An Ant Colony Optimization Algorithm for Solving Traveling Salesman Problem

Zar Chi Su Su Hlaing, May Aye Khine
University of Computer Studies, Yangon
zarchissh@gmail.com, maya.khine@gmail.com

Abstract

Ant Colony Optimization (ACO) is a class of heuristic search algorithms that have been successfully applied to solving combinational optimization (CO) problems. The traveling salesman problem (TSP) is among the most important combinatorial problems. ACO has very good search capability for optimization problems. But it still has some drawbacks such as stagnation behavior, long computational time, and premature convergence problem of the basic ACO algorithm on TSP. Those problems will be more obvious when the complexities of the considered problems increase. The proposed system based on basic ACO algorithm based on well-positioned the ants on the initiation and information entropy which is applied to tuning of the algorithm's parameters. Then, ACO for TSP has been improved by incorporating local optimization heuristic. Therefore, the proposed system intends to reach superior search performance over traditional ACO algorithms do.

1. Introduction

In recent years, many research works have been devoted to ant colony optimization (ACO) techniques in different areas. It has been successfully used in many applications especially problems that belong to the combinatorial optimization.

ACO has been inspired by the observation on real ant colony's foraging behavior, and that ants can often find the shortest path between food source and their nest. One of the main ideas behind this approach is that the ants can communicate with one another through indirect means by making modifications to the concentration of highly volatile chemicals called pheromones in their immediate environment. So, the ants release pheromone on the ground while walking from their nest to food and then go back to the nest. Since a shorter path has a higher amount of pheromone in probability, ants will tend to choose a shorter path. Artificial ants imitate the behavior of real ants how they forage the food, but can solve much more complicated problem than real ants can.

ACO has been widely applied to solving various combinatorial optimization problems such as Traveling Salesman Problem (TSP], 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 those problems will be more obvious when the problems increase. In the proposed system, the main modifications introduced by *ACO* are the following. First, to avoid search stagnation, the system places ants with at least one ant on each node on the initiation. Second, information entropy is introduced which is adjust the algorithm's parameters. Additionally, the best performing ACO algorithms for the TSP improve the solutions generated by the ants using local search algorithms.

2. Related Work

Cheng-Fa Tsai and Chun-Wei Tsai proposed an algorithm on the basic of the ant evolution rules. In addition, a method called nearest neighbor (NN) to EA to improve TSPs and obtain good solution quickly.

Kuo-Shen Hung, Shun-Feng Su and Zne-Jung Lee [3] proposed the analysis of lower pheromone trail bound and a dynamic updating rule for the heuristic parameters based on entropy to improve the efficiency of ACO. SHU Yunxing et.al [4] presented an ACO based on basic ACO algorithm on nearest neighbor node choosing rules and with crossover operator to increase the convergence speed of the ACO. Rongwei Gan et.al [5] is to partition artificial ants into two groups: scout ants and common ants for solving the problems of basic ACO algorithm.

All these studies have contributed to the improvement of the ACO to some extent, but they have little obvious effect on increasing the convergence speed and obtaining the global optimal solution.

3. Ant Colony Optimization

The Ant Colony Optimization (ACO) techniques has emerged recently as a relatively novel meta-heuristic for hard combinational optimization problems. It is designed to simulate the ability of ant colonies to determine shortest paths to food. Although individual ants posses few capabilities, their operation as a colony is capable of complex behavior.

Real ants can indirectly communicate by pheromone information without using visual cues and are capable of finding the shortest path between food sources and their nests. The ant deposits pheromone on the trail while walking, and the other ants follow the pheromone trails with some probability which are proportioned to the density of the pheromone. The more ants walk on a trail, the more pheromone is deposited on it and more and more ants follow the trail. Through this mechanism, ants will eventually find the shortest path. Artificial ants imitate the behavior of real ants how they forage the food, but can solve much more complicated problems than real ants can. A search algorithm with such concept is called Ant Colony Optimization [2]. Figure 1 shows how the ants find the shortest path [7].

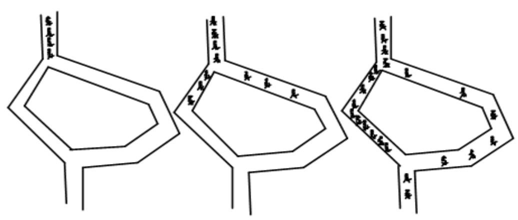


Figure 1: Sketch Map of the Ant Theory

3.1 Tour Construction

At each construction step in ACO, ants choose the next city using the pseudorandom-proportional action choice rule. When located at city i, ant k chooses the next city j according to this equation

$$j = \begin{cases} \arg\max_{u \in allowed_k(i)} \{ [\tau_{ij}]^{\alpha} . [\eta_{ij}]^{\beta} \} & \text{if } q < q_0 \\ J & \text{otherwise} \end{cases} \quad ...(1)$$

where q is a random variable uniformly distributed in [0,1]. $q_0 (0 \le q_0 \le 1)$ is a pre-defined parameter the pheromone trail, $\eta_{ij}=1/d_{ij}$ is the heuristic information (heuristic visibility), where d_{ij} is the distance between city i and city j, α and β are are two adjustable positive parameters that control the relative weights of the pheromone trail and of the heuristic information, and $allowed_k(i)$ is the set of unvisited cities yet when ant k is located at city i. J is selected according to the transition probability given by [1]

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha} [\eta_{ij}]^{\beta}}{\sum_{l \in J_k(i)} [\tau_{il}(t)]^{\alpha} [\eta_{il}]^{\beta}} & if \;\; j \in J_k(i) \\ 0 & otherwise \end{cases} \quad ...(2)$$

3.2 Pheromone Update

In order to improve future solutions, the pheromone trails of the ants must be updated to reflect the ant's performance and the quality of the solutions found. This updating is a key element to the adaptive learning technique of ACO and helps to ensure improvement of subsequent solutions. Trail updating includes local updating and global updating.

Local updating of pheromone: After each ant has chosen a city, the amount of pheromone on each side will be updated according to equation

$$\tau_{ij}(t+1) = (1-\rho)\,\tau_{ij}(t) + \rho.\tau_0 \qquad (3)$$

where 0< ρ<= 1 is a decay parameter, $\tau_0=1/n.L_{nn}$ is the initial values of the pheromone trails, where n is the number of cities in the TSP and L_{nn} is the cost produced by the nearest neighbor heuristic. Eq. (3) is mainly to avoid very strong pheromone paths to be chosen by other ants and to increase the explorative probability for other paths. Once the edge between city i and city j has been visited by all ants, the local updating rule makes pheromone level diminish on the edge. So, the effect of the local updating rule is to make an already edge less desirable for a following ant.

Global updating of pheromone. After all the ants have travelled through all the cities, update only the amount of the pheromone on the optimal path according to this equation

$$\tau_{ij}(t+1) = (1-\rho)\,\tau_{ij}(t) + \Delta\tau_{ij}(t) \qquad (4)$$

In this equation,

$$\Delta\tau_{ij}(t) = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}(t) \qquad (5)$$

$$\Delta\tau_{ij}^{k}(t) = \begin{cases} \dfrac{1}{L_k} & \text{, if the global best result is through path ij} \\ 0 & \text{otherwise} \end{cases}$$

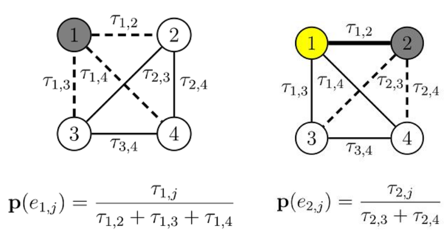
The ant colony optimization algorithm can briefly described as follows:

Procedure ACO algorithm for TSP
Set parameters, initialize pheromone trails
Loop
    Each ant is positioned on a starting node
    Loop
        Construct Solutions
        Apply Local Search
        Local_Pheromone_Update
    Unitl all ants have built a complete solution
    Global_Pheromone_Update
Until End_condition
End ACO algorithm for TSPs

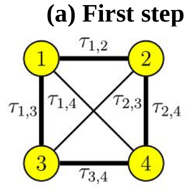
4. Traveling Salesman Problem

The traveling salesman problem (TSP) is a well known optimization problem. TSP defines the task of finding a tour of minimal total cost given a set of fully connected nodes (cities) and costs associated with each pair of nodes. The tour must be closed and contain each node exactly once.

Concerning the ACO approach, the edges of the given TSP graph can be considered solution components, i.e., for each $e_{i,j}$ is introduced a pheromone value $\tau_{i,j}$. The task of each ant consists in the construction of a feasible TSP solution, i.e., a feasible tour. In other words, the notion of task of an ant changes from “choosing a path from the nest to the food source” to “constructing a feasible solution to the tackled optimization problem” [8].



(a) First step **(b) Second step**



(c) The complete solution after the final construction step

Figure 2: Example of the solution construction for a TSP problem consisting of 4 cities (modelled by a graph with 4 nodes). The solution construction starts by randomly choosing a start node for the ant; in this case node 1. Figures (a) and (b) show the choices of the first, respectively the second, construction step. Note that in both cases the current node of the ant is marked by dark gray color, and the already visited nodes are marked by light gray color (respectively yellow color). The choices of the ant (i.e., the edges) are marked by dashed lines. The probabilities for the different choices (according to Eq. (1))are given underneath the graphics. Note that after the second construction step, in which we exemplary assume the ant to have selected node 4, the ant can only move to node 3, and then back to node 1 in order to close the tour.

5. Overview of the Proposed Algorithm

5.1 Positioning of initial ants

In the ACS algorithm, ants are put onto their start cities in the initiation at each step of construction, and select the next city to be visited according to the equation (1) and thus some cities may have many ants while some cities may have no ant at all. Because the amount of pheromone on each path is initially identical, therefore the ant mainly uses the distance between the two cities as the heuristic factor when it chooses the next city. In this way, when there are relatively more ants in a certain city, the density of the pheromone on a certain path will be strengthened due to the relatively larger number of ants travelling along the path. However, the path is not necessarily the shortest path.

In order to solve this problem, we adopted a method to distribute the ants evenly, i.e., position m ants to n cities and make sure that each city receives at least one ant (suppose m≥n). Thus, the search space of the solution is enlarged and the probability of getting the best result is increased.

5.2 Analysis of Parameter Tuning

The concept of entropy is known from Shannon’s information theory. It is a measure of uncertainty concerning an event and is used to denote the degree of disorder in a system. Shannon’s entropy represents the information regarding the probability of occurrence of an event. In ACO, pheromone is the basis of path selection, and the selection is uncertain in nature. Thus, we propose to consider the entropy information in ACO to estimate the variation of the pheromone matrix. Each trail is a discrete random variable in the pheromone matrix. The entropy of a random variable X is defined as

$$H(X) = -\sum_{t=1}^{r} P(x_t)\log P(x_t) \tag{6}$$

where X represents the trails in the pheromone matrix. For a symmetric n cities TSP, there are n(n-1)/2 distinct pheromone trails and r= n(n-1)/2. It is easy to see that when the probability of each trail is the same, H will be the maximum (denoted as H_{max}) and is given by

$$H_{\max} = -\sum_{i=1}^{r} P_i \log P_i = -\sum_{i=1}^{r} \frac{1}{r} \log \frac{1}{r} \qquad (7)$$

We propose to use the entropy value as an index to indicate the degree about how much information has been learned into the pheromone trails and then the heuristic parameter can be tuned accordingly. Notice that in this study, the heuristic parameter β is set to be an integer so as to avoid complicated computation because β is used as a power in Eqs. (1) and (2). Hence, we propose that β is tuned according to the rule given by

$$\beta = \begin{cases} 5 & \text{threshold A} < H' \le 1 \\ 4 & \text{threshold B} < H' \le \text{threshold A} \\ 3 & \text{threshold C} < H' \le \text{threshold B} \\ 2 & 0 < H' \le \text{threshold C} \end{cases} \qquad (8)$$

where $H' = H_{iteration}/H_{max}$, $H_{iteration}$ is the entropy value for the current pheromone matrix and A B, and C are thresholds.

5.3 Local Search for the TSP

Local Search starts from some initial assignment and repeatedly tries to improve the current assignment by local changes. If in the neighborhood of the current tour *T* a better tour *T*' is found, it replaces the current tour and the local search is continued from *T*'. The most widely known iterative improvement algorithms for the TSP are certainly 2-opt and 3-opt. They proceed by systematically testing whether the current tour can be improved by replacing 2 or at most 3 arcs, respectively.

5.4 The Proposed Algorithm

Initalize
Position a number of ants, each node has at least one ant
Set $\tau_{ij}(0) = \tau_0 = (nL_{nn})^{-1}$
Calculate the maximum entropy
While stopping criterion not satisfied do
  For k=1 to m **do**
    At the first step moves each ant at different route
    Repeat
      Select node j to be visited next (the next node must not be visited by the ant)
      Apply local updating rule
    Until ant k has completed a tour
  End for
  Local search apply to improve tour (2-opt or 3-opt)
  Apply global updating rule
  Compute iteration entropy of pheromone trails
  Update the heuristic parameter
End while
End

6. Conclusion

This paper proposed an ant colony algorithm to improve search efficiency. In the initiation, the algorithm puts all ants into various cities to obtain the global optimal solution and introduces the entropy rule to adjust algorithm's parameters. Then, local search applied tour improvement. So, the proposed system reach the superior search performance over traditional ACO algorithms do.

7. References

[1] M. Dorigo and L. M. Gambardella, "The colony system: A cooperative learning approach to the traveling salesman problem," IEEE Transactions on Evolutionary Computation, Vol.1, No.1, April,1997.

[2] C. F. Tsai, C. W. Tsai. "A New Approach for Solving Large Traveling Salesman Problem Using Evolutionary Ant Rules", In: Neural Networks, IJCNN 2002,Proc.of the 2002 Int'l Joint Conf. on, Vol 2.Honolulu:IEEE Press,2002.1540~1545.

[3] K. S. Hung, S.F. Su, S. J. Lee, "Improving Ant Colony Optimization for Solving Traveling Salesman Problem", Jouranl of Advanced Computational Intelligence and Intelligent Informatics, 2007.

[4] S. Yunxing, G. Junen, G. Bo, " An Ant Optimization based on the Nearest Neighbor Node Choosing Rules and Crossover Operator", IEEE Computer Society, 2008.

[5] R. Gan, Q. Guo, H. Chang, Y. Yi, " Improved Ant Colony Optimization Algorithm for the Traveling Salesman Problems", Jouranl of Systems Engineering and Electronics, April 2010, pp 329-333.

[6] G.Ma, H.Duan, S.Liu, " Improved Ant Colony Algorithm for Golbal Optimal Trajectory Planning of UAV under Complex Environment", International Journal of Computer Science and Applications, Vol4 Issue 3, pp57-68.

[7] L.Min, J. Yant, " A Shortest Path Routing Based on Ant Algorithm", Journal of Communication and Computer, ISSN1548-7709, USA, September 2005, Volume 2, No.9.

[8] C.Blum, "Ant Colony Optimization: Introduction and recent trends", Science direct, Physics of Life Reviews 2(2005)353-373.