluster formation is used for handling the increased number of nodes. Besides this, it emphasize on cluster maintenance for reducing the overhead and delays of the network.

In future we will simulate this protocol and will compare it with other hierarchical routing algorithms. Also we will take new QoS metrics to provide better throughput for real time communication among the heterogeneous nature of nodes.

ACKNOWLEDGEMENTS

The authors would like to thank West Bengal University Technology, West Bengal, India for the supports and facilities provided to carry out this research. The authors also thank the reviewers for their constructive and helpful comments.

REFERENCES

- [1] J., Macker, S., Corsen, IETF Mobile Ad Hoc Networks (MANET) Working Group Charter, <http://www.ietf.org/html.charters/manet-charter.html>.
- [2] K. L., Pang (2006), The Comparison Study of Flat Routing and Hierarchical Routing in Ad Hoc Wireless Networks, ICON '06. 14th IEEE International Conference, vol. 1.
- [3] D., Sensarma, K., Majumder, (2012), "A Comparative Analysis of the Ant Based Systems for QoS Routing in MANET," Recent Trends in Computer Networks and Distributed Systems Security, Communications in Computer and Information Science, Vol. 335, pp. 485-496.
- [4] C.F., Huang, H.W., Lee, Y.C., Tseng (2004), A two-tier heterogeneous mobile ad hoc network architecture and its load-balance routing problem, *Mob. Netw. Appl.*, vol. 9, no. 4, pp. 379-391.
- [5] K. Y., Lee, J., B., Hwang, J. D., Ryoo (2005), End-to-end QoS routing in physically hierarchical wireless ad-hoc networks, Vehicular Technology Conference.
- [6] H., Safa, H., Artail, M., Karam, H., Ollaic, and R., Abdallah (2007), HAODV: a New Routing Protocol to Support Interoperability in Heterogeneous MANET, *Computer Systems and Applications, AICCSA '07. IEEE/ACS International Conference on*, pp. 893-900, May 2007.
- [7] E. H. K., Wu, Y.Z., Huang (2004), Dynamic adaptive routing for a heterogeneous wireless network, *Mob. Netw. Appl.*, vol. 9, no. 3, pp. 219-233.
- [8] Y., Wang, H., Chen, X., Yang, D., Zhang (2007), Wachm: Weight based adaptive clustering for large scale heterogeneous manet, *Communications and Information Technologies, ISCIT '07. International Symposium on*, pp. 936-941, Oct. 2007.
- [9] K., Xu, M., Gerla (2002), A heterogeneous routing protocol based on a new stable clustering scheme, *MILCOM 2002 Proceedings*, vol. 2, pp.838-843 vol.2, Oct. 2002.
- [10] M., Chatterjee, S.K., Das, D., Turgut (2001), WCA: A Weighted Clustering Algorithm for Mobile Ad hoc Networks, *Journal of Cluster Computing (Special Issue on Mobile Ad hoc Networks)*.

Authors

Debajit Sensarma has received his B.Sc degree in Computer Science in the year 2009 from university of Calcutta, Kolkata, India and M.Sc degree in computer Science in the year 2011 from West Bengal State University, Kolkata, India. He obtained his M.Tech. degree in Computer Science and engineering from West Bengal University of Technology, Kolkata, India, in the year 2013. He is now pursuing his PhD degree from the department of Computer Science and Engineering, University of Calcutta, Kolkata. He has published several papers in International journals and conferences.



Koushik Majumder has received his B.Tech and M.Tech degrees in Computer Science and Engineering and Information Technology in the year 2003 and 2005 respectively from University of Calcutta, Kolkata, India. He obtained his PhD degree in the field of Mobile Ad Hoc Networking in 2012 from Jadavpur University, Kolkata, India. Before coming to the teaching profession he has worked in reputed international software organizations like Tata Consultancy Services and Cognizant Technology Solutions. He is presently working as an Assistant Professor in the Dept. of Computer Science & Engineering in West Bengal University of Technology, Kolkata, India He has published several papers in International and National level journals and conferences. He is a Senior Member, IEEE.

