Ant Colony Optimization method for Travelling Salesman Problem (TSP)

Team Members:

Dolly Gupta (181IT115) - 8081761629
Mansi Saxena (181IT126) - 9686638444
Meghna Kashyap (181IT127) - 9880921333
Priyanka B. G. (181IT135) - 9449936518

OUTLINE

- Summary
- Problem Statement
- Literature Survey
- Dataset Used
- Ant Colony Optimization (ACO)
- Solution Construction, Pheromone Updation
- Methodology
- Pseudo Code
- Time and Space Complexity
- Experimental Values, Results and Analysis, Optimal Parameters
- Comparison Of Our Results With Base Paper
- Future Work
- Conclusion

SUMMARY

The project aims to implement the Ant Colony Optimization method for TSP and compare its results with the base paper. The data set used is the same as the one used in the base paper "*The Effectiveness of Parameter Tuning on Ant Colony Optimization for Solving the Travelling Salesman Problem*" - Kush Shrivastava and Dr Shishir Kumar.

The link for the paper is added in 'Literature Survey' and has also been uploaded on our Github Link for easy reference.

PROBLEM STATEMENT - TSP

- Given a set of cities and distance between every pair of cities.
- The problem is to find the shortest possible route that visits every city exactly once and returns back to the starting point.
- The shortest path will be the minimum weight Hamiltonian Cycle.
- The problem is a famous NP hard problem.
- There is no polynomial time know solution for this problem.

LITERATURE SURVEY

1. *“Solving the Travelling Salesman Problem Using the Ant Colony Optimization” - Ivan Brezina Jr. Zuzana Čičková*
   This paper was used as reference to further understand the details and design of the ACO method.
2. *“The Effectiveness of Parameter Tuning on Ant Colony Optimization for Solving the Travelling Salesman Problem” - Kush Shrivastava and Dr Shishir Kumar*
   This base paper proved to be a key tool for the drafting and framework of this project because it helped lay a foundation to understanding the problem statement and implementing the ACO method.

DATASET USED

- We have used the directory which contains some examples of data for the traveling salesperson problem.
- Most of these examples come from TSPLIB, a collection of traveling salesman problem datasets maintained by Gerhard Reinelt at "http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/"
- **GR17: Set of 17 cities**, from TSPLIB.
  - gr17_d.txt, the inter-city distance table.
  - The minimal tour has length 2085.
  - The dataset is in the matrix format that it contains distance of all the cities with another cities.

ANT COLONY OPTIMIZATION (ACO)

- ACO is highly motivated by the foraging behavior of ants and the way they utilize pheromone trail to discover the food source.
- It is a soft computing metaheuristic that belongs to Swarm Intelligence(SI) methods.

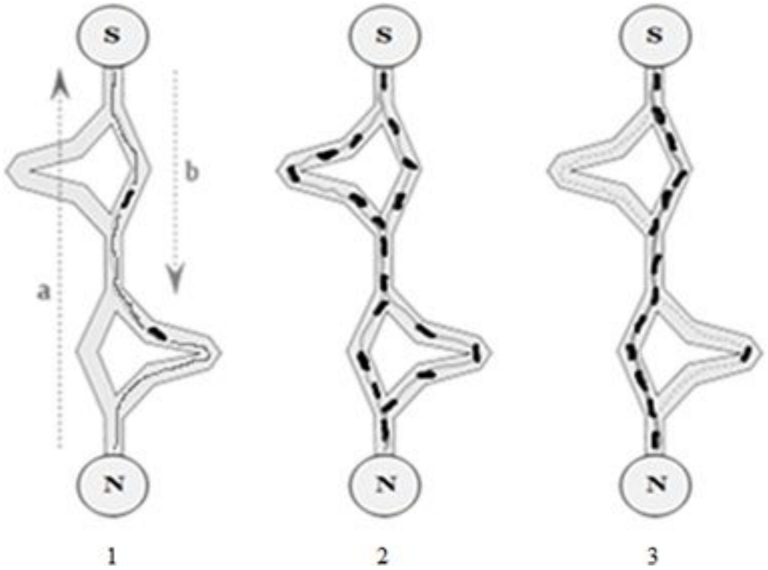


Fig. Ant Colony Optimisation Process

Source: https://plos.figshare.com/articles/_Ant_Colony_Optimization_Algorithm_processes_/1418788

Contd..

- ACO has proven a well performance in solving certain NP-hard problems in polynomial time.
- The bad selection of parameters increases the computation time and accuracy of the algorithm.

Contd..

The performance of an ACO algorithm is directly dependent on the choice of its parameter.

Parameters	Definition
α - Relative Importance of Pheromone	This parameter sets the amount of the pheromone on the edges would vanish after each cycle. This parameter really set how much ”memory” of past arrangements we need.
β - Relative Importance of Heuristics Value	This parameter sets the relative significance of pheromone versus heuristic value
ρ - Evaporation Rate	This parameter sets the amount of the pheromone on the edges would dissipate after each cycle. This parameter sets of the amount one ant impacted by different ants.

SOLUTION CONSTRUCTION

$$p_{ij}^{k} = \frac{[\tau_{ij}]^{\alpha} * [\eta_{ij}]^{\beta}}{\sum_{u \epsilon N_i^k} [\tau_{iu}]^{\alpha} * [\eta_{iu}]^{\beta}}$$

Where N^k_i is the set of cities not yet visited by ant k.
τ_{ij} the intensity of the pheromone trail between cities i and j
α is the parameter to regulate influence of τ_{ij}
η_{ij} the visibility of city j from city i = $1/d_{ij}$ (d_{ij} is the distance between city i and j)
β is the parameter to regulate the influence of η_{ij}

PHEROMONE UPDATION

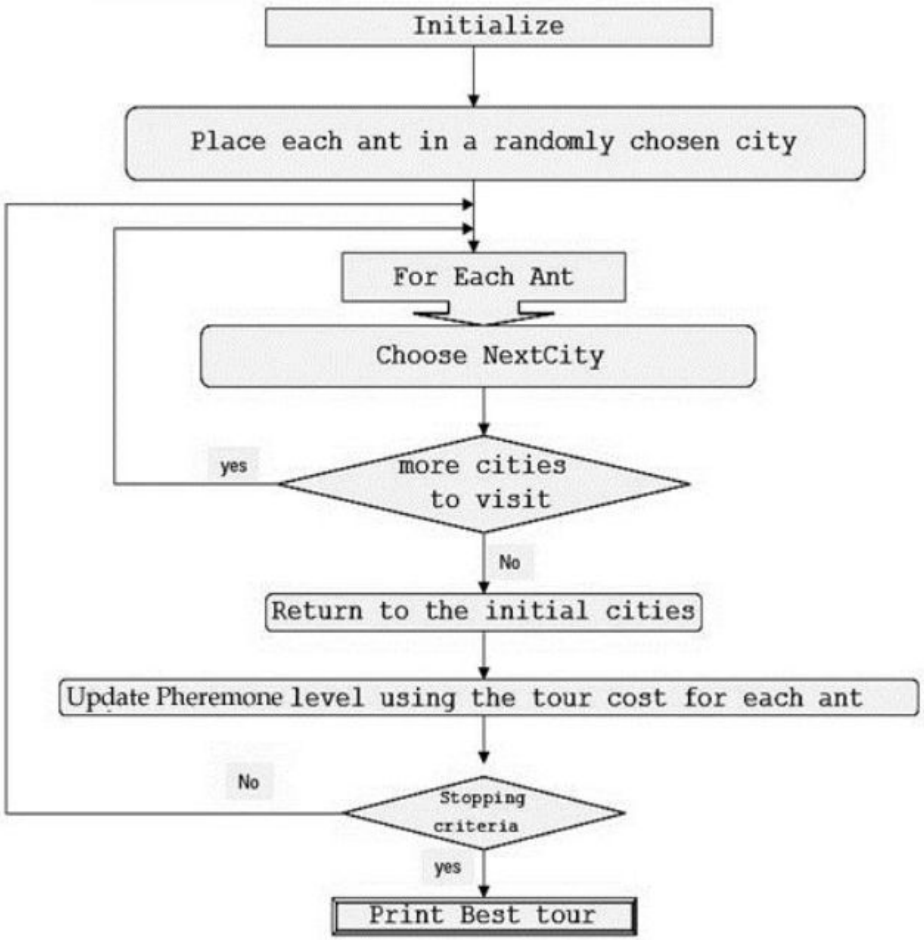
1. After each ant completes n iterations, the complete tour is obtained
2. The pheromones are updated in such a way that shorter path gets more pheromone than longer path
3. L_k is the length of the ants tour
4. Q is a constant
5. t is the iteration counter
6. ρ is the evaporation factor
7. Δτij is the total increase of trail level on edge (i,j) by all ants

$$\tau_{ij}(t+1) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij}$$

$$\Delta\tau_{ij} = \sum_{k=1}^{l} \Delta\tau_{ij}^{k}$$

$$\Delta\tau_{ij}^{k} = \begin{cases} Q/L_k & \text{if ant } k \text{ travels on edge } (i,j) \\ 0 & \text{otherwise} \end{cases}$$

METHODOLOGY



PSEUDO CODE

1. Initialize Trail
2. Do While(Stopping Criteria not Satisfied) - Cycle Loop

    For ant k=1 to Ith ant

        Do Until (Each Ant Completes the Tour) - Tour Loop

            Local Trail Update - ant moves based on P_{ij}^{k}

        End Loop

        Analyze Tours - Calculate L_k

    End Loop

    Global Trail Update - Update pheromone levels by formula $\Delta\tau_{ij}$

3. End Do

Hardness of the Problem

- The travelling salesman problem is NP Hard
- The decision version (not search) of TSP is NP-Complete
- Hence heuristics are used to get solutions in reasonable time

TIME COMPLEXITY

Step	Time Complexity
Initialize the parameters	$O(n^2+m)$
Set the tabulist	$O(m)$
Get separate solution for each ant	$O(n^2m)$
Update the pheromone	$O(n^2)$
Check whether the stopping criteria is met	$O(nm)$
Output the result	$O(1)$

Where n is number of cities and m is the number of ants.

Time Complexity

The computational complexity of ACO algorithm can be expressed as the function of the problem scale which is the (number of cities)^n.

When n is large enough, the impact of low power is negligible. Therefore after N iterations, the time complexity of ACO algorithm is $O(Nn^2m)$

SPACE COMPLEXITY

Through comprehensive analysis of each step the space complexity is $O(n^2) + O(nm)$. It can be seen that the space complexity of ACO algorithm is very simply and it is easy to program.

EXPERIMENTAL VALUES

For $\beta=1$, $\rho=0.1$ and varying value of α

No. of Iterations	**Values of α**					
	0.2	0.5	0.8	1	2	5
50	2279.3	2131.2	2122.2	2140.5	2130.0	2240.5
100	2187.7	2145.2	2094.5	2080.5	2219.5	2248.2
300	2178.0	2085.0	2096.0	2088.0	2278.0	2221.0
500	2160.3	2077.9	2084.0	2070.0	2256.8	2217.9
1000	2150.5	2065.0	2076.0	2067.8	2223.9	2209.3

EXPERIMENTAL VALUES

For α = 0.5, ρ=0.1 and varying values of β

No. of Iterations	**Values of β**					
	0.2	**0.5**	**1**	**3**	**5**	**10**
50	2331.3	2219.8	2161.2	2117.2	2132.6	2146.5
100	2287.3	2193.3	2120.0	2102.9	2130.5	2152.8
300	2221.6	2138.5	2104.8	2097.7	2130.5	2151.9
500	2210.8	2139.9	2102.9	2090.7	2128.2	2140.8
1000	2190.8	2120.9	2109.8	2089.0.	2110.8	2139.8

EXPERIMENTAL VALUES

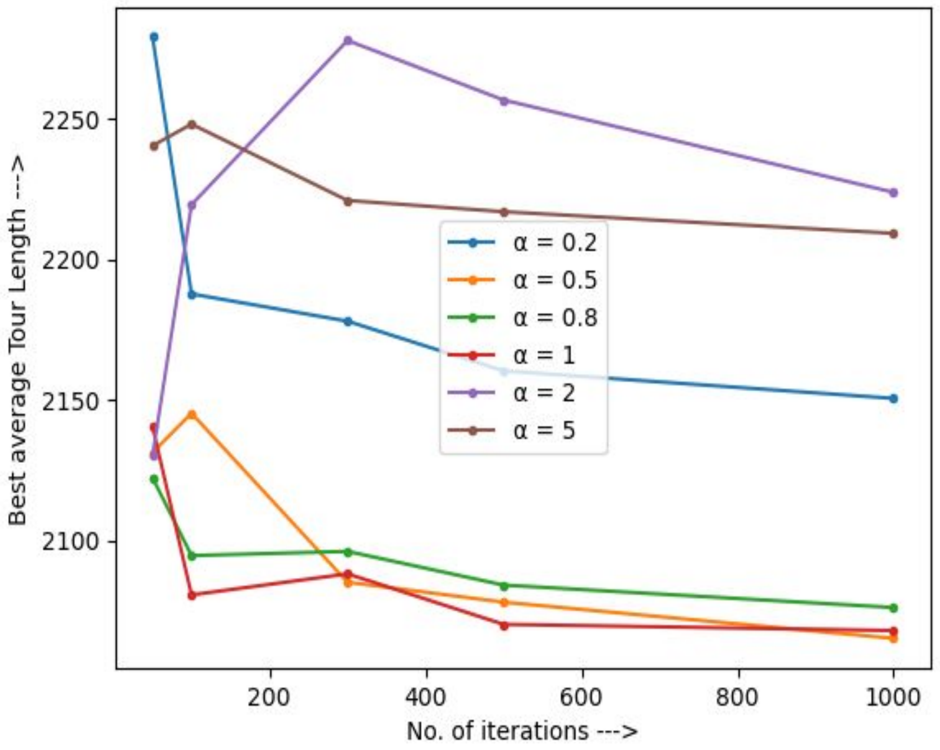
For $\alpha = 0.5$, $\beta=1$ and varying values of ρ

No. of Iterations	Values of ρ				
	0.1	0.3	0.5	0.7	0.9
50	2168.0	2120.0	2143.0	2160.0	2144.0
100	2088.0	2143.0	2161.0	2152.0	2174.0
300	2076.0	2057.0	2085.0	2168.0	2097.0
500	2060.9	2050.9	2080.9	2150.7	2080.6
1000	2040.4	2051.9	2078.4	2149.9	2078.7

RESULTS AND ANALYSIS

Impact of Pheromone factor

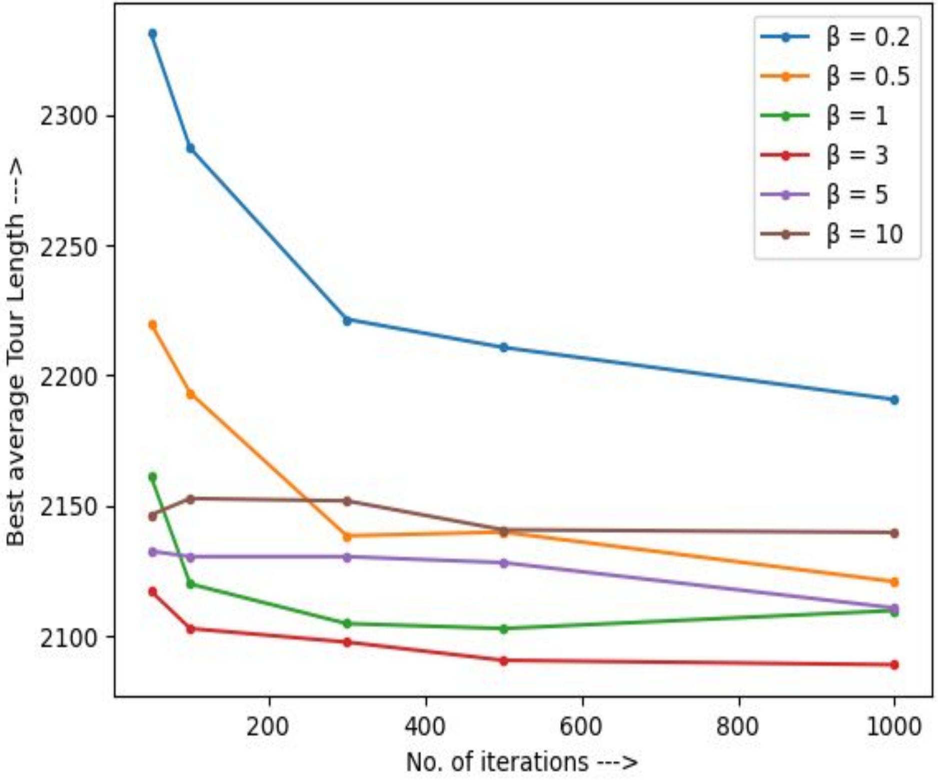
- Fo α = 0.2, we see that tour length is very high but decreases when iterations reaches 1000. It means enough importance is not given to alpha (pheromonic trail).
- For α = 0.5, we found better solution.
- For α = 0.8 and 1 we found very good solutions.
- For α is higher than 1 then it does not give very good solutions.
- Best value of α = 1



RESULTS AND ANALYSIS

Impact of Heuristic factor

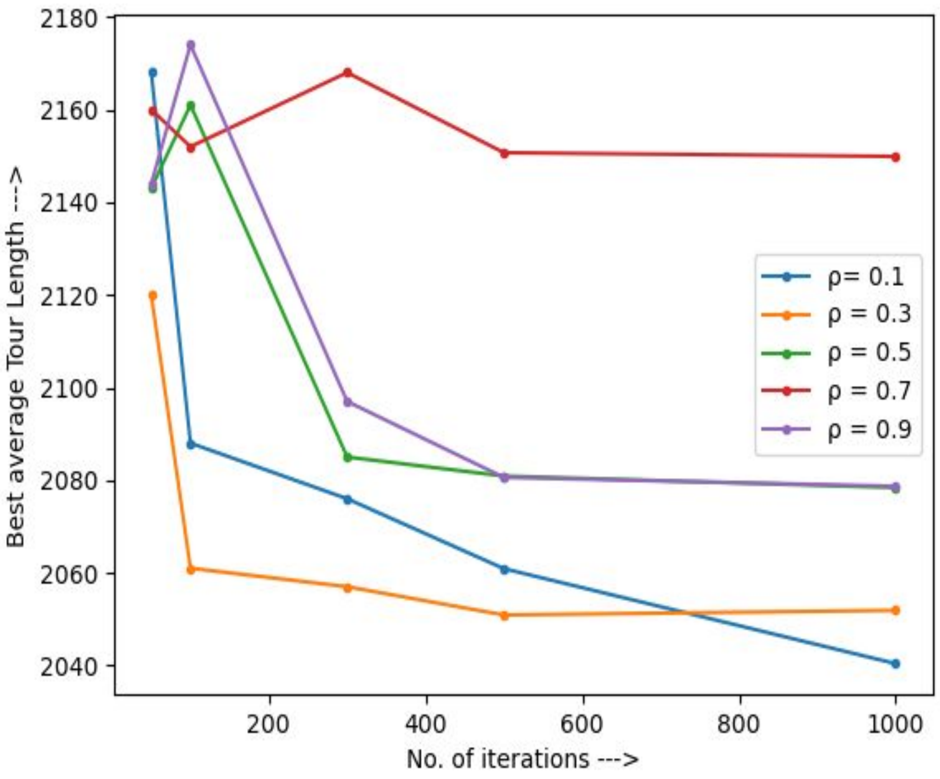
- If β = 0.2 and 0.5, the algorithm finds a poor solution this is because β is very small so it does not have enough emphasis on heuristic values.
- If β = 1 and 3 , when the iterations are less then tour length is not optimal but in higher values, we find the best tour length. It gives more emphasis to heuristic values.
- If β = 5 and 10 we, we don't see much variations in values, it indicates that the system has ceased to explore new possibilities.
- Best value of β = 3



RESULTS AND ANALYSIS

Impact of decaying factor

- When ρ = 0.1 and 0.3, we see that tour length is very high for lesser number of iterations but for higher values we get very good results because decaying factor is low and iterations are more.
- For ρ = 0.5, we don't see much variation on increasing the number of iterations
- For ρ = 0.7 and 0.9, we don't get good solution this is because when the evaporation rate will be higher, ants will forget the better path.
- Best value of ρ = 0.1 and 0.3



OPTIMAL PARAMETERS

Parameters	Optimal Value
α	1
β	3
ρ	0.3

COMPARISON OF OUR RESULTS WITH BASE PAPER

- We got the values of the tour length that are very close to the values given in the paper for all the set of parameters and number of iterations.
- Hence the graph also follows the trend similar to the paper.
- We have also got the values of parameters (For α, β and ρ) exactly same as the paper.

CONCLUSION

- The combinatorial optimization problems are NP-hard problems but can be solved using some metaheuristics.
- Ant Colony Optimization is a metaheuristic defined to solve the combinatorial optimization problems.
- In this work, we have presented an Ant Colony Optimization algorithm for solving travelling salesman problem.
- After that we have shown that the performance of Ant system algorithm depends on the appropriate setting of parameters which requires both human experience and luck to some extent.

FUTURE WORK

- Implement ACO for other dataset available in TSP dataset directory.
- Implement it for datasets other than TSP e.g. we can take dataset of a state or a country (containing coordinates of the cities).
- Apply ACO on 3D traveling salesman problem on a sphere.

THANK YOU