Route Optimisation by Ant Colony Optimisation Technique
Dhammpal Ramtaka, Sanjay Kumarb, V.K. Patlec
abc School of Study in Computer science & IT, Pt. Ravishankar Shukla University, Raipur (Chhattisgarh) 492010 India

Abstract

In communication networks, we need to find the shortest path frequently. For finding the shortest path there are several methods. Travelling salesman problem (TSP) is also a problem of finding a shortest path. TSP has several applications such as planning, logistics, manufacture of microchips, Flight simulation etc. There are several methods for solving TSP such as dynamic programming, branch and bound method and linear programming.

Here we analyze a Meta heuristics swarm intelligent approach to solve TSP approximately. An Ant Colony Optimization (ACO) is a Meta heuristics swarm intelligent optimization technique. An ACO is one of the best methods to find the shortest path. ACO uses parameters called alpha, beta (also called control parameters) and evaporation rate, to find the shortest path on probability basis. We have tried to optimize these parameters to find the path of minimum length and cost.

Keywords: Ant System; Traveling Salesman Problem; Parameters;

E-mail address: dhammpal88@gmail.com, sanraipur@rediffmail.com, patlevinod@gmail.com,
1. Introduction

The Travelling Salesman Problem (TSP) is a problem in combinatorial optimization studied in both, operations research and computer science. Given a list of cities and their pairwise distances, the task is to find a shortest possible route that visits each city exactly once. It was first formulated as a mathematical problem in 1930 and is one of the most intensively studied problems in optimization [1].

Ant Colony Optimization is a population-based metaheuristic for solving optimization problems, originally proposed by Dorigo and Di Caro [2]. ACO uses artificial ants to construct solutions by incrementally adding components that are chosen considering heuristic information of the problem and pheromone trails that reflect the acquired search experience [3]. The Basic algorithm structure of Ant Colony algorithm applied to a combinatorial optimization problem for minimizing the objective function is as under:-

ACO Algorithm
Set parameters, initialize pheromone trails
While (termination conditions not met) do
Construct_Ant_Solutions
Apply_Local_Search (optional)
Update_Pheromones
end while
In this structure first initialize the initial heuristic values in it’s parameters which is used for further computation of ACO algorithm. After that the algorithm computes again and again until the termination condition comes.

In each iteration step is divided in three stages.First, each ant of the colony concurrently, independently, and asynchronously constructs a solution by selecting components using a probabilistic rule that considers both the experience acquired during the search through the trace of pheromone deposited and heuristic information of the considered components (through the visibility).The next stage optionally applies a local search method to improve the solutions. In the third stage, the pheromone trails are updated the trace values are decreased by evaporation and increased by depositing pheromone in the components used to construct solutions the net change in the pheromone value depends on the contributions of these two update processes[3,4].

ACO have been applied extensively to a variety of discrete combinatorial optimization problems like the Travelling Salesman Problem, the Quadratic Assignment Problem and the Network Routing Problem. More recently, the approach has been extended to continuous search domains [5].

2. The Travelling Salesman Problem (TSP)

The TSP consists of a set of locations (cities) and a travelling salesman that has to visit all the locations once and only once. The distances between the locations are given and the task is to find a shortest path of minimal length. The travelling salesman problem is widespread in engineering applications. It has been employed in designing hardware devices and radio electronic devices, in communications, in the architecture of computational networks, etc. In addition, some industrial problems such as machine scheduling, cellular manufacturing and frequency assignment problems can be formulated as a TSP.
A general definition of the traveling salesman problem is the following. Consider a set \( N \) of nodes, representing cities, and a set \( C \) of paths fully connecting the nodes \( N \). Let \( C_{ij} \) be the length of the path \((i, j) \in C\), that is the distance between cities \( i \) and \( j \), with \( i, j \in N \). The TSP is the problem of finding a minimal length Hamiltonian circuit on the graph \( G = (N, C) \), where an Hamiltonian circuit of graph \( G \) is a closed tour visiting once and only once all the \( n = |N| \) nodes of \( G \), and its length is given by the sum of the lengths of all the path of which it is composed[6].

3. **Ant Colony System**

Ant System (AS) was the first ACO algorithm to be proposed by the M. Dorigo, V. Maniezzo, and A. Colorni. Stem [10]. Its main characteristic is that the pheromone values are updated by all the ants that have completed the tour. Thepheromone update for \( \tau_{ij} \) that is for edge joining cities \( i \) and \( j \) is performed as follows:

\[
\tau_{ij} = (1 - \rho) \tau_{ij} + \rho \sum_{k=1}^{m} \Delta \tau_{ij}^k \quad \text{........ (1.1)}
\]

Where \( \rho \) is the evaporation rate, \( m \) is the number of ants, and \( \Delta \tau_{ij}^k \) is the quantity of pheromone per unit length laid on edge \((i, j)\) by the \( k \)-th ant:

\[
\Delta \tau_{ij}^k = \begin{cases} 
Q / L_k & \text{if ant } k \text{ used edge}(i, j)\text{in its tour,} \\
0 & \text{Otherwise} 
\end{cases} \quad \text{ ........... (1.2)}
\]

Where \( Q \) is a constant and \( L_k \) is the tour length of the \( k \)-th ant, when constructing the solutions, the ants in AS traverse a construction graph and make probabilistic decision at each vertex. The transitional probability \( P_{ij}^k \) of the \( k \)-th ant moving from city \( i \) to city \( j \) is given by:

\[
P_{ij}^k = \begin{cases} 
\frac{\tau_{ij}^a \cdot \eta_{ij}^\beta}{\sum_{l \in \text{allowed } k} \tau_{il}^a \cdot \eta_{il}^\beta} & \text{if } j \text{ is allowed } k, \\
0 & \text{Otherwise} 
\end{cases} \quad \text{ ............ (1.3)}
\]

Where allowed\(_k\) is the list of cities not yet visited by the \( k \)-th ant and \( \alpha \) and \( \beta \) are parameters that control the relative
importance of the pheromone versus the heuristic information $\eta_{ij}$ given by:

$$\eta_{ij} = \frac{1}{d_{ij}}$$

Where $d_{ij}$ is the length of edge $(i, j)$.

4. Analysis of Problem

Here we define the Problem of finding minimum length Hamiltonian circuit of the graph, where a Hamiltonian circuit is a closed path visiting each of the nodes exactly ones. We have try to find the possible optimal solution for shortest path that takes him through a given set of nodes and then back, visiting each nodes exactly ones using the Ant system and impact of various control parameters by implementing ACO algorithm.

4.1 ACO Parameters

Here we take the parameter which is $0 < \rho < 1$ is the pheromone evaporation rate the parameter rho ($\rho$) is used to avoid unlimited accumulation of the pheromone trails and it enables the algorithm to forget bad decisions previously taken.

The role of the parameters $\alpha$ and $\beta$ is the following. If $\alpha > 0$, the closest cities are more likely to be selected this corresponds to a greedy algorithm (with multiple starting points since ants are initially randomly distributed over the cities). If $\beta > 0$, only pheromone increases at work that is only pheromone is used, without any heuristic bias. This generally leads to rather poor results and in particular for values of $\alpha > 1$ it leads to the rapid emergence of a stagnation situation that is a situation in which all the ants follow the same path and construct the same tour, which, in general, is strongly suboptimal [10, 13].

5. Proposed Acots Algorithm

**Step 1:** Initialize Parameters
- Rho = { set of values };
- Alpha = { set of values };
- Beta = { set of values };
- Ant = {Number of ants};
- Distance = {Distances of all city};
- Initial Pheromones = {Minimum value};

**Step 2:** Initial local search to find the minimum distance path
- Loop start = first ant to last ant
- Best Trail = set Ants moves to visits all the source
- Total Length = Best Trail
- If Minimum length
- Return Best Trail and Minimum length
- End of loop

**Step 3:** Ant System Steps Update Ants
- Traveling Ant = Ant visiting probability using the Equation 1.3
- Trail = Traveling Ant
- Ant = randomly assign city from trail
- End Update Ants

**Step 4:** Update Pheromones
Decreasing = First part from Equation 1.1
Increasing = Using the Equation 1.2
Initial Pheromones = decreasing + increasing
End Update Pheromones

**Step 5:** Minimum length path after update Pheromones and Ants
  - Best Trail = set Ants moves to visits all the source
  - Total Length = Best Trail
  - If Minimum length
  - Best Trail and Minimum length
  - End of Minimum length path

6. **RESULT AND DISCUSSION**

From the Ant Colony Optimization according to Marco Dorigo Thomas Stutzle on page no. 71 box 3.1 it is written that ‘Our experimental study Parameter Settings table for Ant System is \{α=1, β=2 to 5, ρ=0.5\}[13]. In our experiment we vary β from 1 to 10, and find the best value of β for TSP with 10 cities and 4 salesmen.

<table>
<thead>
<tr>
<th>Sequence No</th>
<th>Beta</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>7</td>
</tr>
</tbody>
</table>
Fig 2. Graph for Beta value

Table 2.(a) Average cycle time to find minimum length.

<table>
<thead>
<tr>
<th>No of Ants</th>
<th>No of city</th>
<th>α</th>
<th>β</th>
<th>ρ</th>
<th>Average cycle</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>10</td>
<td>1</td>
<td>3</td>
<td>0.5</td>
<td>8.33</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>8.18</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td>7.81</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>10</td>
<td>7.6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td>6.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>5.83</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2.(b) Average cycle time to find minimum length.

<table>
<thead>
<tr>
<th>No of Ants</th>
<th>No of city</th>
<th>α</th>
<th>β</th>
<th>ρ</th>
<th>Average cycle</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>10</td>
<td>8.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>7.66</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td>7.5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>10</td>
<td>6.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td>6.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td>5.83</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>5.66</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 2. (c) Average cycle time to find minimum length

<table>
<thead>
<tr>
<th>No of Ants</th>
<th>No of city</th>
<th>α</th>
<th>β</th>
<th>ρ</th>
<th>Average cycle</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td>7.66</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td>7.5</td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td>7.33</td>
</tr>
<tr>
<td>7</td>
<td>10</td>
<td>1</td>
<td>5</td>
<td>0.5</td>
<td>6.16</td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td>5.55</td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td>5.33</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td>4.5</td>
</tr>
</tbody>
</table>

Fig 3. Graph for β value

For α=1, β=5 and ρ=0.5 time taken to find shortest route is minimum upon increasing number of salesmen time taken to find shortest route decreases.

7. Conclusion

For the solution of travelling salesmen problem we have applied an ant colony optimization technique to find the shortest path. In ant colony optimization α, β, ρ are important parameters which are to be adjusted, for 10 cities and traveling salesman being 4 up to 10 salesman, we calculated the time to find shortest path in each condition subsequently, we arrived at final values of α, β and ρ which give minimum paths in minimum time, we found that α=1, β=5, ρ=0.5 give best performance. We conclude that the effect of parameters in Ant Colony Algorithm have great impact on algorithm performance. Good parameter combination will increase the overall route optimization performance.
References