Ant Colony Optimisation Meta Heuristics

Vinayak Sareen, Joseph McGeever, Sergiu Harjau, Alicja Osicka

1 Introduction

Ant colony is a optimisation problem inspired by foraging nature of ants to solve well known computer science combitoral problem Tsp(travelling sales person) problem. Combitoral optimisation problems intend to solve combination and permutation based problems in optimal combinations and permutations (Socha and Dorigo, 2008). The TSP purposes a problem which requires optimial path treversed in list of cities visited exactly once. The problem is NP-hard problem, which means that there doesnot exist a optimal polynomial solution for solving the problem (Socha and Dorigo, 2008). Heuristic based solution can be applied to solve such problem which doesnot ensure the best optimal path in covered in list of cities but solves the problem with descent solution in limited time constraints (Socha and Dorigo, 2008).

2 Ant Colony Optimisation

ACO(Ant colony optimisation) algorithm was first purposed in as 'Ant System' as a novel heurstic solution to approach combitorial problems (Dorigo and Di Caro, 1999). The source of inspiration for ant system algorithm is taken from real ants. The inital path covered by the ant in search of food is in random direction and deposits a chemical known as phermone. The journey returning from destination to source point of journey the ants follows the phermone trails (Socha and Dorigo, 2008).

Algorithm 1 Ant Colony Optimisation

```
while termination condition != False do
    SolutionConstruction()
    PheromoneUpdate()
    LocalSearch() [Optional]
end while
```

Construction Function

The Solution construction function requires to build a appropriate solution in graph $G = (V, E)$ where V represents a vertices and E represents the weighted edge between the vertices. The construction of path is based on a probabilistic method mentioned in equation below

$$p(c_{ij}) = \frac{\tau_{ij}^{\alpha}.\eta(c_{ij})^{\beta}}{\sum_{c_{ij}} \tau_{ij}^{\alpha}.\eta^{\beta}}, \forall c_{ij} \in \mathbb{N} \tag{1}$$

The $p(c_{ij})$ shows the probabilty of choosing the path i and j. τ_{ij} is the pheramone on the i and j vertices in the graphs. η is path visibilty value, which is assigned at each construction a heuristic value to optimal solution edges (Socha and Dorigo, 2008).α and β are the hyper parameters in the euation controlling the relationship between pheramone information and heuristic information (Socha and Dorigo, 2008).

PheromoneUpdate Function

The objective of the pheranome update function is to associate more phermone for path which has short edge distance between the vertices which can be achieved by increasing the $\Delta\tau_{i,j}^{k}$ pheromone level depsoited on i^th and j^th edge and decreasing the evaporation associate with edge with larger distances (Socha and Dorigo, 2008). The equation below shows the mathematical expression for calculating the pheromone deposited by k^thant in the graph.

$$\tau_{i,j}^{k} \begin{cases} \frac{1}{L^k}, & \text{L = length travelled by k ant on edge i and j} \\ 0, & \text{otherwise} \end{cases}$$

Pheromone without evaporation

The equation above shows the euation for pheromone by single ant. Furthermore, if there exists m ants then the following equation would be used to calculate the pheromone without evaporation.

$$\tau_{i,j} = \sum_{k=1}^{m} \Delta\tau_{i,j}^{k} \tag{2}$$

the equation will compute the sum of all pheromone deposited by all the ants, where m is total number of artifical ants in the system (Dorigo, Birattari and Stutzle, 2006).

Pheromone with evaporation

$$\tau_{i,j} = (1-\rho)\tau_{i,j} + \sum_{k=1}^{m} \Delta\tau_{i,j}^{k} \tag{3}$$

the equation above introduces new symbol ρ which is a constant that ranges between 0 and 1 and is called evaporation rate. $(1-\rho)\tau_{i,j}$ substracts evaporation constant from 1 and multiply it with the current phermone, which simulates the process of evaporation (Dorigo, Birattari and Stutzle, 2006).

Time Complexity Analysis

1. Initialise the solution. $O(n^2)$
2. Each ant construct the solution $O(n^2 m)$
3. Update the pheramone in the system $O(n^2)$
4. Output the result $O(1)$.

From the above steps it can be inffered that the time complextity for the ant colony optimisation algorithm after I_{max} iterations can be estimated to be $O(I_{max} n^2 m)$, where m are the number of the ants in the system (Li, 2015).

References

Dorigo, M., Birattari, M. and Stutzle, T. (2006), 'Ant colony optimization', *IEEE Computational Intelligence Magazine* **1**(4), 28–39.

Dorigo, M. and Di Caro, G. (1999), Ant colony optimization: a new meta-heuristic, *in* 'Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)', Vol. 2, pp. 1470–1477 Vol. 2.

Li, M. (2015), 'Efficiency improvement of ant colony optimization in solving the moderate ltsp', *Journal of Systems Engineering and Electronics* **26**(6), 1300–1308.

Socha, K. and Dorigo, M. (2008), 'Ant colony optimization for continuous domains', *European Journal of Operational Research* **185**(3), 1155 – 1173.
URL: *http://www.sciencedirect.com/science/article/pii/S0377221706006333*