Analysis and conclusions Ant Colony

Analysis

The probability formula allowed us to establish how likely it was that an ant would move to a city, this depended on the parameters Alpha and beta, an absolute value was used since when calculating the distance and multiplying it by beta the result could be a negative number and the objective was to find a probability and this had to be positive. The pheromone was subtracted from the distance since the probability of moving from one city to another decreased depending on how far one city was from another.

When testing with different values the graphs were different each time, one of the factors was the random factor of the program since cities were generated randomly so this is one of the possibilities why different distances are generated each time

Starting with initial values

number_cities = 50

number_ants = 100

number_iterations = 100

alpha = 1

beta = 1

evaporation_rate = 0.5

Q = 1

When generating 50 random cities the result of best_path_length was approximately 23.78

Varying the value of number_cities to 100 and taking the rest of the same values, the result of best_path_length will be 47.27, which is almost double the initial value, it can be said that it makes sense since the number of cities is doubling and the best distance is also multiplied by approximately 2

If the value of number_cities is taken to be 200 and the rest of the values are equal, the best distance will be 99.08, which is approximately double 47.27, so if the best distance is increased

By taking number_ants as 200 the best distance will be approximately 21.46 which will mean that the more ants the best path will be reduced, although it is reduced very little

By varying the number of iterations, it works to give more precision to the pheromones, since the ants will repeatedly pass through the same place.

By varying Alpha and Q that will affect the number of pheromones and leaving the rest of the values the same will influence which

Considerations

Actually, the numpy functions are quite simple and understandable and they really help a lot both to the efficiency of the code and to make the code quite simple, for example, when creating the function to generate cities, the random function of numpy makes it quite easy, although it was a little confusing to make the distance between cities and especially the probability formula since working with arrays lists and a matrix caused it to become a little complicated

The help of the probability formula was very significant since it allowed us to solve more quickly how the probability will be applied to the problem and program it itself.