**Mode Fair Home Assessment**

**Algorithm chosen**

Ant Colony Optimization (ACO)

The algorithm is inspired by the ant behavior in searching for food. At first, ant take random route to the food source. Pheromones are left by the ants along the trail as a mean of communication. As the pheromones evaporates over time, the pheromones concentration of the longer path decreases as ants take more time to travel along the longer path. On the contrary, the concentration of pheromones along the shorter path increases as ants travel shorter path more frequently. Eventually, all ants will follow the single shortest path.

In order to apply ACO, the problem needs to be modeled into finding shortest path on a weighted graph. Hence, it is suitable for this task as this task is a vehicle routing problem. This problem can be modeled into graph, with nodes representing the locations / customers while edges represent the path between the locations/ customers.

**Reference**

I have referred to the following paper to construct the solution. The formula for global update, local update as well as the probability of selecting next location are taken from the paper. I did not implement the ACS state transition rule in the paper but implemented the AS state transition rule instead.

M. Dorigo and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," in IEEE Transactions on Evolutionary Computation, vol. 1, no. 1, pp. 53-66, April 1997, doi: 10.1109/4235.585892.

**Overview**

Loop for iteration

Loop for ants

Construct a feasible solution, apply local update along the way

Compare and swap with global best solution

Apply global update

The high level pseudocode is shown above. Each ant will construct a feasible solution, that is: visit each location exactly once and total demand of each route must not exceed capacity of vehicle. In the process of constructing the solution, local pheromone update will be applied on the path chosen. The process of constructing the solution is as follows

while there are unvisited location(s)

check capacity, return to depot if not enough for remain locations

randomly select next location

apply local pheromone update

After an ant have completed the construction, the solution will be compared with the global best solution and replace it if it has a lower cost. After that, apply global pheromone update based on the global best solution.

**Result**

A screenshot of a computer code

Description automatically generated

*ACO result*

A computer code with numbers and symbols

Description automatically generated with medium confidence

*Greedy approach result*

The lowest cost obtained is RM121.10. I have compared with the cost obtained using a greedy approach (select the next location with lowest cost). The cost is RM128.25. This shows that ACO yield a better result.