

# A Hybrid Ant Colony Optimization Algorithm for Network Routing and Planning

**Mahendra Pratap Panigrahy**  
Associate Professor, ECE  
Institute of Technology Roorkee  
Haridwar, Uttarakhand, India

**Paramjeet Kaur**  
M.Tech. (CS) Student  
Radha Govind Engineering College, Meerut,  
Uttarpradesh

## **ABSTRACT:**

The mechanism of route search problem is applied to various engineering fields. The various researches for the dynamic routing only revile the shortest path from source to destination but not vice versa. Ant colony optimization (ACO) algorithms have proved to be able to adapt to dynamic optimization problems (DOPs) when stagnation behavior is avoided. Several approaches have been integrated with ACO to improve its performance for DOPs. The main objective of this work is to search out the least-time-cost route in a variable-edge-weight graph. It can find multiple transmission paths from sources to the goal by parallel search. We introduce time-dependent pheromones and feedback path model as two heuristic factors to improve the basic ACO. Finally, this proposed heuristic algorithm is verified to be steady-going by repeated testing [1]. Ant Colony Optimization Technique has been applied in different network models with different number of nodes and structure to find the shortest path with optimum throughput. The simulation results show that the proposed dynamic ACO algorithm can effectively reduce time cost by avoiding the dynamic congestion areas. The experimental results show that the algorithm is more effective than the existing ones and it improves the Quality of Service (QoS) [13][14][15].

**General Terms:** Route Optimization, Shortest path, Real-time

**Keywords:** NP-Complete, ACO, DOPs, Time-Dependent Pheromones, Feedback Path Model, Dynamic Congestion Areas

## **1. INTRODUCTION:**

Internet is the highly evolving towards the complex systems which comprises and integrates a number of wired and wireless networks covering the needs of different community of users. Many new complex interactive and real-time applications have been shown a typical challenge for the traditional transport network. To meet these requirements, optical networks have achieved enormous progress especially the enhancement of automatic resource discovery and dynamic bandwidth allocation capabilities. Traditional optical networks depend on the network management system to configure network resources and service statically which can not satisfy the increasing new service demands. Swarm Intelligence is one of the new approaches which is based on the natural phenomena and commonly known as Meta-Heuristics which is a new field in terms of its application to combinatorial optimization problems[2][3]. The term was first used in [Beni, 1988, Beni and Wang, 1989] to describe the coordination of robots. Since then, two main version of SI have been used for originating schemes to solve optimization problems. The Particle Swarm Optimization (PSO), and the other is, Swarm Intelligence (SI) is defined to include any scheme which employs the swarm

mechanics derived from the behavior of social insects, particularly ants, for solving optimization problems [Bonabeau et al., 1999]. The Ant Algorithms was introduced in [Dorigo, 1992] and after that, it has been gained a great success by applying it on both theoretical and practical optimization problems. Ant algorithms are an iterative, probabilistic meta-heuristic for finding solutions to combinatorial optimization problems. They are based on the foraging mechanism employed by real ants trying to find a short path from their nest to a food source. In foraging, the ants communicate circuitously via pheromone, which they use to stain their respective paths and which catch the attention of other ants. In the ant algorithm, artificial ants use virtual pheromone to update their path through the decision graph, i.e. the path that reflects which alternative an ant chooses at certain points. Ants of the later iterations use the pheromone inscription of previous good ants as a means of orientation when constructing their own solutions, which ultimately result in focusing the ants on promising parts of the search space. Ant algorithms provide adaptation, robustness and decentralized nature, which are appropriate for routing in modern communication networks.

## **2. ANT COLONY OPTIMIZATION:**

The Ant colony optimization is based on the foraging behavior of ants. When ants search for food, they wander randomly and upon finding food return to their colony while laying a chemical substance called pheromone. Many ants may travel through different routes to the same food source. Thus the ants, which travel the shortest path, reinforce the path with more pheromone that aids other ants to follow. Subsequently more ants are attracted by this pheromone trail, which reinforces the path even more. This autocatalytic behavior quickly identifies the shortest path. Ants are simple autonomous agents that interact via indirect communication known as stigmergic [4]. The agents, that is, the ants of the colony, act in a completely asynchronous, concurrent and distributed fashion. Each ant can be considered as autonomous, and the overall control is completely distributed. In this perspective, the colony realizes a form of concurrent computation. Multiple paths are repeatedly tried out back and forth and some information related to each followed path is released on the environment, which acts as a shared distributed memory encoded in the pheromone trails. In turn, the local content of this memory affects the stochastic decisions of the ants, such that, when there is a significant difference in the lengths of the possible paths, implicit path evaluation gets at work and, coupled with autocatalysis, results in a distributed and collective path optimization mechanism. Given enough time (depending on the number of ants, length and relative length difference of the paths, and other factors), this can result in the convergence of all the ants in the colony on the shortest among the possible paths each ant gives a contribution to the overall behavior. But, although a single ant is capable of building a “solution” (i.e., finding a path between its nest and a food reservoir), is only the simultaneous presence

and synergistic action of an ensemble of ants that makes possible the shortest path finding behavior (i.e., the convergence to the shortest path), which is a property of the colony and of the concurrent presence of all the discussed ingredients, and not a property of the single ant. Stigmergic is an indirect form of communication where individual agents leave signals in the environment and other agents sense them to drive their own behavior. This form of communication is local wherein simple agents interact locally without having any global information. The success of any stochastic search method heavily depends on striking an optimal balance between exploration and exploitation. These two issues are conflicting but very crucial for all the meta-heuristic algorithms. Exploitation is to effectively use the good solutions found in the past search whereas exploration is expanding the search to the unexplored areas of the search space for promising solutions. The reinforcement of the pheromone trail by the artificial ants exploits the good solution found in the past. The optimization procedure of ACO is explained using an example of which ACO is applied to TSP [5]. First, the random number  $q$  between from 0 to 1 is generated. Next,  $q$  is compared with benchmark  $q_0$ . When  $q$  is smaller than  $q_0$ , the city that has the largest value of the product of the static evaluation and the dynamic one is selected as the next destination. Otherwise, ant  $k$  in city  $i$  selects the move to city  $j$  according to probability  $p_k$  and it is defined as follows.

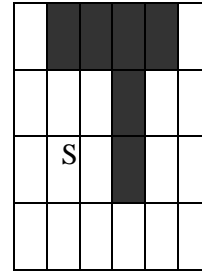
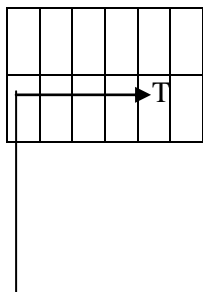
$$P^k(i,j) = \frac{[\tau(i,j)]^\alpha [\eta(i,j)]^\beta}{\sum_{l \in n_k} [\tau(i,l)]^\alpha [\eta(i,l)]^\beta} \quad (1)$$

Where,  $\tau(i,j)$  is a pheromone amount between city  $i$  and city  $j$ ,  $\eta(i,j)$  is a reciprocal of the distance between city  $i$  and city  $j$ ,  $\alpha, \beta$  is a parameter which controls the balance between static evaluation value and dynamic one, and  $n_k$  is a set of un-visit cities. Therefore, the selection probability is proportional to the product of the static evaluation and the dynamic one.

The main characteristics of the ACO is that, after each iteration, the pheromone values are updated by all the number of ants (packets) that have reached to the destination successfully and found a solution in the iteration itself. The pheromone value  $\tau(i,j)$  while traveling from node  $i$  to node  $j$  is updated as follows:

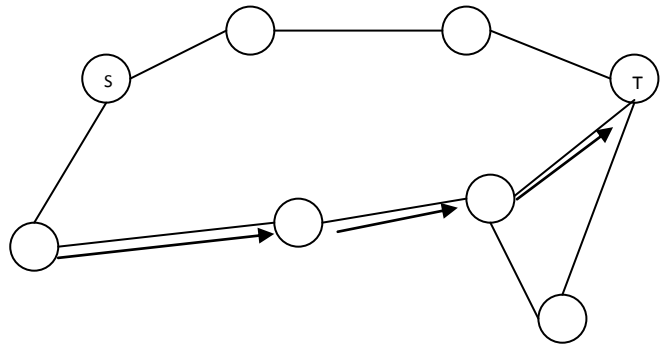
$$\tau(i,j) = (1-\rho) \cdot \tau(i,j) + \sum_{k=1}^m \Delta\tau(i,j)^k \quad (2)$$

Where  $\rho$  is the evaporation rate,  $m$  is the total number of successful ants i.e. data packet) and  $\Delta\tau(i,j)$  is the quantity of pheromone laid on edge  $(i,j)$  by packet  $k$ . Here the roulette wheel selection method has been used to select the next node in the path. Mathematically  $\Delta\tau(i,j)$  can be expressed as:



1. Grid

S=Start Point  
T= Target Point



2. Graph

Fig.1 Example of the shortest path problem

$$\Delta\tau(i,j) = \begin{cases} 1/L^+ & \text{if } (i,j) \in T^+ \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

Where,  $T^+$  is the best route and  $L^+$  is the distance of the best route.

## 2.1 ACO & NETWORK ROUTING:

The ACO is a best measure for the Network Routing because due to the following advantages.

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 paths in a less selfish way than in pure shortest path schemes favoring load balancing.
6. Showing limited sensitivity to parameter settings.

## 3. HYBRID ROUTING ALGORITHM:

The ACO hybrid algorithm for the Forward AntNet and the Backward AntNet is as follows:

### ForwardAnt Algorithm:

```

procedure ForwardAnt(source node, destination node)
  k ← source node;
  hops f w ← 0;
  
```

```

V [hops fw] ← k;
T [hops fw] ← 0;
while (k = destination node)
    tarrival ← get current time();
    n ← select next hope node(V, destination node, T k , Lk );
    wait on data link queue(k, n);
cross the link(k, n);
    Tk→n ← get current time() – tarrival ;
    k ← n;
    if (k ∈ V)
        hops cycle ← get cycle length(k, V );
        hops f w ← hops f w – hops cycle;
    else
        hops f w ← hops f w + 1;
V [hops fw] ← k;
T [hops fw] ← Tk→n ;
    end if
end while
become a backward ant(V, T );
end procedure

```

### **BackwardAnt Algorithm:**

```

procedure BackwardAnt(destination node, source node)
k ← destination node;
hops bw ← hops f w;
T ← 0;
while (k = source node)
    hops bw ← hops bw – 1;
    n ← V [hops bw];
    wait on high priority link queue(k, n);
    cross the link(k, n);
    k ← n;
    for (i ← hops bw + 1; i ≤ hops f w; i ← i + 1) /* UPDATES
FOR ALL SUB – PATHS*/
        δ ← V [i];
        Tk→δ ← T + T [i];
        T ← Tk→δ ;
        if (Tk→δ ≤ Isup (μδ , σδ ) ∨ δ ≡ d)
            kkMδ ← update traffic model(k, δ, Tk→δ , Mδ );
            kr ← get reinforcement(k, δ, Tk→δ , Mδ );
            Tδk ← update pheromone table(Tδk , r);
            Rk ← update data routing table(Rk , Tδk );δδ
        end if
    end for
end while
end procedure

```

The explanation of the both Algorithm i.e. ForwardAnt and BackwardAnt is as follows.

1. Let the source node S has data to send to a destination D with QoS requirements higher transmission rate, less delay and more bandwidth. A list of nodes that are progressively visited by the ant is called visited nodes list. This list forms the route R from the source node to destination node.
2. Initially choose the source node S. The visited nodes list will be initialized to source node (S).
3. S initiates a Path\_Request\_Ant to destination D through all its neighbors which are in 1-hop distance from S. The Path\_Request\_Ant contains source address, destination address, hop count and bandwidth.
4. After that the pheromone evaporation of all the 1-hop distance nodes will be calculated. Each node (i) maintains a table called "PMtab" a table of Pheromones specifying the quantity of available pheromone on each link (Vi, Vj). This quantity is initialized to constant C.

5. Then we calculate the pheromone evaporation of all the 2-hop distance nodes.
6. At last we calculate the path preference probability value of each path from source S with the help of pheromone evaporation of every node. A node j from a set of adjacent nodes (j, k... n) of i is selected as MPR node such that it covers all the 2-hop distance nodes and its path preference probability is better than others.
7. If calculated path preference probability value is better than the requirements, the path is accepted and stored in memory.
8. When the Path\_Request\_Ant reaches the destination, it will be converted as Path\_Reply\_Ant and forwarded towards the original source. The Path\_Reply\_Ant will take the same path of the corresponding Path\_Request\_Ant but in reverse direction.
9. The path with higher path preference probability will be considered as the best path and data transmission can be started along that path.

### **3.1. Stochastic Data Routing:**

The path setup phase described above creates a number of good paths between source and destination, indicated in the routing tables of the nodes. Data can then be forwarded between nodes according to the values of the pheromone entries. Nodes in AntHocNet forward data stochastically [6]. When a node has multiple next hops for the destination d of the data, it will randomly select one of them, with probability P<sub>nd</sub>. P<sub>nd</sub> is calculated in the same way as for the reactive forward ants, but with a higher exponent, in order to be more greedy with respect to the better paths:

$$P_{nd} = \frac{(T_{nd}^i)^{\beta_2}}{\sum_{j \in N_d^i} (T_{jd}^i)^{\beta_2}} \quad (4)$$

According to this strategy, we do not have to choose a priori how many paths to use: their number will be automatically selected in function of their quality. The probabilistic routing strategy leads to data load spreading according to the estimated quality of the paths. If the estimates are kept up-to-date (which is done using the proactive ants described in Subsection 3.3), this leads to automatic load balancing [7]. When a path is clearly worse than others, it will be avoided, and its congestion will be relieved. Other paths will get more traffic, leading to higher congestion, which will make their end-to-end delay increase. By continuously adapting the data traffic, the nodes try to spread the data load evenly over the network.

### **3.2. Path Maintenance Phase:**

Node managers can detect link failures (e.g., a neighbor has moved far away) when unicast transmissions (of data packets or ants) fail, or when expected pheromone diffusion agents were not received. When a link fails, a node might lose a route to one or more destinations, before they can lead to transmission errors and packet loss. If the node has other next hop alternatives to the same destination, or if the lost destination was not used regularly by data, this loss is not so important, and the node manager will just update its routing table and broadcast an effectors agent, termed failure notification ant. The agent carries a list of the destinations it lost a path to, and the new best estimated end-to-end delay and number of hops to this destination (if the node still has entries for the destination). All its neighbors receive the notification and update their pheromone table using the new estimates. If they in turn lose their best or their only path to a destination due to the failure, they will in turn generate and broadcast a failure ant, until all nodes along the different paths are notified of the new situation.

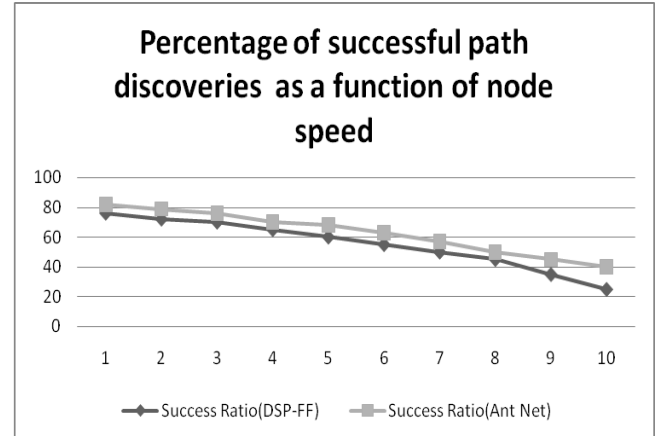
If the lost destination was regularly used for data traffic, and it was the nodes only alternative for this destination, the loss is important and the node should try to locally repair the path. This is the strategy followed in AntHocNet, with the restriction that a node only repairs the path if the link loss was discovered with a failed data packet transmission [8][9]. After the link failure, the node manager broadcasts a route repair ant that travels to the involved destination very alike a setup ant: it follows available routing information when it can, and is broadcast otherwise (with a limit of the maximum number of broadcasts). The node manager waits for a certain time and if no backward route repair ant is received, it concludes that it was not possible to find an alternative path to the destination which is then removed from the routing table and a failure notification ant is generated to advertise the new situation.

#### 4. FACTORS ASSOCIATED WITH DESIGNING THE ALGORITHM:

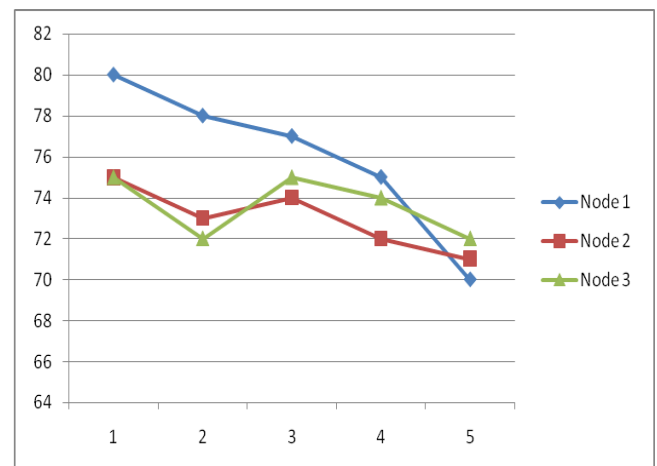
- **Packet Delay:** - A good algorithm should be able to deliver packets with minimum delay.
- **Session Delay:** - In case of session-oriented traffic, the most important parameter is time needed to complete a session. An application layer at the destination node only gets the packets after all the packets are received in the correct order. Packet delay factors out this waiting time and hence favors multi-path algorithms which deliver packets in an out-of-order manner but with smaller delays [10].
- **Sessions Completed:** - The percentage of sessions which are completed without any support from transport layer protocols.
- **Packet Delivery Ratio:** - This measure tells us how many data packets are successfully delivered at their destinations [11] [12].
- **Packet Drop Ratio:** - The percentage of data packets that are dropped because their time to live timer (TTL) value expired or the queue buffers were full.

#### 5. SIMULATIONS AND ANALYSIS:

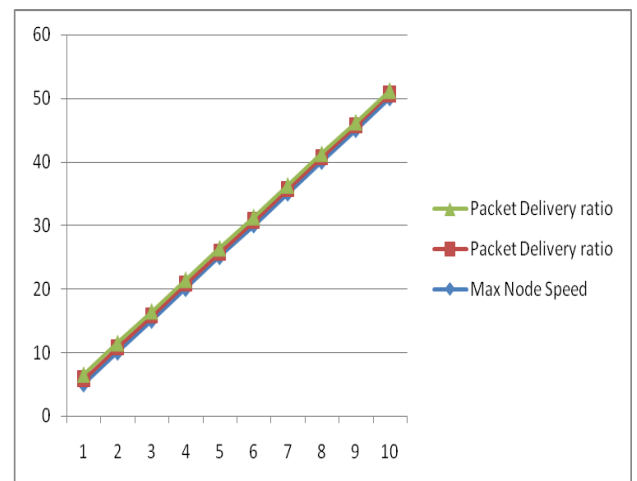
In this experiment we study the performance of ACO based algorithms in a high node mobility network. The results show how the performance of ACO based algorithms degrade as network mobility increases. Two nodes are connected if each one is in the transmission range of the other one. If the network graph is not fully connected, the source and destination may reside in different partitions which mean that there is no route to the destination. In our experiments we tried the routing algorithms only on connected graphs which are generated as mentioned. To estimate the performance, we ran the algorithms on 100 networks and averaged the results. For each network, one source-destination pair is selected randomly. The standard values used for the simulation results are plotted in Figure 5.1, Figure 2, and Figure 3. Figure 4, Figure 5.1the shows the percentage of successful path discoveries as a function the speed of node movement.



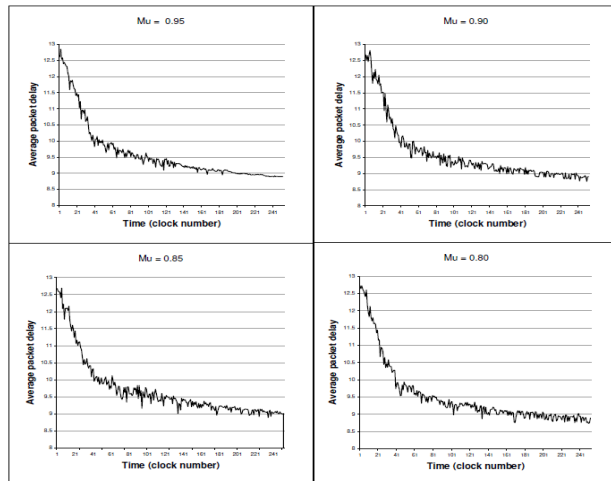
**Figure 2: Percentage of successful path discoveries as a function of node speed**



**Figure 3: Packet delivery ratio of ant packets experiences by three different nodes**



**Figure 4: The average ratio of successful packet delivery**



**Figure 5: Average packet delay in AntNet with different values assigned to parameter  $\mu$**

## 6. CONCLUSION:

In this article, we present a novel multi-parameter constrained QoS routing scheme based on a new ant colony algorithm for mobile ad hoc network. Simulation results demonstrated that with our Hybrid Ant-Net scheme, the performances of packets delivery ratio and average end-to-end delay are superior to the traditional dynamic routing protocol and ant-colony based routing algorithm. Our future work will include applying our Hybrid Ant-Net scheme to the wireless network with various scales and node densities to test its performance, and to further improve the algorithm

## 7. REFERENCES:

- [1] S. Goss, S. Aron, J. L. Deneubourg, and J. M. Pasteels. Self-organized shortcuts in the Argentine ant. *Naturwissenschaften*, 76:579–581, 1989.
- [2] C.Gui and P.Mohapatra. SHORT: Self-healing and optimizing routing techniques for mobile adhoc networks. In *Proceedings of MobiHoc*, 2003.
- [3] M. Gunes, " U. Sorges, and I. Bouazizi. ARA - the ant-colony based routing algorithm for MANETS. In *Proceedings of the ICPP International Workshop on Ad Hoc Networks (IWAHN)*, 2002.
- [4] M. Heissenbuttel " and T. Braun. Ants-based routing in large scale mobile ad-hoc networks. In *Kommunikation in verteilten System (KiVS03)*, March 2003.
- [5] D. Johnson and D. Maltz. *Mobile Computing, chapter Dynamic Source Routing in Ad Hoc Wireless Networks*, pages 153–181. Kluwer, 1996.
- [6] P.Deepalakshmi and Dr.S.Radhakrishnan, "Ant Colony Based QoS Routing Algorithm For Mobile Ad Hoc Networks," *International Journal of Recent Trends in Engineering*, Vol. 1, No. 1, May 2009.

- [7] Ajay C Solai Jawahar, "Ant Colony Optimization for Mobile Ad-hoc Networks," Department of Electrical Engineering, Rutgers.
- [8] Rajeshwar Singh, D K Singh and Lalan Kumar, "Ants Pheromone for Quality of Service Provisioning In Mobile Adhoc Networks," *International Journal of Electronic Engineering Research*, ISSN 0975 – 6450, Vol.2 , Number 1, pp. 101–109, 2010.
- [9] D. Camara ^ and A. Loureiro. Gps/ant-like routing in ad hoc networks. *Telecommunication Systems*, 18(1–3):85–100, 2001.
- [10] T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. *Wireless Communications & Mobile Computing: Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, 2002.
- [11] R. Schoonderwoerd, O. Holland, J. Bruten, and L. Rothkrantz. Ant-based load balancing in telecommunications networks. *Adaptive Behavior*, 5(2):169{207, 1996.
- [12] C.-C. Shen and S. Rajagopalan. Protocol-independent multicast packet delivery improvement service for mobile ad hoc networks. *Journal of Computer Science*, 5(2):210{227, 2007.
- [13] E. Sigel, B. Denby, and S. Le He\_arat-Masclle. Application of ant colony optimization to adaptive routing in a LEO telecommunications satellite network. *Annals of Telecommunications*, 57(5{6):520{539, 2002.
- [14] K.M. Sim and W.H. Sun. Ant colony optimization for routing and load-balancing: Survey and new directions. *IEEE Transactions on Systems, Man, and Cybernetics {Part A*, 33 (5):560{572, 2003.
- [15]D. Subramanian, P. Druschel, and J. Chen. Ants and reinforcement learning: A case study in routing in dynamic networks. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 832{838, San Francisco, CA, 1997. Morgan Kaufmann.

## AUTHORS PROFILES:

Mr. Mahendra Pratap Panigrahy received the degree Master of Technology in Electronics Engineering in 2005. He is a research student of Utkal University, Orissa. Currently, he is an Associate Professor at Institute of Technology Roorkee, Roorkee. His interests are in Image Processing & Embedded System.

Ms. Paramjeet Kaur, presently continuing her M.Tech. (CS), from Radha Govind Engineering College, Meerut under Mahamaya Technical University, Noida, UP. She has served more than three years in teaching at various engineering colleges. Her area of interests is in Computer Networks, Image Processing & Data Structure.