VECTOR ANT COLONY OPTIMIZATION AND TRAVELLING SALESMAN PROBLEM

Chiranjib Patra[1], Pratyush[2]

[1]Department of Information Technology
Calcutta Institute of Engineering and Management,Kolkata-40
chiranjibpatra@gmail.com
[2]Department of Information Technology
Jadavpur University Sec-3, Block-LB/8, Salt-Lake, Kolkata- 700098
pratyush.cse1987@gmail.com

ABSTRACT

This paper introduces Vector Ant Colony Optimization (VACO), a distributed algorithm that is applied to solve the traveling salesman problem (TSP). In Any Colony System (ACS), a set of cooperating agents called ants cooperate to find good solutions of TSPs. Ants cooperate using an indirect form of communication mediated by pheromone they deposit on the edges of the TSP graph while building solutions. The proposed system (VACO) based on basic ACO algorithm with well distribution strategy in which the entire search area is initially divided into 2^n *number of hyper-cubic quadrants where n is the dimension of search space for updating the heuristic parameter in ACO to improve the performance in solving TSP. From our experiments, the proposed algorithm has better performance than standard bench mark algorithms.*

KEYWORDS

Ant colony optimization, traveling salesman problem, pheromone, global minima, VACO.

1. INTRODUCTION

In recent years, many research works have been devoted to ant colony optimization (ACO)[1,2,3,9] techniques in different areas. It is a relatively novel meta-heuristic technique and has been successfully used in many applications especially problems in combinatorial optimization. ACO algorithm models the behavior of real ant colonies in establishing the shortest path between food sources and nests. Ants can communicate with one another through chemicals called pheromones in their immediate environment. The ants release pheromone on the ground while walking from their nest to food and then go back to the nest. The ants move according to the amount of pheromones, the richer the pheromone trail on a path is, the more likely it would be followed by other ants. So a shorter path has a higher amount of pheromone in probability, ants will tend to choose a shorter path. Through this mechanism, ants will eventually find the shortest path. Artificial ants imitate the behavior of real ants, but can solve much more complicated problem than real ants can. [14,15].

Natarajan Meghanathan et al. (Eds) : ICCSEA, SPPR, VLSI, WiMoA, SCAI, CNSA, WeST - 2014
pp. 31–39, 2014. © CS & IT-CSCP 2014 DOI : 10.5121/csit.2014.4711

ACO has been widely applied to solving various combinatorial optimization problems such as Traveling Salesman Problem (TSP)[4], Job-shop Scheduling Problem (JSP), Vehicle Routing Problem (VRP), Quadratic Assignment Problem (QAP), etc. Although ACO has a powerful capacity to find out solutions to combinational optimization problems, it has the problems of stagnation and premature convergence and the convergence speed of ACO is very slow. Those problems will be more obvious when the problem size increases. Therefore, several extensions and improvements versions of the original ACO algorithm were introduced over the years.

We have proposed new optimization algorithm (vector ant colony optimization) based on ACO for solving travelling salesman problem both for discrete and continuous domains. The search in this optimization technique uses number of ants, dimension, number of iteration, upper pheromone value, and lower pheromone value and the search process of the optimization is directed towards the region of hypercube in a multidimensional space where the amount of pheromone deposited is maximum after predefined number of iterations. The entire search area is initially divided into 2^n number of hyper-cubic quadrants where n is the dimension of search space. Each ant traverse the path equals to number of iteration time. We short the coordinates according to the distance from the source. The node which is the nearest from the source has maximum dimension (i.e. number of nodes in the simulation -1) and the node which has largest distance from the source has lowest dimension (i.e. zero). Here, dimension of a point denotes that a node have a path to how many nodes, Or how many unexplored edges it has. The VACO system uses pheromones updates to find the shortest path from source to destination. After a number of iteration we find global minima for source to destination. A global minimum is found with the help of artificial ants. When we call the algorithm a number of artificial ants uses best ant technique to find the path and based on which we found the global minimum of that path.

The main objective of our work is applying VACO in Travelling Salesman Problem and compares it with other bench mark algorithms. The paper is organized as follows 1. Introduction. 2. ACO Background 3.Travelling Salesman Problem 4.Vector Ant Colony Optimization 5. Using VACO for TSP tour construction. 6. Comparison of performance for VACO TSP implementation versus other algorithm 7. Conclusion

2. ACO BACKGROUND

Ant System was first introduced and applied to TSP by Marco Dorigo [5, 9]. Initially, each ant is randomly put on a city. During the construction of a feasible solution, ants select the following city to be visited through *a probabilistic decision rule*. When an ant *k* states in city *i* and constructs the partial solution, the probability moving to the next city *j* neighboring on city *i* is given by $\tau_{ij} = (1-\rho)*\tau_{ij} + \Delta * \tau_{ij}$ (h Є N^k) where, $\tau_{i,j}^{\alpha}$ is the intensity of trails between edge (i and j) and η_{ij} is the heuristic visibility of edge (i, j), and $\eta_{ij}=1/d_{ij}$.N^k is a set of city which remains to be visited when the ant is at city *i*. α and β are two adjustable positive parameters that control the relative weights of the pheromone trail and of the heuristic visibility.

After each ant completes its tour, the pheromone amount on each path will be adjusted with equation

$$\tau_{ij} = (1-p)*\tau_{ij} + \Delta_{\tau_{ij}}$$

*(1-p)*is the pheromone decay parameter (*0<p<1*) where it represents the trail evaporation when the Ant chooses a city and decide to move.

$\Delta_{\tau_{ij}}$ is defined as

$\Delta_{\tau_{ij}}$ = F(k),if edge (*i*, *j*) is part of the solution constructed by ant *k*,

0 otherwise,

F(k)=1/L_k where L_k is the cost of k^{th} ant tour.

3. TRAVELING SALESMAN PROBLEM (TSP)

The traveling salesman problem (TSP)[4] is the problem of finding a shortest closed tour which visits all the cities in a given set. In this article we take the data set of China, Greece, Burma and Argentina and we assume the TSP graph is completely connected.

TSP asks for the shortest roundtrip of minimal total cost visiting each given city (nod) exactly once. TSP is an NP- hard[11,12,13,16] problem and it is so easy to describe and so difficult to solve. The definition of a TSP is: given N cities, if a salesman starting from his home city is to visit each city exactly once and then return home, find the order of a tour such that the total distances (costs) traveled minimum. Cost can be distance, time, money, energy, etc. or a combination of two or more factor. In this paper, we assume that the distance between two cities is their Euclidean distance. Namely, each distance between cities i and j is d(i, j)=d(j, i)= $\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$. Given a tour T, TSP is to find a tour which minimizes the objective function S:S =$\sum d(i,j)$.

4. VECTOR ANT COLONY OPTIMIZATION

We have proposed an efficient ant colony optimization function namely Vector ant colony optimizations (VACO) technique for optimizing mathematical functions. The search process of the optimization approach is directed towards the region of hypercube in a multidimensional space where the amount of pheromone deposited is maximum after predefined number of iterations. The entire search area is initially divided into 2^n number of hyper cubic quadrants where n is the dimension of search space. Then the pheromone level of each quadrant is measured. Now the search jumps to the new region of max pheromone level and restarts the search process in the new region. However the search area of new region is reduced compared to the previous search area. Thus the search advances and jumps to anew search space with reduced search area in several stages until the algorithm is terminated. The space of the new search region is smaller than the previous hyper-cubic search area. The reduction of search space is done along all dimensions. The pace is reduced in multiple stages with progress of the search process. If the search space is reduced slowly, then the possibility to come out of local optima and the convergence possibility to the global optimum are increased. On the other hand if the search space is reduced faster then there is a possibility to miss the global optimum since the process has no back tracking capabilities.

4a. Function optimization using VACO

The global optimization problem can generally be formulated as a pair (S, f) where S is the subset of R^n is abounded set on R^n and f: S --> R is an n – dimensional real valued function. The objective of the problem is to find a point x_{opt} belongs to S on R^n such that $f(x_{opt})$ is a global optimum on S. We have to find x_{opt} belongs to S according to the following equation, for min or max problems respectively:

$\forall$X that is subset of S: $f(x_{opt}) <= f(x)$………………(1)
$\forall$X that is subset of S :$f(x_{opt}) >= f(x)$……………(2)

Where f may not be continuous, but bounded.

Initially VACO algorithm starts searching to find the optimum in the entire search space. The search space is divided into a number of quadrants depending on the problems dimensionality in the multi-dimension space where each quadrant will form a hypercube. The search space partitioning is necessary to measure the pheromone level in each partition. If the problem dimension is denoted by n then the number of quadrants will be calculated as follows.

$q=2^n$ ……………………………….(3)

The VACO method runs for a certain number of iterations say I_k and measures the pheromone level in each quadrant after the completion of I_k iterations. The pheromone level is measured to direct the search process towards the area with maximum amount of pheromone. We have considered the amount of pheromone deposited in each iteration as p defined by the following expression p=1/n.

VACO technique runs iteratively in multiple stages and we find the quadrant in each iteration in which the best value of that iteration lies. Then the pheromone level of the corresponding quadrant is increased. The amount p_j in the j^{th} quadrant where j $\in$\{1,2,3……\} is increased by i/n in each iteration. Once the optimization method completes I_k iteration the amount of pheromone p_j deposited in each quadrant is calculated. If the amount of deposited pheromone p_m in the m^{th} ($1<=m<=2^n$) quadrant is maximum, the search then moves towards the m^{th} quadrant. The search space is re-defined around .The ant population is regenerated except the elite one. The VACO approach restarts in the new search area and continues for I_k times before its transferred to another new space considering the highest pheromone level. The VACO finally terminates on the completions of I_{max} iterations.

4b. Algorithmic representation of VACO technique

STEP 1: Initialize the population of ants and other parameters and iteration I=1;
STEP 2: Create solutions for all ants and partition the search space into 2^n quadrants.
STEP 3: Find the quadrant where the best solution lies in each iteration. Increase the pheromone level of the corresponding quadrant by 1/n following STEP 4.
STEP 4:Identify the quadrant q_m with highest amount of pheromone deposited after the completion of I_k iterations.
STEP 5: Refine the search space surrounding the area of the quadrant q_m and regenerate the population of ant following the elitist model. Move the search to the new search space

which is smaller in size than the previous search space and restart the search space process.

STEP 6: Increment $I=I+I_k$ if $I <= I_{max}$ then go to STEP 2.
STEP 7: Stop.

4c. Example of Search Space Reduction used in VACO method

Here we take 2-D search space and the repetitive reduction is described. In FIGURE –IA it's shown for the reduction of search space every after I_k (here we take 10) in three steps. Initially the highest number of solutions is in 2nd quadrant hence the intensity of pheromone in 2nd quadrant is 6*(1/2) and where as 2*(1/2),1/2,1/2 for 1st, 3rd and 4th respectively, so VACO jumps to 2nd quadrant as it contains highest intensity of pheromone. Once this is determined, now let's reduce the search space by u% on both X and Y axis, thus generating X' and Y' respectively Figure II we thus again partition the space into four quadrants, again we find the 1st quadrant contain large number of solutions and then VACO jumps to 1st quadrant unlike previous step which in accordance with the concurrence with an elitist model of ACO. From Figure-IC again a new space is designated and a set of solutions exists in the respective four quadrants viz. . x21……x30.. Now the pheromone concentration measurement is not necessary as the subdivision of the search space will not be economic as all the values are very close to one another. A chosen value close to the average of the found solutions is taken, which becomes the near-global optimum solution of the problem at the end of the final stage or at end of pre-specific number of iteration.

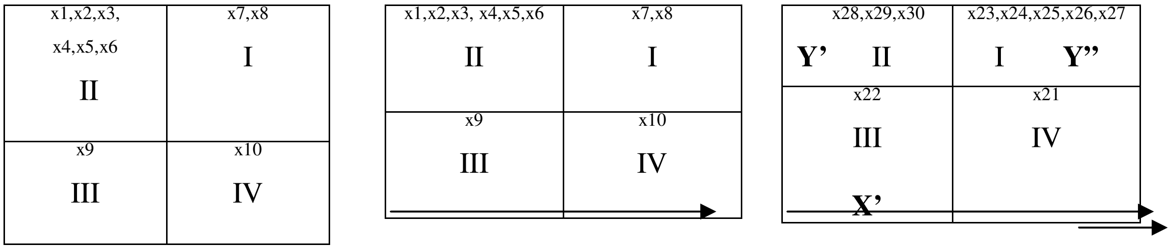

FIGURE-IA

FIGURE-IB

FIGURE-IC

Fig-1: The implementation VACO on hypercube

5. USING VACO FOR TSP TOUR CONSTRUCTION

Considering the above assumptions we deduce the algorithm for TSP construction
Algorithmic representation:

STEP 1: Initialize the co-ordinates.
STEP 2: Initialize the starting position.
STEP 3: Calculate the distance between starting node and all neighboring node.
STEP 4: Sort the coordinates point on the basis of distance from source.
STEP 5: Initialize the dimension for each sorted coordinates.
STEP 6: Call the vector ant colony optimization for each sorted coordinates.
STEP 7: Find out the coordinate which has lowest amount of global minima.
STEP 8: New source is the coordinates which has the lowest minima from the source.

STEP 9: New travelling salesman problem space is STEP 1 coordinates point minus STEP 7 coordinate point.
STEP 10: Go to STEP 3.
STEP11: Stop.

In the given Fig 2, it is shown that we start from a point (x_i , y_i) and move towards that point that point has minimum value of global minima. By doing this technique iteratively we build a tour for TSP. The cost of this tour is the summation of all global minimum value between two connected points

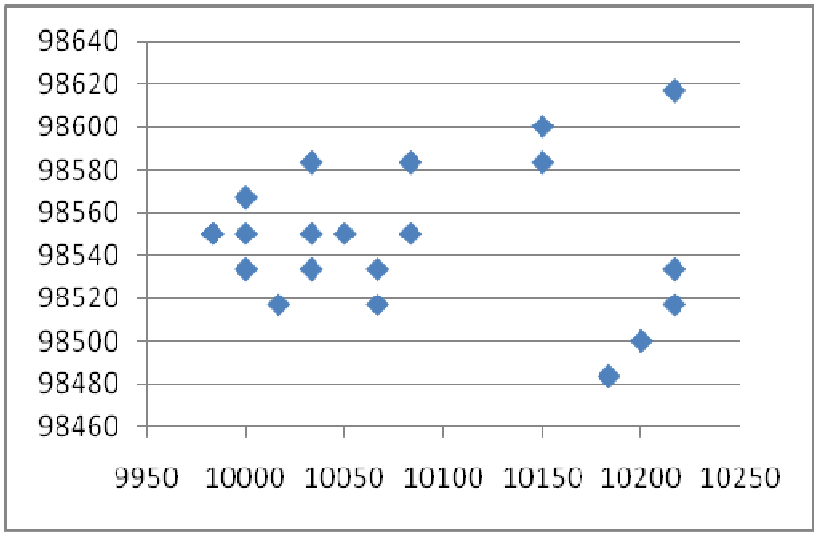

Fig-2A

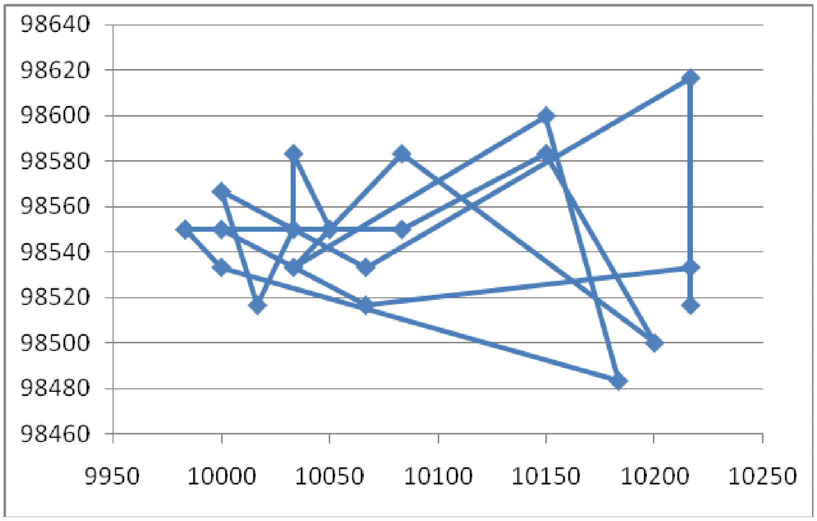

Fig-2B

Fig 2: Graphical Representation of 20 test nodes in space

The performance of VACO algorithm on a set of point is evaluated to find out how tour construction is efficient than other algorithms. In this setup we have considered a set of population size of ants as 50 till the end of the proposed optimization algorithm. We take 20 nodes for illustrating tour construction for which pheromone upper and lower limits are respectively +100 and -100 respectively. We select the path which has lowest value of global minima, we add that path to our resultant solution and dimension is indicated in the table. We did this procedure iteratively to complete the tour.

TSP Solution using Ant Colony Optimization Algorithm:

TSP function	Search Space	Dimension	Optimum value
	[9983.3333,10216.6667] [98483.3333,98616.6667]	19	793.3227
	[10000, 10216.6667] [98483.3333,98616.6667]	18	784.6709
	[10000, 10216.6667] [98483.3333,98616.6667]	17	789.0375
	[10000, 10216.6667] [98483.3333,98616.6667]	16	831.794
	[10000, 10216.6667] [98483.3333,98616.6667]	15	847.5783
	[10000, 10216.6667] [98483.3333,98616.6667]	14	889.6033
	[10000, 10216.6667] [98483.3333,98616.6667]	13	853.8038
	[10000, 10216.6667] [98483.3333,98616.6667]	12	811.2816
	[10000, 10216.6667] [98483.3333,98600]	11	886.8193
	[10000, 10216.6667] [98483.3333,98600]	10	892.7845
	[10000,10200] [98483.3333,98600]	9	907.407
	[10000,10200] [98483.3333,98600]	8	876.8203
	[10000,10200] [98483.3333,98600]	7	881.545
	[10000,10200] [98483.3333,98600]	6	962.8069
	[10000,10200] [98483.3333,98600]	5	889.4022
	[10000,10183.3333] [98483.3333,98600]	4	918.7506
	[10000,10183.3333] [98483.3333,98600]	3	852.6672
	[10000,10183.3333] [98483.3333,98600]	2	988.7577
	[10000, 10183.3333] [98483.3333,98533.3333]	1	986.0945
	[9983.3333,10000] [98533.3333,98550]	1	992.3018
Sum of global minima for all path			17637.25

6. Comparison of Performance For VACO TSP Implementation Versus Other Algorithm

The performance of VACO technique on a TSP functions was evaluated to find out how VACO is efficient than other algorithm.

In this setup we have considered a set of population size of ants as 50 till the end of the proposed optimization algorithm. For each algorithm VACO was run 50 times and an average of the 50 optimum results is tabulated in table 1 for VACO method. We have run VACO technique for I_k iterations to measure the amount of pheromone deposited in each quadrant for directing the search in the redefined search space for the next stage. The new search space is generated by reducing U% of the length in all directions of the previous search space. For all algorithms tested for the TSP function , we have considered the value of I_k to be 5 and the value of u is set to 25 . If the value of I_k is increased towards the higher side or more of the range, the search process slows down ie the convergence will require more number of iteration. On the other hand the value of u set to more than 25, the VACO may not converge to global or near global optimum solutions .Due fast reduction of the search space after I_k iteration. In this experiment we have tied to maintain the values I_k and u within the specified range so that VACO converges fast.

Function	Algorithm	MFE	G_{min}	SD
TSP function	OGA/Q M-L LEA VACO	167,836 13700 168,910 15,550	$7.56*10^{-1}$ 2934.78 $5.5*10^{-1}$ 0	$1.1*10^{-1}$ 133.674 $1.08*10^{-1}$ $1.77*10^{-13}$

MFE denotes the average number of function evaluation to reach the desired value

G_{min} indicates the mean of the best value found in last generation for 50 runs

SD denotes the Standard Deviation

Table -1 Comparative study of VACO with other benchmark algorithm

7. CONCLUSION

In this work we studied the performance of Travelling Salesman Problem (TSP) using Vector Ant Colony Optimization. Simulation results revealed that TSP solution using our VACO method is capable of providing remarkably improved performance compared to other solution of TSP using other Optimization techniques. Similarly unlike traditional ACO the following statement is valid as based on the experiments, it can be seen that the quality of solutions depends on the number of ants. The lower number of ants allows the individual to change the path much faster. The higher number of ants in population causes the higher accumulation of pheromone on edges, and thus an individual keeps the path with higher concentration of pheromone with a high probability.

REFERENCES

[1] Ramlakhan Singh Jadon, Unmukh Datta "Modified Ant Colony Optimization Algorithm with Uniform Mutation using Self Adaptive Approach for Travelling Salesman Problem" 4th ICCCNT 2013

[2] A. Colorni, M. Dorigo, V. Maniezzo, "Distributed optimization by ant colonies". Proceedings of European Conference on Artificial Life, Paris, France, pp. 134-142.1991.

[3] Krishna H. Hingrajiya, Ravindra Kumar Gupta, Gajendra Singh Chandel, "An Ant Colony Optimization Algorithm for Solving"International Journal of Scientific and Research Publications, Volume 2, Issue 8, August 2012.

[4] Zar Chi Su SuHlaing, May Aye Khine"An Ant Colony Optimization Algorithm for Solving Traveling Salesman Problem"2011 International Conference on Information Communication and Management.

[5] Marco Dorigo, Luca Maria Gambardella,"Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem"IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, VOL. 1, NO. 1, APRIL 1997.

[6] Zulaiha Ali Othman, Helmi Md Rais, and Abdul Razak Hamdan "DACS3:Embedding Individual Ant Behavior in Ant Colony System"International Journal of Computer, Information Science and Engineering Vol:2 No:8, 2008. [7] Marco Dorigo, Mauro Birattari, and Thomas Stutzle "Ant Colony Optimization: Artificial Ants as a Computational Intelligence Technique".2001

[8] Alberto Colorni, Marco Dorigo, Vittorio Maniezzo Dipartimento di Elettronica, Politecnico di Milano Piazza Leonardo da Vinci 32, 2013 Milano, Italy "Distributed Optimization by Ant Colonies"- EUROPEAN CONFERENCE ON ARTIFICIAL LIFE,PARIS,FRANCE ,ELSEVIER PUBLISHING, 134–142

[9] Thomas Stutzle, Manuel Lopez-Ibanez, Paola Pellegrini, Michael Maur, Marco Montes de Oca, Mauro Birattari, and Marco Dorigo "Chapter 8 Parameter Adaptation in Ant Colony Optimization"

[10] Bifan Li, Lipo Wang, and Wu Song "Ant Colony Optimization for the Traveling Salesman Problem Based on Ants with Memory" 2008 IEEE.

[11] David L. Applegate, Robert E. Bixby ,Vasek Chvatal, William J. Cook. The Traveling Salesman Problem: A Computational Study . Princeton University Press, USA, 2007.

[12] Elmedina Fejzagic, Adna Oputic "Performance Comparison of Sequential and Parallel Execution of the Ant Colony Optimization Algorithm for Solving the Traveling Salesman Problem" MIPRO 2013, MAY 20-24, 2013.

[13] Dorigo M., Maniezzo V., Colorni A.: The Ant System: An autocatalytic optimizing process. Tech. Rep. 91-016 Revised, Dipartimento di Elettronica, Politecnicodi Milano, Italy (1991).

[14] Dorigo M., Maniezzo V., Colorni A.: Ant System: Optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics- Part B 26(1):29–41 (1996).

[15] H. Md. Rais, Z. A. Othman, A.R. Hamdan, Improvement DACS3 Searching Performance using Local Search, Conference on Data Mining and Optimization, IEEE, 27-28 October 2009.

[16] Ahuja, A. & Pahwa, A. (2005). Using ant colony optimization for loss minimization in distribution networks, Proceedings of the 37th Annual North American Power Symposium , pp. 470–474.

AUTHORS

Chiranjib Patra received his undergraduate and post graduate degrees from Calcutta University and Jadavpur University. Currently he is the candidate of PhD at the department of Information Technology, Jadavpur University. His is currently working as the Assistant Professor and Head of the Department, Information Technology at Calcutta Institute of Engineering and Management.
His interests are in application of evolutionary computing in wireless sensor networks and scale free networks.

Pratyush received his undergraduate degree from MMMEC, Gorakhpur and currently he is the candidate of M.E. (Software Engineering) at the Department of Information Technology, Jadavpur University.
His interests is in evolutionary computing and algorithms