**ANT COLONY OPTIMIZATION**

**ASSUMPTIONS:** 1. Equal number of ants will be starting from all the cities

2. For a particular iteration once all the nodes have been traversed the ants move back to the initial node.

**METHODOLOGY**

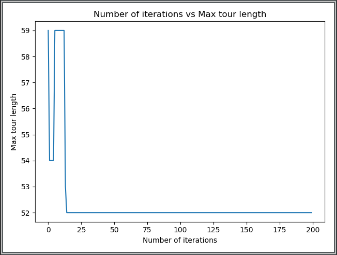
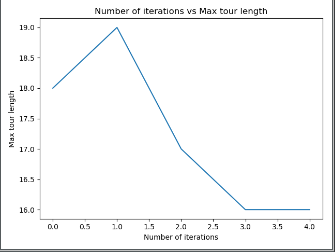
1. Initially the number of iterations for the ACO is taken as input
2. The distance matrix is for the TSP problem is generated as random providing edge weights as random values from 1 to 15.
3. The graph created using the adjacency matrix is always the complete graph to represent a tsp problem.
4. The number of nodes in the graph represents the number of cities which is taken as input from the user.
5. The decomposition rate is fixed as 0.5
6. The alpha and beta values as fixed as 0.7
7. The Q values is fixed as 1.
8. The initial pheromone content for all the edges are 1.

**ALGORITHM**

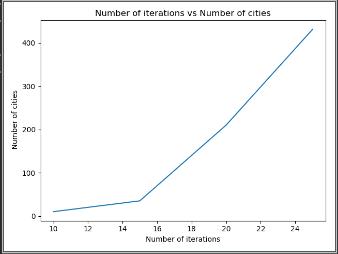
1. Create a random graph representing a TSP from input
2. Place n ants per nodes as n is taken as input
3. For each city as the starting position of ants perform the below steps
4. Select the next city based on the probability
5. Once all the nodes have been covered it traverse back to the initial node
6. Compute the tour length of all the ants and update it
7. Once all the ants have traversed back to its initial node one iteration is completed
8. After the completion of the iteraton the pheromone content of all the edges are updated
9. This continues for m iterations where m is taken as input from user
10. Once all the iterations have been covered the optimal path length can be found out.

**RESULTS AND ANALYSIS**

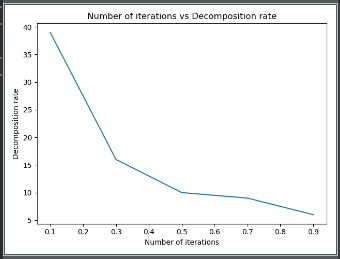
1. Graphical representation for number of iteration for convergence for city = 5 ants per city = 1 and city = 10 and ants per city = 1



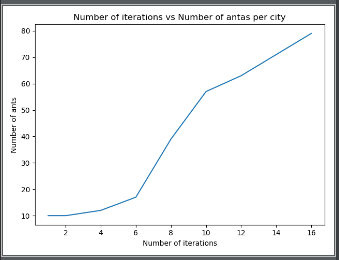
1. Graphical representation of number of cities against convergence rate where ants per city =1



3.Graphical representation on varying decomposition rate for max = 10 and ants per city = 1



1. Graphical representation on varying number of ants per city on city =10



**INFERENCE**

1. ACO solves the standard TSP problem in much faster way as compared to dynamic programming for smaller number of nodes.