Applied Computational Intelligence

1st Semester 2022/2023

Group: 28

Marco da Cruz da Mata nº 84297

marcocmata@tecnico.ulisboapt

João Pedro Andrade Gonçalves nº 85211

jpedrogoncalves@tecnico.ulisboa.pt

Table of Contents

Part 1: Single-Objective Optimization Problem 2
Part 2: Multiple-Objective Optimization Problem 4
Conclusions 5

Part 1: Single-Objective Optimization Problem

Our evaluation function, appropriately called EVALUATE, takes an individual, which is defined as an array of which customers to deliver to, in order, not including warehouse visits, and computes the distance traveled during the entire course. Even though we don't represent warehouse visits in the genome, the EVALUATE function makes sure the load never surpasses 1000 by sending our truck back to the warehouse and to the next point of delivery, by adding the distances to and from the warehouse to the total distance.

An example genome, for the 10 customers problem, is as follows:

[3, 4, 8, 1, 5, 7, 2, 10, 6, 9]

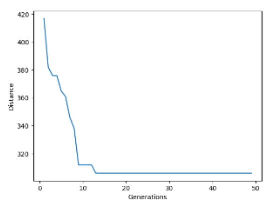
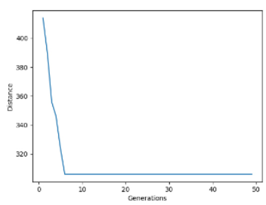
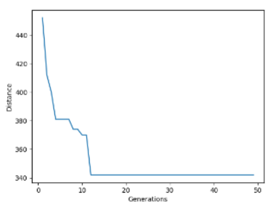
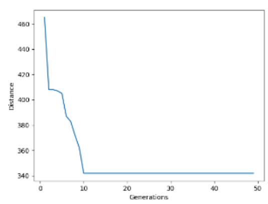
Results

#Costumers	WHCentral_OrdFile		WHCentral_Ord50		WHCorner_OrdFile		WHCorner_Ord50	
	Mean	**STD**	**Mean**	**STD**	**Mean**	**STD**	**Mean**	**STD**
10	308.6	2.615	307	2	384.4	8.16	347.8	6.615
30	754.1	67.55	731.5	56.363	951.1	68.66	869.3	48.33
50	1384.9	108.0			1756.8	132.2	1676.1	122.2

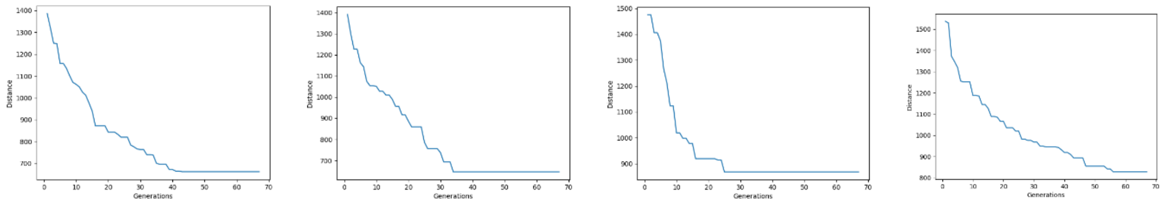
Table 1 – Results based on execution of 30 runs for single-objective

Graphs (Convergence Curves)

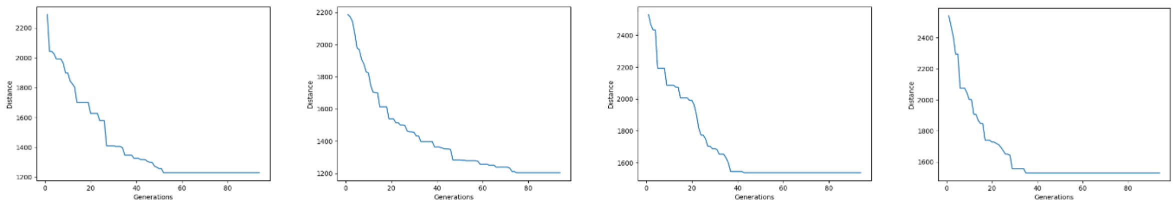
10 Costumers: Central, Central 50 orders, Corner, Corner 50 orders

30 Costumers: Central, Central 50 orders, Corner, Corner 50 orders

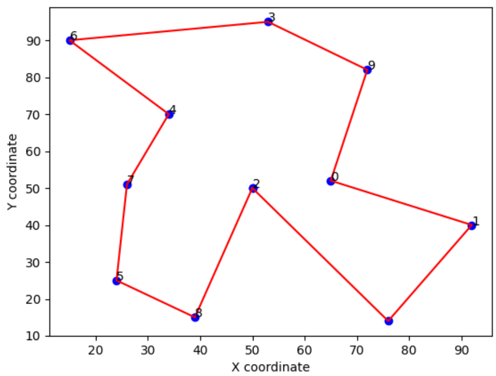


50 Costumers: Central, Central 50 orders, Corner, Corner 50 orders



Using Heuristics

The Heuristics solution will start in the warehouse and will divide the population by splitting the horizontal axis. Then it will move to the population on the left side from the lowest to the highest vertical position. After this it will move to the right side moving from the highest vertical position to the lowest and then returning to the warehouse. An example can be observed in the following image.



For the heuristics variation, we will use this solution as the first solution that enters the EA.

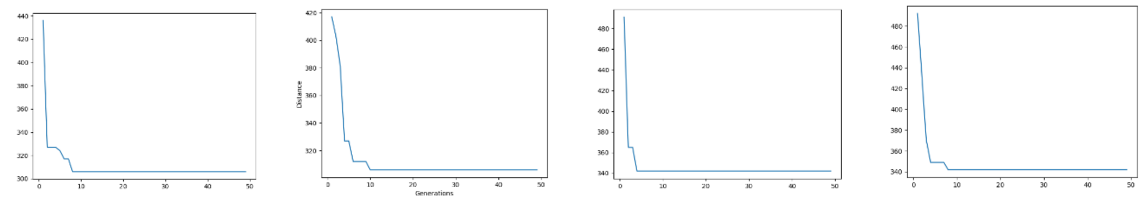
Results

#Costumers	WHCentral_OrdFile		WHCentral_Ord50		WHCorner_OrdFile		WHCorner_Ord50	
	Mean	**STD**	**Mean**	**STD**	**Mean**	**STD**	**Mean**	**STD**
10	310.567	3.922	310.67	3.815	342.233	1.257	342.7	2.1
30	593.867	24.55	589.2	17.53	695.2	30.56	686.3	31.077
50	935.1	76.93	921.1	27.47	1199.8	48.54	1147.1	34.4

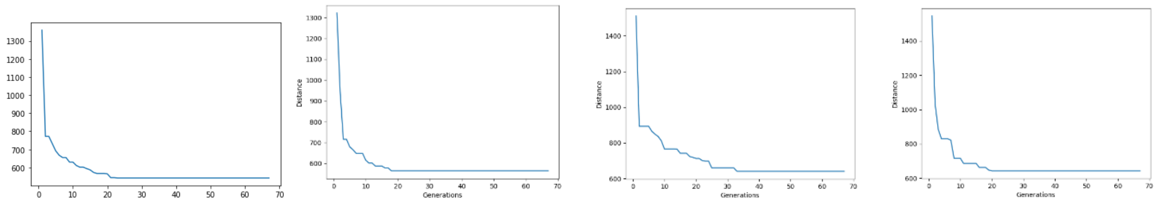
Table 2 - Results based on execution of 30 runs for single-objective using heuristics

Graphs (Convergence Curves)

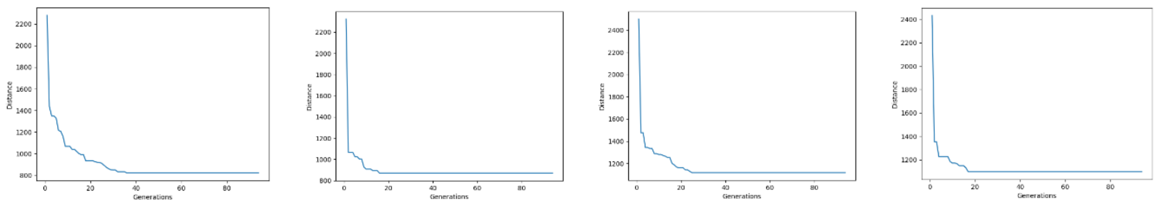
10 Costumers: Central, Central 50 orders, Corner, Corner 50 orders



30 Costumers: Central, Central 50 orders, Corner, Corner 50 orders



50 Costumers: Central, Central 50 orders, Corner, Corner 50 orders



Part 2: Multiple-Objective Optimization Problem

The evaluation function for the multi-objective problem works similarly to the single-objective problem, except this time we store the current trip before the truck needs to go back to the warehouse and we compute the cost by adding all distances traveled between costumers and multiplying it by the load the truck had when traveling between those costumers. For the multi-objective section, the function returns a tuple of the total distance traveled and the total cost of the entire trip. The genome used is the same as in the single objective optimization.

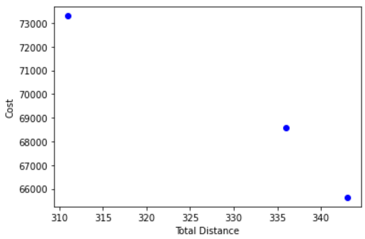
Results

#Costumers	Min Cost		Min Dist	
	Dist	Cost	Dist	Cost
10	343	65640	311	73300
30	687	255170	682	255330
50	1560	544960	1371	598780

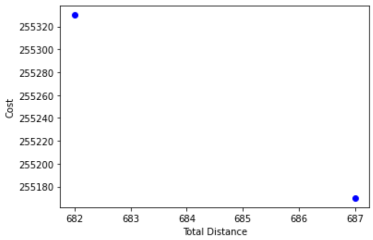
Table 3 - Results for multiple-objective

Graphs (Pareto Front)

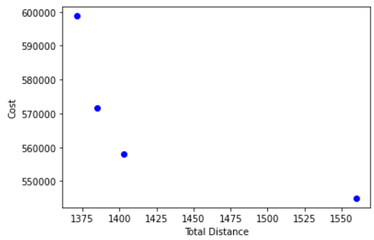
10 Costumers:



30 Costumers:



50 Costumers:



Conclusions

For the single-objective problem, specifically when using heuristics, we notice that the algorithm converges much faster, however we feel that it would've been possible to make it so that it didn't reach the "minimum" as fast as it does for most situations. The performance using the heuristics solution outperforms the ones where we don't use it, which makes total sense since we are giving as the initial solution a solution that is already optimized instead of a random one. However, due to variance, for 10 costumers, with the warehouse in a central position and using the orders from the file, the best solution using heuristics underperforms when compared to the version when not using the heuristics.

For the multi-objective problem, we were able to showcase the Pareto Front, despite the varying number of points in the graph, since they were always at least 2. Unfortunately, however, due to poor planning on our part, we were unable to compute the hypervolume in due time.