Report PA-3

Kaustav Vats (2016048)

Assumptions

1. Pheromone is updated only after all Ants have reached another city.
2. Total Number of Ants is equal to Number of Cities present in the Graph.
3. Stopping condition is check for number of iterations reached or not.

Methodology

1. I created Ant class, which basically represent Ant as a object. Storing the cities visited, their current location, their previous location and some method to control state of the ant.
2. Population class to randomly generate graph and show using Networkx Library. And also to show final Result with edge weight as the pheromone present on the edge.
3. Aco class which contains various method to run Aco algorithm on random graph and show result using Networkx Library

Algorithm

1. Initialize all variable, graph etc.
2. Place each ant at random vertex in graph.
3. Repeat below steps till stopping condition.
4. For each ant find a city to visit next. Based on the probability received using Pheromone calculation formula.
5. Return to start vertex then update Pheromone level for each edge.
6. Go back to step 3.
7. Print Best tour.

Observations & Result

Evaporation Rate - On increasing the evap rate, I observed that convergence time is increased require high Repeating Iterations to get an Optimal Path. Whereas If we decrease Evap rate, algorithm converges faster but might not be an optimal path.

Alpha & Beta Values - On Increasing Alpha value I observed that edges with higher pheromone are given more preference, even edges with more length or close length are also considered.

On increasing Beta value, Distance is considered as the major factor, Shorter distance get more Pheromone.

N Value - On increasing N value Number of iterations required for convergence also increases and Takes lot of time to Converge. On high N value it's difficult to visualize the graph.