Report

Task 2 is about solving the travel salesman problem with ant colony optimization

The ant colony optimization strategy is about testing different passes using ants and their pheromones as a indicator on how good or bad the passes are

The algorithm for the ant colony optimization:

1. simulating a population of artificial ants, each representing a potential solution to the problem
2. As they explore the solution space, they deposit pheromone on the paths they traverse.
3. Ants are attracted to paths with higher pheromone concentrations. Ants also have a certain probability of choosing paths based on their length, favoring shorter paths
4. Over time, as more ants traverse a particular path, the pheromone concentration increases, making that path more attractive to subsequent ants.
5. the pheromone trails are updated based on the quality of the solutions found by the ants. The paths leading to better solutions receive a higher amount of pheromone, while paths that lead to poorer solutions gradually lose their pheromone due to evaporation.
6. Eventually, as the ants repeatedly explore and update the pheromone trails, a convergence is achieved

In practical use to the ant colony we did the following:

Distance\_matrix : a variable that holds the distance between each city and every other city

Tour\_distance : a function used to calculate the total distance the ant takes to complete its mission

def ant\_colony\_optimization(num\_cities, num\_ants, num\_iterations, alpha, beta, rho, q0):

This is the main function for the ant colony algorithm

First the parameters :

* Num\_cities : int for the number of cities the ant has to visit
* Num\_ants: int for how many ants we will use to test the passes
* Num\_iterations: int for how many updates we will make for the pheromones on the passes
* Alpha: parameter controls the importance of pheromone
* Beta: parameter controls the importance of distance
* Rho: parameter controls the pheromone evaporation rate
* Q0: parameter controls the exploration

We have 4 for loops itr->ant->two for loops to calculate the cities visited and unvisited

If condition to select the next city to visit

2 for loops to calculate the best tour and update the pheromone matrix

We give the best tour and the best distance as output of that function