Programming Task - Particle Swarm Optimization
April 30, 2021
Sisilia Rosari Widyastika

In the lecture (Programming Exercises), a PSO program is given for VRP problems with the following problem specifications:
• Number of destination points : 6
• Number of vehicles : 1
• no capacity limit.

Modify the given program for the following development of the problem :

• Added number of destination points → 15 destination points
• Each destination point has a number of requests for goods according to the demand data.
• There are 3 vehicles with a maximum capacity of each vehicle is 100 Kg.

Task Answer :
In solving problems using PSO it is necessary to:
1. Defining parameters
Parameters are the number of particles to be made, then the number of iterations to be carried out as well as the W, and C values. Because the problem is discussing CVRP, a parameter is needed that explains how many cars are used, their capacity, as well as a list of destinations to be traversed and the number of destinations to be traversed.
2. Definition of particle and velocity
In this problem, the particles to be defined are 3 (route particles, car particles, and demand particles) because they will find a route traversed by 3 vehicles that have demand capacity that can be transported.
3. fitness calculation
The fitness calculation is done first on each car. Find the value of the distance traveled by the first car then the second and third. The fitness value that will be used is the total value, the result of adding the entire total distance of the first to third cars.
4. Evaluate the Pbest and Gbest values
After getting the Pbest and Gbest values from each iteration, is it still necessary to update the Pbest?
5. Updating pBest
6. Recalculate the new velocity.
7. Update the particles.

A. increasing the number of destination points

The addition of the number of destination points is done when defining the particles of the route. First, create a list of destinations or destination points in the form of an array. Then to perform randomization of the route passed will use random choice. particle value is made an array based on the number of particles to be created, then to determine the value in each index in the array is done random choice from the list of destinations as many as the number of destinations to be passed. To avoid repeated values, replace = False is used.

The expected result is

Route	1	4	3	2

B. Increasing the number of vehicles and capacity limits

The parameters that will be needed are defined, namely the number of cars and the required capacity. After that, to define which route the car will take, demand data is needed which is the constraint for each car. To define the route the car will take as follows:

Route	1	2	4	3	5
Demand	30	70	100	100	20
Car	0	0	0	1	2

To define a car particle, an array is needed along the destination to be used, namely 15, then call the demand data which will be used as a reference for changes from the car used based on the route traversed. So that it takes two conditions, the condition that defines the number of cars (k) and the condition that defines the demand of the car (j) in defining the particles of the car. After that, the if-else condition is carried out by entering the constraint demand from the vehicle. So when the vehicle is full, the program will switch to another K. The reason for calculating the total distance is done by calculating the total distance per vehicle because each vehicle has a different start and end destination where each vehicle will have a different distance traveled.

D. overcoming in feasible particles

To overcome repetitive routes or destinations that are not reached by car. We can do a comparison of destinations by looking at the list of missing routes to fill in later. The missing route is a repeating route and so on.

Route	1	2	4	3	3
Feasible Solution for Route	1	2	4	3	5

Here are the results of PSO with several different setting parameters (W, C1, C2) :

Run	Inertia Weight = 0.7 C1 = 1 C2 = 2	Inertia Weight = 0.7 C1 = 1 C2 = 3	Inertia Weight = 0.7 C1 = 2 C2 = 3
1	Total Distance: 79 Car Route 1: 5 -3 -11-15-14- 6 -4 Car Route 2: 2-1-8-7-12 Car Route 3: 10 -9	Total Distance: 74 Car Route 1: 7-9-8-10-4-1 Car Route 2: 5-6-3-13-15-14-11 Car Route 3: 2-12	Total Distance: 79 Car Route 1: [12 7 4 15 14 3 1] Car Route 2: [5 6 2 8 9 13] Car Route 3: [10 11]
2	Total Distance: 77 Car Route 1: 15-13-11-1-5-3 Car Route 2: 2-7-12-8-9-4 Car Route 3: 10-14	Total Distance: 78 Car Route 1: [5 9 4 8 3 1] Car Route 2: [2 7 6 13 15 12] Car Route 3: [10 14 11]	Total Distance: 76 Car Route 1: [12 10 14 6 7 4 15] Car Route 2: [1 3 9 8 5 11] Car Route 3: 2 13]
3	Total Distance: 75 Car Route 1: 13-14-15-12-8-7-4 Car Route 2: 5-1-6-10-9-3 Car Route 3: 2-11	Total Distance: 76 Car Route 1: [5 3 1 9 10 4] Car Route 2: [12 8 15 7 11 13 6] Car Route 3: [2 14]	Total Distance: 78 Car Route 1: [7 12 14 11 13 3 1] Car Route 2: [15 4 6 8 9 10] Car Route 3: 2 5]
4	Total Distance: 77 Car Route 1: 15-14-11-1-8-5 Car Route 2: 13-3-9-10-4-6 Car Route 3: 7-12	Total Distance: 81 Car Route 1: [11 1 4 8 15 6 3] Car Route 2: [7 12 10 5 9 13] Car Route 3: [2 14]	Total Distance: 80 Car Route 1: [5 6 4 1 11 12] Car Route 2: [7 8 13 15 14 3 10] Car Route 3: [2 9]
5	Total Distance: 76 Car Route 1: 11-15-14-6-5-3-12 Car Route 2: 2-1-8-7-9-13 Car Route 3: 10-4	Total Distance: 78 Car Route 1: [9 12 10 5 14 15] Car Route 2: [7 4 8 1 13 3 11] Car Route 3: [2 6]	Total Distance: 82 Car Route 1: [9 11 6 7 12 3 13] Car Route 2: [2 15 5 1 8 4] Car Route 3: [10 14]
6	Total Distance: 76 Car Route 1: 7-4-8-1-6-3 Car Route 2: 10-9-15-14-11-13-5 Car Route 3: 2-12	Total Distance: 77 Car Route 1: [10 11 1 15 4 8 13] Car Route 2: [5 9 6 14 3 2] Car Route 3: [7 12]	Total Distance: 79 Car Route 1: [12 8 10 15 4 1] Car Route 2: [5 9 6 13 3 2] Car Route 3: [7 11 14]
7	Total Distance: 78 Car Route 1: 7-10-13-5-1-4 Car Route 2: 2-3-6-8-9-11 Car Route 3 :14-15-12	Total Distance: 78 Car Route 1: [10 13 5 3 1 4 6] Car Route 2: [2 7 9 11 15 12] Car Route 3: [8 14]	Total Distance: 74 Car Route 1: [7 9 3 2 1 4] Car Route 2: [10 5 6 8 13 11] Car Route 3: [15 14 12]
8	Total Distance: 77 Car Route 1:10-5-1-4-6-11 Car Route 2:	Total Distance: 81 Car Route 1: [13 15 7 8 6 3 4] Car Route 2: [5 2 1	Total Distance: 72 Car Route 1: [7 9 10 4 8 1] Car Route 2: [5 3 6

	15-12-7-8-9-3-13 Car Route 3: 2-14	9 11] Car Route 3: [10 14 12]	15 14 11 12] Car Route 3: [13 2]
9	Total Distance: 77 Car Route 1: 13-9-5-2-3-1 Car Route 2: 15-4-6-8-7-11 Car Route 3: 10-14-12	Total Distance: 81 Car Route 1: [7 9 3 6 2 13 11] Car Route 2: [5 1 4 8 14 12] Car Route 3: [15 10]	Total Distance: 79 Car Route 1: [9 10 8 7 4 1] Car Route 2: [5 3 6 15 14 11 12] Car Route 3: [13 2]
10	Total Distance: 81 Car Route 1: 7-9-13-1-2 Car Route 2: 4—6-10-5-11 Car Route 3: 15-14-12	Total Distance: 77 Car Route 1: [12 8 9 6 14 1] Car Route 2: [15 4 11 7 10 5 3] Car Route 3: [13 2]	Total Distance: 78 Car Route 1: [12 7 11 15 5 2] Car Route 2: [1 13 3 8 9 6 4] Car Route 3: [10 14]
Average total Distance	77.3	78.1	77.7
Standart Deviation	1.702938637	2.330951165	2.945806813
Best Value	75	74	72

Based on the results of several parameters, the best parameter is the parameter setting with a value of W = 0.7, C1 = 1, C2 = 2. This is because the standard deviation value is small, meaning that the data distribution is not too large or the range of total distance results is several times. running is not great. These parameters are quite focused and stable. Although the smallest total distance value is found in the third setting parameter, this parameter has the highest standard deviation. The time needed to run the program is 1.0140321254730225 seconds.