International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 3 (2015) pp. 6933-6944

http://www.ripublication.com

Bacteria Foraging Optimization Problem For Dynamic Shortest Path Routing Problem In Mobile Adhoc Network

S.Mangaiarkarasi and Dr. M. Karnan

Assistant Professor, Dept. of CSE,
Shanmuganathan Engineering College, Pudukkottai, Tamilnadu, India
mangaiprabhu@gmail.com
Principal, Arignar Anna College of Engineering and Technology,
Palani, Tamilnadu,India
drmkarnan@gmail.com

ABSTRACT

In Internet computing, Swarm Intelligence Algorithm has a fully grown interest in learning dynamic optimization problems like Ant Colony Optimization (ACO), Genetic Algorithm (GA) and Bacteria Foraging Optimization Algorithm (BFOA). In the recent years, there has been a growing interest in addressing Dynamic Optimization Problems (DOPs) using Swarm Intelligence (SI) algorithms. Several Algorithms are developed for SIs to enhance the diversity of the population and to reinforce the performance of the DOPs. In this paper, Bacteria Foraging Optimization Algorithm (BFOA) is applied for the Dynamic Shortest Path Routing Problem (DSPRP). Moreover, Experimental results show that Bacteria Foraging Optimization Algorithm efficiently enhances the performance of routing in dynamically changing environments.

Keywords: MANET, BFOA, GA, ACO.

I. INTRODUCTION

Mobile Ad hoc NETwork (MANET) [12,15] is a collection of wireless nodes that forms a temporary network without centralized administration. It is also called self organizing and self configuring multihop network. In multihop network, routing is one of the challenging issues that has a significant impact on the network performance. It controls the cost of the whole network as average end-to-end delay. So far, there are two types of routing protocols in MANETs, namely topological routing and geographical routing [17]. In this chapter, the shortest path (SP) routing problem was investigated, which belongs to the topological routing. In the topological routing, mobile host uses the topological information, to construct routing tables or search directly. An ideal routing algorithm should be capable of finding shortest (optimal) path, within a specified time so as to maintain the quality of service of a network. There are some search algorithms for solving optimization path problems - breadth-first search algorithm, the Dijkstra's algorithm and the Bellman–Ford algorithm and so on. Since these algorithms can solve the shortest path (SP) problems in polynomial time, they will be efficient in fixed infrastructure wireless or wired networks [1, 8]. Ad hoc networks routing protocols should be more dynamic to find a route faster in order to have a good response time to the speed of topology change [2].

In the recent years, several optimization algorithms are used for solving SP problem. The two most predominant solving methods of SP problem, involve Evolutionary Algorithms and Swarm based Algorithms [3] .One main principle behind swarm based algorithms is the concept of efficiency, interpreted as the potential of an individual to obtain a sufficient energy source [4] in the least amount of time. This process called foraging ,vital in natural search, since the animals with poor foraging strategies are removed, and successful ones tend to spread. Hence, to survive, an animal or a group of animals must expand an optimal foraging policy [11, 16,18].

The Dynamic Shortest Path Routing Problem (DSPRP) in MANETs is a real world Dynamic Optimization Problem (DOP)[4,5]. One of the easiest ways to address the DOP is using Bacteria Foraging Optimization Algorithm (BFOA) is popular in recent years because it has information distribution and transmission mechanisms. Among swarm optimization methods like Particle Swarm Optimization (PSO) [13], Genetic Algorithm [10] and Ant Colony Optimization (ACO) [7,9], BFOA is better in terms of faster convergence, global search space, robustness and precision. This method has a different set of advantages regarding local minima, randomness, direction of movement, attracting/repelling, swarming and so on.

This paper focuses on bacteria foraging optimization algorithm to deal with DOPs to solve the DSPRP in MANETs. Whenever the topology of the network is changed, the optimal solutions in the new environment can be investigated using this algorithm. The experiment results indicate that the proposed BFOA improves the performance of routing in dynamic environments efficiently.

The rest of the paper is outlined as follows. Related work is mentioned in section II. The Adhoc network model and the DSPRP model are described in section III. Section IV presents the design of a BFOA for the DSPRP. The extensive experimental study and relevant analysis are presented in section V. Section VI concludes this paper with some discussions.

II. RELATED WORKS

Several search algorithms were formulated for SP routing problem. In [2], a genetic algorithm approach was presented for solving SP routing problem. Simulation studies show that the algorithm is indeed intensive to network topologies in respect of both route optimality and convergence. The quality of solution found to be better than other deterministic algorithms.

In [6], several modifications are applied to the standard GA on track in a changing environment. An experiment shows that the algorithm exhibits difficulties in tracking continuously changing environment. In [7,9], an Ant Colony Optimization (ACO) was proposed for solving SP routing problem. Observation shows that, ants can find the shortest path between food sources and their nest. But it doesn't always find the optimal solution. In [13], a PSO based algorithm was presented for solving SP problems. The PSO based algorithm is superior to GA [2,6]. In [1] Hopfield neural network was proposed. This algorithm produces a faster convergence and better route optimality than other HNN based algorithms. However, the above said algorithms are not suitable alternative for solving DSPRP in MANETs; here we implement the Bacteria Foraging Optimization Algorithm (BFOA) to obtain the optimal solution for DSPRP in MANETs.

III MODEL FOR DYNAMIC SHORTEST PATH ROUTING PROBLEM

In this section, let us consider ad hoc network model and then devise the DSPRP [17]. We model an ad hoc network operating within a fixed environmental region. It can be represented by an undirected and connected topology graph G_0 (N_0, E_0). Where N_0 specifies the set of wireless nodes and E0 specifies the set of its links (edges) connecting two adjacent nodes falling into the radio transmission range. If there exists a packet transmission in the link (i, j) then both nodes i and node j should have a radio interface, each with a universal channel.

The parameters used in the paper:

G (N_0, E_0) initial Adhoc Network topology graph;

G_i (N_i, E_j) Adhoc network topology after i^{th} chance;

S Source node;

T Sink node;

Pi (s, t) path from s to t in graph G_i;

C_l cost on communication link l;

Adhoc network can be represented as follows: Initially it is given a network of wireless nodes, a delay upper bound, a source node, a sink node. We wished to find a delay bounded least cost loop free path on the undirected topology graph. In mobile Adhoc networks, the topology changes from time to time. The objective of problem (DSPRP) is discovering the optimal path after every topology change.

IV BACTERIA FORAGING OPTIMIZATION ALGORITHM FOR SP ROUTING PROBLEM

Bacteria Foraging Optimization Algorithm (BFOA),introduced by Kevin M.Passion in 2002 is one of the bio- inspired optimization algorithm based on the principle of social foraging behavior of Escherichia Coli(E. Coli) bacteria and the natural selection, which has been quite effective applied in machine learning and optimization problems. To solve a problem, a BFOA maintains a population of bacteria and probably modifies the population by reproduction and elimination and dispersal operator, with the objective of seeking a near optimal solution to the problem. The BFOA design is governed by representation of bacteria, chemotaxis, swarming, reproduction and elimination and dispersal[11].

A. Representation of Bacteria

In the proposed algorithm, any path from the source node to the destination node is a feasible solution. The optimal solution is the shortest one. At the start, a random population of strings are generated which represent feasible or unfeasible solutions. Unfeasible solutions are strings that cannot reach the destination. A bacterium corresponds to the possible solution of the problem of the optimization problem. Thus, each bacteria represents path which consists of sequences of positive integers that represents the IDs of nodes through which a routing path passes with the source node followed by an intermediate node (via nodes) and the last node indicating the destination, which is the goal. The default maximum bacteria length is equal to the number of nodes.

B. Chemotaxis(population initialization)

A chemotaxis step is a set of consequences swim steps followed by tumble. At this stage, the node or processor has to set t_M which serves as an upper bound to the run-time of the algorithm. The fitness of bacterium is evaluated which further decides next movement of the bacterium. The routing paths (RP) for each source to designation nodes (SD) are chosen. From the BFO point of view, the RP and the SD are the bacteria, which carry the bacterium information over swimming operations. The assignment to RP and SD are random(Tumble).

C. Swarming

Every bacterium in the population is set to travel to the rich nutrition gradient. The groups in the cells have two kinds of behavior, either it may be attractant or repellant. The attractant behavior used to swarm with high fitness value i.e. more better path when moving to nutrition gradient. The RP of the all possible Source-Designation pair’s nodes are found. The probabilities are recalculated to produce the next operations. The cost function we use will reward and penalize the next node.

The node to node signaling measured using the following equation

$$C_i = \sum_{p=1}^{l} H(p, k, r, e) \tag{1}$$

Where C_i represents cost function of l links, p represents current node, k represents p's next node, r represents reproduction, e represents elimination.

D. Reproduction

Health status (fitness) of each bacterium is calculated after each complete chemotaxis process. It is overall sum of the cost function

$$f(SDp) = \frac{1}{\sum_{h\in(S,D)} C_{i\,h}} \tag{2}$$

Where f(SDp) represents to fitness function of Source to Destination Pairs, C_i represents cost of i^{th} pair ,'h' represents number of links between SD.

To simulate the reproduction character in nature and to accelerate the swarming speed, all the bacteria are sorted according to their health values in an ascending order and each of the first bacteria splits into two bacteria. The characters including position and step length of the mother bacterium are reproduced to the children bacteria. In MANET, when the routing path reaches a static equilibrium, the cost functions of nodes are ordered in ascending order. The Routing Path(RP) cost of Source-Destination(SD) which have values higher than optimum values are eliminated.

E. Elimination and Dispersal

For the purpose of improving the global search ability, elimination-dispersal event is defined after reproductive steps. The bacteria is eliminated and dispersed to random positions in the optimization domain according to the elimination-dispersal probability. This elimination-dispersal event helps the bacterium to avoid being trapped into local optima. In MANET sometimes the elimination can occur, but the reproduction did not take place. In this scenario there is difficulty to identify the local optimum and global optimum cost values for the RPs, such that the elimination dispersal can maintain constant population in the search space.

V EXPERIMENTAL SETUP

In this scenario, mobile ad hoc network is of 100 nodes placed randomly by using a uniform distribution in an area of 1000 X 1000 m^2 is considered for simulation study. The nodes in the network have the transmission range of 60 to 70 m and a channel capacity of minimum 750Kbps to maximum 2Mbps.The network model used in the simulation is composed of mobile nodes and wireless links that are considered bidirectional. The mobility model uses the Random Waypoint Model (RWP) to create the movement patterns of independent nodes for the simulation scenarios needed.

RWP is one of the most widely used random-based synthetic mobility models in performance analysis of ad-hoc networks. In this model, the mobile nodes start their journey from a random location and move to a random destination without any restrictions, the velocity with which the nodes move are randomly selected from a uniform velocity distribution. After reaching a random destination, the node will pause (wait) before moving to the next destination. Several scenarios were obtained from RWP by varying the velocity of the nodes and the pause times.

A. Simulation Results

In this section, we present a comprehensive simulation based evaluation of routing metrics using the popular NS2 simulator. For evaluating the routing performance, the investigator proposed three schemes in this paper: (i) Genetic Algorithm (GA) (ii) Ant Colony Optimization (ACO) (ii) Bacteria Foraging Optimization Algorithm BFOA (BFOA) (see section V). We conduct two sets of experiments. In the first set of simulation, the researcher demonstrates the node adaptation in a dynamic changing environment by considering the impact of data traffic on different metrics. This enables the researcher to investigate which schemes contribute to the performance more significantly. The researcher uses a set of metrics to evaluate the impact of proposed schemes on routing performance. These includes: (i) Packet delivery ratio (ii). Throughput (iii) end to end delay (iv) Jitter (v) Routing overhead. Also, to show the excellence of BFOA with other algorithm the fitness graph is drawn (section c).

B. Impact of Data Traffic

First, the researcher evaluates the impact of varying number of connections with different metrics for the performance of the proposed schemes. The different node density levels are obtained by keeping the area size constant and increasing the number of nodes. The results presented here are averaged over 20 runs. The results of these tests are reported in Figure.6.1 to 6.6. BFOA performs better than ACO and GA in terms of the packet delivery ratio, end to end delay, jitter, routing overhead, throughput and path optimality with increase in the difference with the density. The numbers of connections indicate the number of nodes between which the data are transmitted or a data communication has been set. The number is incremented in steps of 4 from 4 to 20. The network is configured for 20 nodes; the nodes are set to move at a maximum speed of 5m/s pausing for every 50 seconds (pause time is set to 50.The packet Delivery Ratio for the network is reported in the Fig.1. with increase in the number of nodes. It is seen that BFOA was able to produce maximum effectiveness when compared to the other algorithms in the network. The packet delivery rate could be increased because of the foraging nature of the BFOA than GA. BFOA on the other hand finds the optimal path when a data packet arrives and thus it was able to deliver the data packets, even under dynamic conditions.

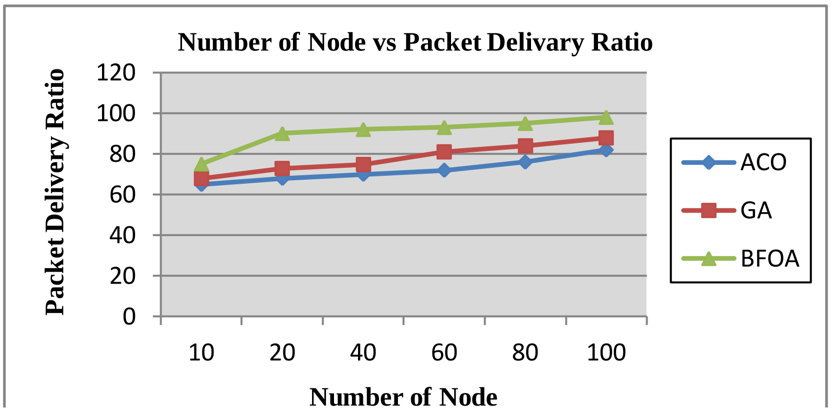

Fig .1 Comparison of Number of Nodes Vs Packet Delivery Ratio.

In Fig.2, throughput has measured in varying number of data connections. At the higher node mobility BFOA outperformed than other algorithm by transmitting maximum number of packets. BFOA showed better results in case of lower and higher mobile conditions (20 connections).

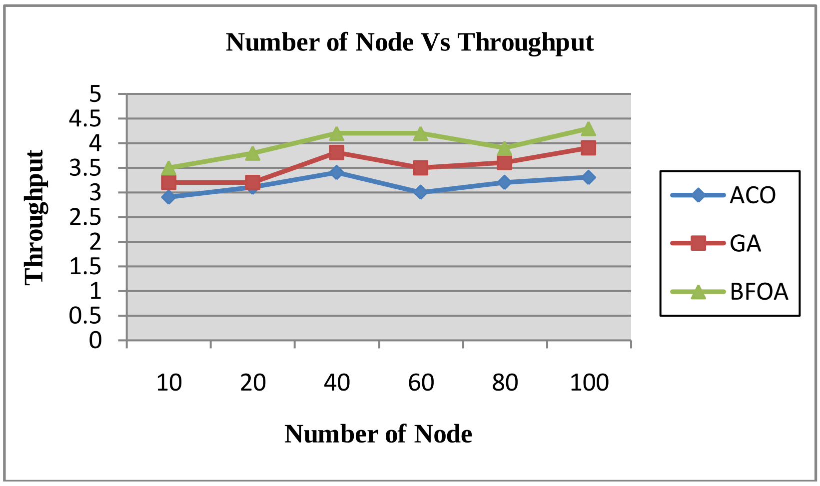

Fig .2 Comparison of Number of Nodes Vs Throughput.

In Fig. 3, the average end-to-end delay is measured in varying number of the nodes. At the higher node density BFOA outperformed than GA and ACO. BFOA showed better results in case of higher mobile conditions.

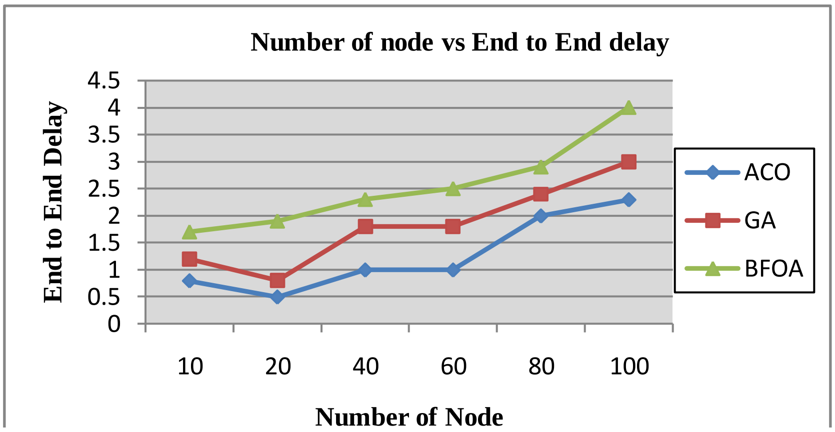

Fig .3 Comparison of Number of Nodes Vs End to end delay

In Fig. 4 shows the variation in the average jitter values with varying number of nodes by the algorithms.

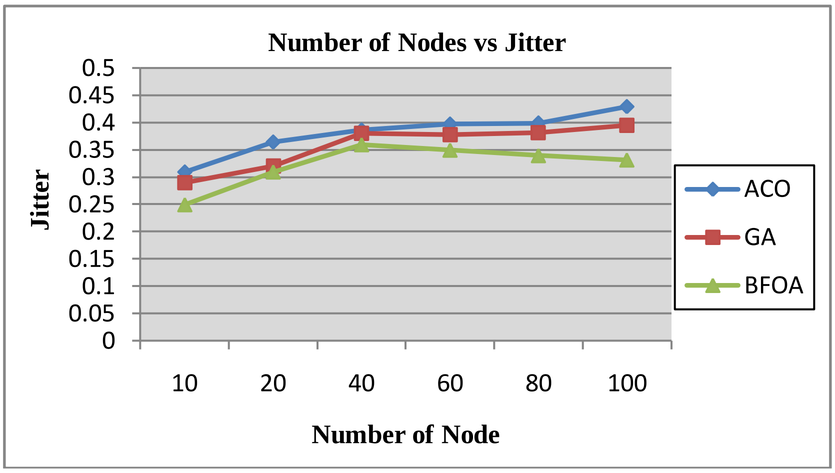

Fig .4 Comparison of Number of Nodes Vs Throughput.

Figure 5 shows the variation of routing overload with node mobility. The routing load in BFOA increases with an increase in the number of nodes. As the routing load for BFOA, GA and ACO algorithm remains independent of the node density. The routing load in the BFOA has also remained varying because the network topology changes in this experiment.

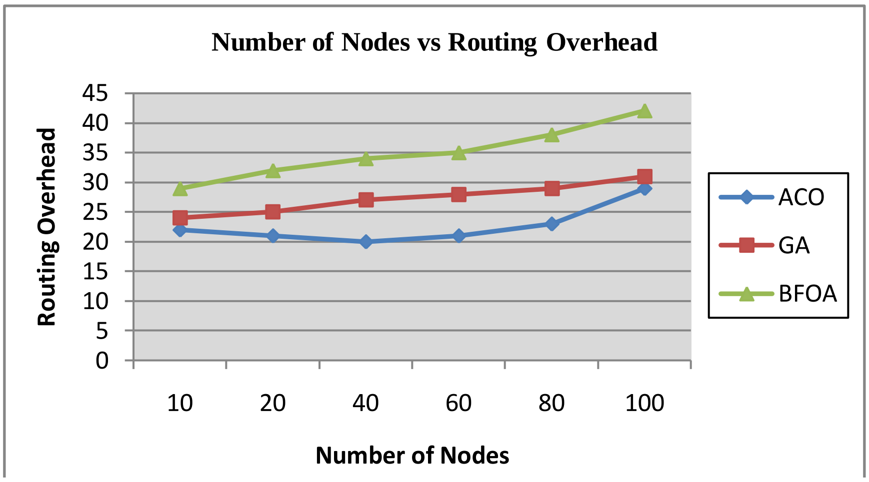

Fig .5 Comparison of Number of Nodes Vs Routing Overhead

This is an important result which indicates that BFOA is more scalable with respect to the number of nodes. BFOA as shown to outperform ACO and GA over the wide range of testing scenarios in terms of delivery ratio, the average end-to-end the delay and average jitter, while generating a comparable amount of control overhead. An important observation was that the advantage of BFOA over ACO, GA grew for larger networks, especially in terms of overhead, suggesting that BFOA is more scalable than ACO and GA.

C. Performance Comparison of Quality of Solution

On the basis of performance of quality of solution, the excellence of BFOA is compared exclusively with other algorithms. The focus of fair comparison is investigated in terms of average number of fitness function evaluations. The number of fitness function evaluation directly measures the excellence of performance as shown in Fig.6. The proposed BFOA can find a quality solution with minimum number of reproduction in the dynamic environment. It is purely due to the dynamically updating the fitness function and replacing the worst solution by generated immigrants.

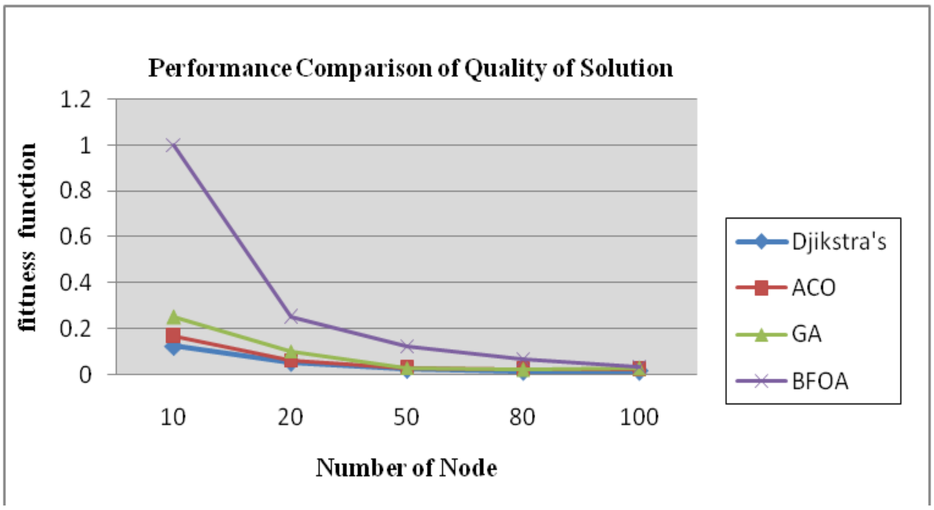

Fig. 6. Performance Comparison of Quality of Solution

The number of fitness function evaluation is calculated with respect to the number of nodes as shown in the figure. From the figure, we can interpret that the average number of fitness function evaluation is smaller in every case, because the maximum difference between all the cases is negligible. The fitness function of BFOA proved to be differing from fitness function of GA, ACO and Dijkstra's algorithm, clearly formulates that the proposed algorithm is efficient in dynamic environment. The proposed bacteria foraging optimization algorithm efficiently improve performance of routing performance in dynamically changing environments.

VI CONCLUSION AND FUTURE WORK

In this paper, the we have identified the need of BFOA for DSPRP in MANET. The proposed algorithm has been tested for several performance metrics and the results obtained are compared with the results of earlier methods such as GA and ACO available in the literature. As compared to other two, the BFOA is easy to implement and there are few parameters to adjust. Therefore, BFOA has been successfully applied in the areas of MANETs. Therefore, BFOA have been successfully applied in the areas of MANETs. From the outcome of the results, it is shown that the proposed BFOA is very effective in giving the optimal solution for Dynamic SP Problems in cyclic dynamic environments. We gives an opportunity to further investigate the BFOA in multicast routing problem in a dynamic network environment in future.

REFERENCES

1. C. W. Ahn, R. S. Ramakrishna, C. G. Kang, and I. C. Choi, "Shortest path routing algorithm using Hopfield neural network," *Electron. Lett.*, Vol. 37, no. 19, pp. 1176–1178, Sep.2001.
2. C. W. Ahn, and R. S. Ramakrishna, "A Genetic algorithm for Shortest Path Routing Problem and the Sizing of Populations," *IEEE Transactions on Evolutionary computation*, Vol. 6, no. 6, Dec. 2002.
3. Bijaya Ketan Panigrahi, Yuhui Shi, and Meng-Hiot Lim, "Handbook of Swarm Intelligence Concepts, Principles and Applications," *Springer*, pp. 487-502, 2011.
4. J. Branke, T. Kaußler, C. Schmidth, and H. Schmuck, "A multi-population approach to dynamic optimization problems," *In Proc. of the Adaptive Computing in Design and Manufacturing*, pages 299–308, 2000.
5. J. Branke, "Memory enhanced evolutionary algorithms for changing problems," *In Proc. of the 1999 Congress on Evolutionary Computation*, volume 3, pages 1875–1882, 1999.
6. H. G. Cobb and J. J. Grefenstette, "Genetic algorithms for tracking changing environments ," In S. Forrest, editor, Proc. of the 5th Int. Conf. on Genetic Algorithms, pages 523–530, 1993.
7. Dilpreet Kaur, Dr.P.S Mundra, "Ant Colony Optimization: A Technique Used In Finding Shortest Path ," *International Journal of Engineering and Innovative Technology (IJEIT)* , Volume 1, Issue 5, ISSN: 2277-3754, May 2012.
8. A. Das and C. Martel, "Stochastic shortest path with unlimited hops," *Inf. Process. Lett.*, vol. 109, no. 5, pp. 290–295, 2009.
9. M. Guns, U. Sorges, I. Bouazizi, Ara, "the ant colony based routing algorithm for Manets," *in: ICPP workshop on Ad Hoc Networks (IWAHN 2002)*, pp. 79-85, August 2002.
10. Jason Brownlee, "Clever Algorithms: Nature-Inspired Programming Recipes," *Jason Brownlee* pp. 257-264, 2011.
11. Kevin M. Passino, "Bacterial Foraging optimization," *International Journal of Swarm Intelligence Research*, pp 1-16, Jan-Mar 2010.
12. C. S. R. Murthy and B. S. Manoj, *Ad HocWireless Networks: Architectures and Protocols*. Englewood Cliffs, NJ: Prentice-Hall, 2004.
13. A. W. Mohammed, N. C. Sahoo, and T. K. Geok, "Solving shortest path problem using particle swarm optimization," *Appl. Soft Comput.*, Vol. 8 no. 4, pp. 1643–1653, Sep.2008.
14. M. Mavrovouniotis and S. Yang, "Memory-based immigrants for ant colony optimization in changing environments," *in EvoApplications 2011: Appl. of Evol. Comput.*, LNCS 6624, pp. 324–333, 2011,.
15. C. E. Perkins Ed., *Ad Hoc Networking*. London, U.K.: Addison-Wesley, 2001.
16. K.M passion, "Biomimicry of bacterial foraging distributed optimization," university press, *prince to NeweJersy*, 2001.

17. Shengxiang Yang, Hui Cheng, and Fang Wang, "Genetic Algorithms With Immigrants and Memory Schemes for Dynamic Shortest Path Routing Problems in Mobile Ad Hoc Network," *IEEE transaction on System, Man, and Cybernetics-Part C: Application and Reviews*, Vol 40, No1, Pages 52-63, January, 2010.
18. SibylleD. Muller, Jarno Marchetto, Stefano Airaghi, and Petros Koumoutsakos," Optimization Based on Bacterial Chemotaxis, " *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, February 2002.
19. K. Trojanowski and Z. Michalewicz, "Evolutionary optimization in nonstationary environments," *J. Comput. Sci. Technol.*, vol. 1, no. 2, pp. 93–124, 2000.
20. R. Tinos and S. Yang, "A self-organizing random immigrants genetic algorithm for dynamic optimization problems," *Genet. Program. Evol. Mach.*, vol. 8, no. 3, pp. 255–286, Sep. 2007.
21. F. Vavak and T. C. Fogarty, "A comparative study of steady state and generational genetic algorithms for use in nonstationary environments," in *Proc. AISB Workshop Evol. Comput.*, 1996, pp. 297–304.