ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology
Volume 1, Issue 5, July 2012

An ACO Approach to Solve a Variant of TSP

Bharat V. Chawda, Nitesh M. Sureja

Abstract – **This study is an investigation on the application of Ant Colony Optimization to a variant of TSP. This paper presents an Ant Colony Optimization (ACO) approach to solve a randomly generated TSP problem known as RTSP. TSP data sets, used in this research, are created randomly with coordinates in the range 0 to 100. The source code for the above has been developed in MATLAB 7. Ant Colony Optimization is applied on several randomly generated TSP data sets. The results obtained from the model has been collected and analyzed based on several criteria like convergence time, quality of solution, and length of the tour. From the analysis, it is found that this approach works very well in terms of convergence time and length of the tour. At the same time, results start to degrade itself with the increase in size of the data set. It is clearly found that this approach can produce best results for any optimization problem if it is applied properly.**

Index Terms – **Ant Colony Optimization, Nature Inspired Approach, Optimization, Random Traveling Salesman Problem**

I. Introduction

Normal mathematical approach generally fails while it is applied to solve a NP-hard optimization problem like TSP. This type of problems leads researchers to develop some alternative approaches to find the solutions. Numerous approaches are proposed by the various authors. Nature and Biology inspired approaches are the part of them. These approaches consist of Simulated Annealing (SA) [1], Genetic Algorithms (GA) [2], Particle Swarm Optimization (PSO) [3], Bee Colony Optimization (BCO) [4], Artificial Immune System (AIS) [5], Firefly Algorithms (FA) [6], Monkey Search (MS) [7], Harmony Search (HS) [8], Bat Inspired Approach (BIA) [9] and Ant Colony Optimization (ACO) [10] as some examples of them. We use Ant Colony Optimization (ACO) approach in this paper. Ant Colony Optimization (ACO) approach is inspired by the foraging behavior of ants. Foraging behavior of the ants are observed and modeled mathematically to solve the optimization problems. Marco Dorigo is the man who proposed this approach first [10]. He applied this to various problems and found that this approach has a big potential in it for solving various problems. We apply ACO approach to Random Traveling Salesman Problem (RTSP) in this paper.

The paper starts with a discussion on Random Travelling Salesman Problem in section II. A discussion on ACO approach is presented in section III. Section IV discusses the proposed ACO model. Implementation details along with results of proposed model are discussed in section V. Finally the paper ends with conclusions and acknowledgment.

Manuscript received June, 2012.
Bharat V Chawda, *Computer Engineering Department, B. & B. Institute of Technology, Gujarat Technological University, (e-mail: bharat.bbit@gamil.com). Ahmedabad, India, Mobile No. 09978917637*
Nitesh M Sureja, *Computer Engineering Department, G.H. Patel College of Engineering and Technology, Gujarat Technological University, (e-mail: nmsureja@gamil.com). Ahmedabad, India, Mobile No. 09825313936*

II. Random Traveling Salesman Problem

Travelling Salesman Problem (TSP) is the most studied problem for finding optimal solution. Given a graph G with n number of cities, the objective is to find a shortest close tour that visits each city once returning to the starting city. As far as the heuristic approach is concerned, TSP has provided many algorithms for finding near optimal solutions for symmetric as well as asymmetric TSP. This paper presents an ACO approach for solving a variant of TSP known as Random Traveling Salesman (RTSP) [14][15]. In this variant TSP dataset is generated randomly instead of using available dataset on TSPLIB. All the city coordinates are generated here are in the range 0 to 100.

III. Ant Colony Optimization

ACO [10][11][12] is an algorithm inspired by the foraging behavior of the real ants. The behavior of the real ants while searching for the food is modeled mathematically in this algorithm. Real ants deposit a definite amount of pheromone in its path while traveling from its nest to the food. They also deposit some pheromone value on the path while returning. The ants following the shorter path return earlier with increasing the amount of pheromone deposit on the path at a faster rate. After some time this path becomes favorite path to travel for all ants as this one is a shorter path. Also the pheromone evaporates by a certain amount at a defined stable rate after a certain interval. The longer paths which are not visited frequently are eliminated due to this evaporation. So here all ants start their journey with the knowledge left by the ants which traversed previously and try to follow the shortest path directed by the pheromone trail created by them. A number of artificial ants try to build solutions for the problem considered with the help of the pheromone deposit knowledge and some other required information related to the specific problem.

A. *Working of ACO*

All the problems are converted in to a graph before applying ACO to them. Traveling salesman problem is a very good example of it to understand. Here, in a graph all cities are represented by the nodes while the arcs represent the path between the cities. We have to find a tour made of nodes and arcs, which represents a shortest tour in terms of distance, cost and quality. We use two things or parameters, distance between the cities and a pheromone value laid/evaporated on/from the arcs (paths). ACO works as follows [10][13].

In the construction of the solution, first we put ants on the nodes randomly. Normally we keep number of ants same as number of cities. After this initialization, all ants start their journey to find a shortest tour. Ants select the next city to be visited using a probabilistic mechanism. When ant k is in city i and has so far constructed the partial solution, the probability of going to city j is given by

$$p_{ij}^{k} = \frac{(\tau_{ij})^{\alpha}(\eta_{ij})^{\beta}}{\sum_{l\epsilon N_i^k}(\tau_{ij})^{\alpha}(\eta_{ij})^{\beta}} \quad if\ j\epsilon N_i^k \qquad (1)$$

Where, N_i^k is the feasible neighborhood of ant k when being at city i, that is, the set of cities that ant k has not visited yet [10]. α and β are the parameters which control the relative importance of the pheromone values and the heuristic information η_{ij}, which is the inverse of the distance between city i and j. We can say, in general, arcs that is used by many ants and which are part of short tours, receive more pheromone and are therefore more likely to be chosen by ants in future iterations of the algorithm. Pseudo code of ACO algorithm is as follows.

1. **Procedure** ACO
2. Initialize pheromone trails and other parameters
3. **while** (termination criteria not met)

   {

   a. Construct the Solutions
   b. Daemon Actions % optional
   c. Update Pheromone Trails

   }
4. **end** ACO procedure

Fig. 01 Pseudo code of ACO [10][11][12]

After all ants construct their solution (tour), each solution is analyzed to find the best solution from all constructed solutions. If the best solution found in current iteration is better than previous best one, a replacement is performed. Next step is known as Pheromone Update. It is done in two ways known as pheromone evaporation and pheromone laying. The pheromone values are updated by all the ants that have built solutions. First the pheromone values on all arcs are lowered by a constant factor which is known as pheromone evaporation. After that, a certain amount of pheromone values are added on the arcs the ants have crossed in their tour which is known as pheromone laying. Pheromone evaporation is implemented as per equation (2).

$$\tau_{ij} \leftarrow (1-\rho)\tau_{ij} \qquad (2)$$

Where, ρ is the evaporation rate.

After evaporation, all ants perform pheromone laying operation by depositing a certain amount of pheromone as per equation (3).

$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \qquad (3)$$

Where,

1. m is the number of ants
2. $\Delta\tau_{ij}$ is the quantity of pheromone laid on edge (i, j) by ant m.

$\Delta\tau_{ij}$ is calculated as

$$\Delta\tau_{ij}^{k} = \begin{cases} 1/C^{k}, & \text{if arc } (i,j) \text{ belongs to } T^{k} \\ 0, & \text{othrwise} \end{cases} \qquad (4)$$

Where C^k represents the length of the tour T^k built by k^{th} ant.

IV. Proposed ACO-RTSP Model

Random Traveling Salesman problem (RTSP) has been already mentioned previously [14][15]. To repeat it, all the TSP data sets are created by randomly generating city coordinates in the range 0 to 100. The task is to find a sequence of cities to minimize traveled distance for those data sets. Pheromone trails and heuristic information is same as in the TSP: pheromone trails are associated with the arcs (edges), and heuristic values are given by the inverse of the distance between cities. Pseudo code of the proposed ACO-RTSP Model to solve RTSP is given as per following:

1. Generate TSP Data Set Randomly
2. **Procedure** ACO
3. Initialize pheromone trails and other parameters
4. **while** (termination criteria not met)

   {

   a. Construct the Solutions
   b. Daemon Actions % optional
   c. Update Pheromone Trails

   }
5. **end** ACO procedure

Fig. 02 Pseudo Code for Proposed ACO-RTSP Model

V. Implementation and Results

The proposed ACO-RTSP Model is implemented by using MATLAB 07. Pentium dual core machine with 1(one) GB RAM is used to run the algorithm. Various data sets, in the range of 10 to 200 cities, of Random Traveling Salesman Problem (RTSP) are used to test the proposed model.

After generating a random data set of TSP, remaining steps of proposed ACO-RTSP model are implemented using the equations 1 to 4. This model runs itself till the termination criteria are satisfied. We can use more than one termination criteria in the model such as predefined number of iterations, stagnations in the result, time-limit etc. Here, stagnation in the result is used as the termination criteria. Results obtained and shown in table 01 are averaged over 25 runs of proposed model for each data set. Results are also represented graphically in figures 03-to-14.

ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology
Volume 1, Issue 5, July 2012

Table 01 Results of Proposed ACO-RTSP Model

City Problem	Length	Iterations
10	302.00	12.12
20	336.00	13.28
30	460.04	19.56
40	509.36	20.60
50	586.20	23.88
75	671.96	25.36
95	771.68	29.56
115	842.32	30.12
135	945.36	30.84
155	965.96	31.68
175	1057.96	32.24
200	1149.56	35.88

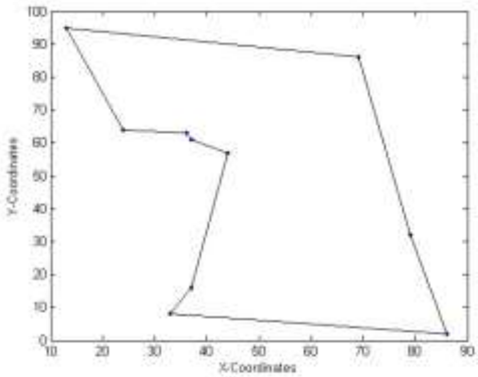

Fig. 03 10-City Data Set Results

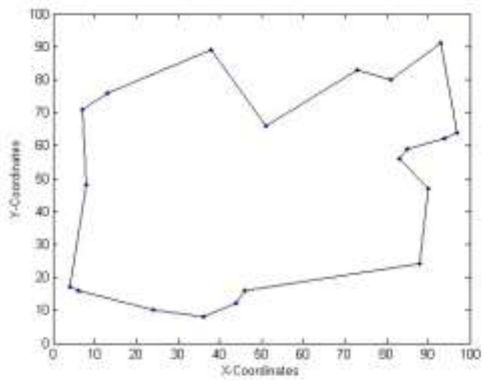

Fig. 04 20-City Data Set Results

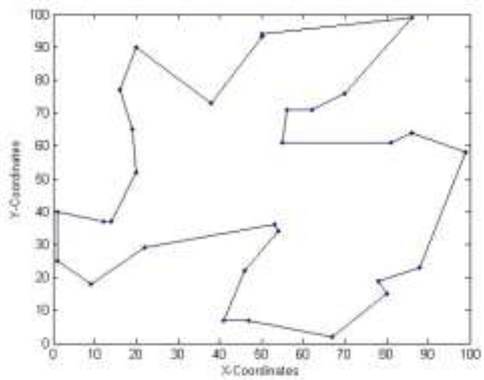

Fig. 05 30-City Data Set Results

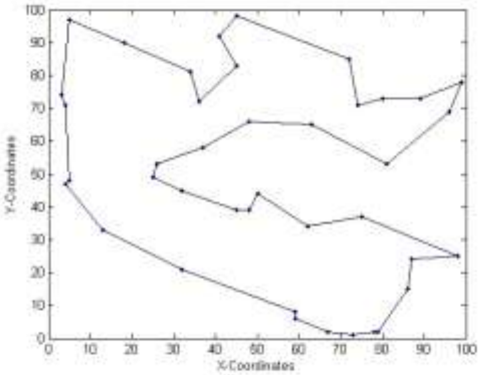

Fig. 06 40-City Data Set Results

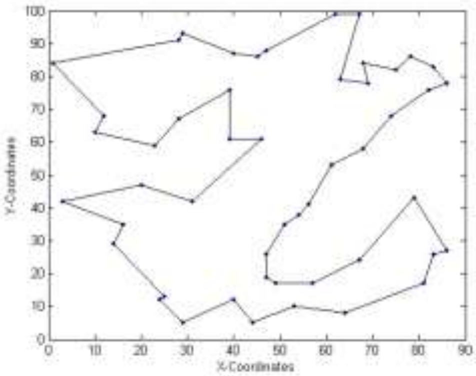

Fig. 07 50-City Data Set Results

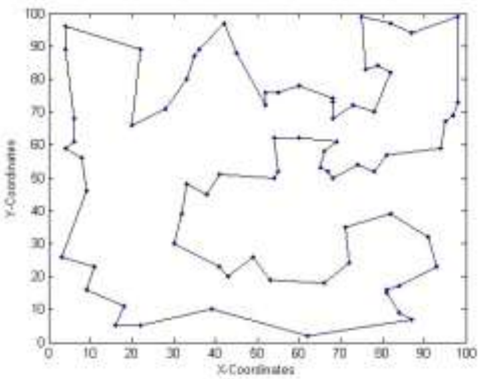

Fig. 08 75-City Data Set Results

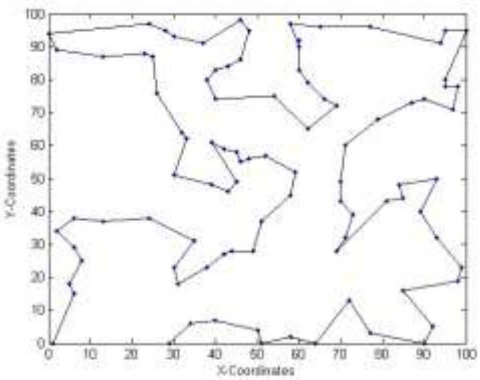

Fig. 09 95-City Data Set Results

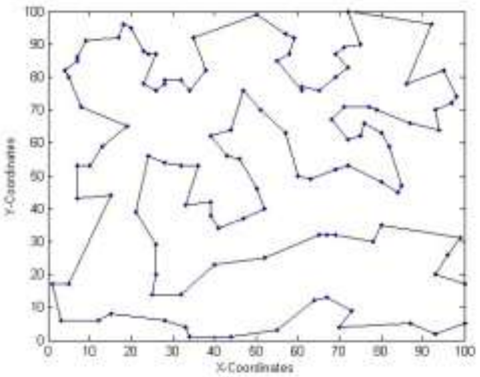

Fig. 10 115-City Data Set Results

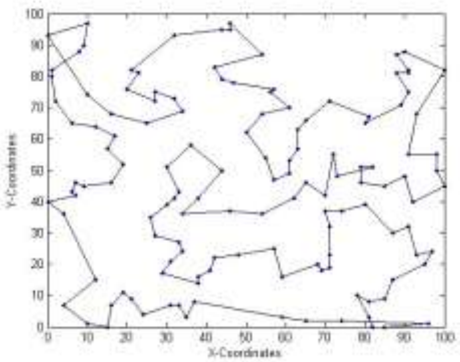

Fig. 11 135-City Data Set Results

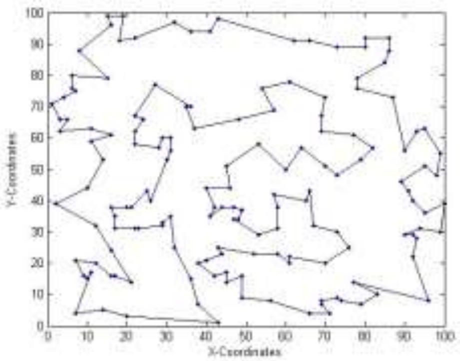

Fig. 12 155-City Data Set Results

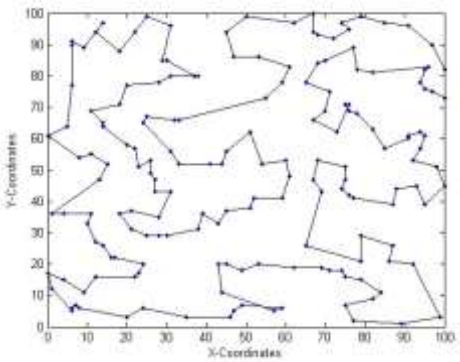

Fig. 13 175-City Data Set Results

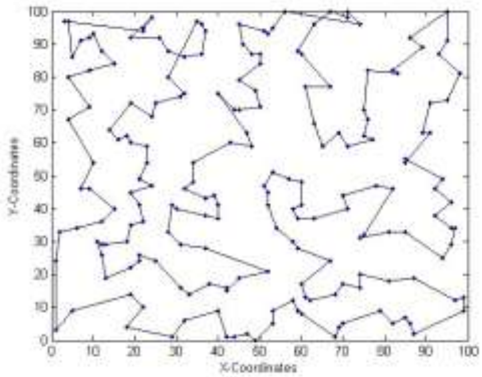

Fig. 14 200-City Data Set Results

VI. CONCLUSION

An approach based on Ant Colony Optimization is proposed in this paper. This approach is applied to various data sets created for Random Traveling Salesman Problem. This Approach produces the acceptable optimal solutions for all data sets ranging from 10 cities to 200 cities. We can also see that model starts to degrade in its performance in terms of solution quality with the increase in the size of problem. We conclude with a statement that this approach has a lot of potential in it which can be applied to solve any kind of optimization problem.

ACKNOWLEDGMENT

We thank all who have supported us for this research.

REFERENCES

[1] David Bookstaber, "Simulated Annealing for Traveling Salesman Problem", Spring, 1997
[2] Holland, J.H. "Adaptation in Natural and Artificial Systems". MIT Press, 1992
[3] Kennedy, J.; Eberhart, R. (1995). "Particle Swarm Optimization". Proceedings of IEEE International Conference on Neural Networks. IV. pp. 1942–1948
[4] D. Karaboga, B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm", *Journal of Global Optimization* 39 (2007) 459–471
[5] D. Dasgupta, "Artificial Immune Systems and Their Applications", Springer, Berlin, 1999
[6] X.S Yang, "Fire fly algorithm for multimodal optimization", in proceedings of the stochastic Algorithms. *Foundations and Applications (SAGA 109) vol.5792 of Lecture notes in Computer Sciences* Springer,Oct.2009
[7] R. Zhao, W. Tang, "Monkey Algorithm for Global Numerical Optimization", *Journal of Uncertain Systems*, Vol. 2, No. 3, pp. 165-176, (2008)
[8] X.S. Yang, "Harmony Search as a Metaheuristic Algorithm", *Studies in Computational Intelligence*, Springer Berlin, Vol. 191, pp. 1-14 (2009)
[9] X.-S. Yang, "A New Metaheuristic Bat-Inspired Algorithm", *Studies in Computational Intelligence*, Springer Berlin, 284, Springer, 65-74 (2010).
[10] M. Dorigo, L. Gambardella, "Ant colonies for the Traveling salesman problem." *Biosystems 43 (1997): (73-81).*
[11] M. Dorigo, T. Stutzle, "Ant Colony optimization", A Bradford book, MIT Press Cambridge, Massachucetts london, England (2004) .
[12] M. Dorigo, V. Maniezzo, A. Colomi, "The Ant System: Optimization by a colony of cooperating agents", *IEEE Transactions on Systems, Man and Cybematics-Part B, Vol. 26, No.1, 1996.*

[13] M. Dorigo, L. Gambardella, "Antcolony System: A Cooperstive learning Approach to the Traveling salesman problem." *IEEE Trans. Evol.Comp. 1 (1997): 53-66.*

[14] N. Sureja, B. Chawda, "Random Traveling Salesman problem using Genetic Algorithms," *IFRSA's International Journal Of Computing, Vol 2, issue 2, April 2012*

[15] N. Sureja, B. Chawda, "Random Travelling Salesman Problem using SA," *International Journal of Emerging Technology and Advanced Engineering, Volume 2, Issue 4, April 2012*

[16] A. Zhou, L. Kang, Z. Yan, "Solving Dynamic TSP with Evolutionary Approach in Real Time", Proceedings of the congress on Evolutionary computation, Canberra, Australia, 8 – 12, December 2003, IEEE Press, 951 – 957,2003.

Mr. Bharat V. Chawda is a B.E. and M.E. in Computer Engineering. He is a Gold-Medalist in M.E. from Sardar Patel University, Vallabh Vidyanagar. His current profile comprises of Lecturer (Computer Engineering) B. & B. Institute of Technology, Vallabh Vidyanagar. He is a life member of ISTE and IEANG. He has been offering his services as reviewer of various national and international Journals and conferences. He has guided more than 35 Diploma level Projects and 1 PG dissertations. He has published books with titles, Operating Systems, Database Management Systems, and Relational Database Management Systems. He has published 3 International Journal papers and 2 Conference papers. His fields of interest and research are Nature and Bio inspired Computing, Artificial Intelligence and Image Processing.

Professor Nitesh M. Sureja is a Diploma, B.E., and M.E., all in Computer Engineering. Currently he is pursuing his Ph.D. in Computer Engineering from NIMS University, Jaipur, Rajasthan, india. He has an industrial experience of four years in Raj Infotech as a Sales and Service engineer of Computer Systems and Peripherals. He worked in CCET Wadhwan City (Saurashtra University) as Lecturer. He also worked in SPCE Visnagar (Hemchandracharya North Gujarat University) as Lecturer and incharge of the Department of Computer and IT. His current profile comprises of Associate Professor (Computer Engineering) G.H. Patel College of Engineering and Technology, Vallabh Vidyanagar. He is a life member of ISTE and member of IEANG and IMETE. He has been offering his services as reviewer of various national and international Journals such as IJCSI and UBICC, and reviewer of various international and national conferences. He has guided more than 50 UG Projects and 2 PG. dissertations. He has published 4 International Journal papers and 3 Conference papers. His fields of interest and research are Swarm Intelligence, Nature and Bio inspired Computing and Artificial Intelligence.