Computational Intelligence for Optimization Project 2021

Capacitated Vehicle Routing Problem:
Trash Collection in Lisbon Area Neighborhoods

Lisbon
Almada
Pontinha
Avenida de Brasília
Via Verde
A5
A2
IC17
A37
A36
EN 117
EN 377

Done by:
Allan Nhapulo (m20201091)
Lyinder Swale (m20201009)
Zhenghao Li (m20201007)

Abstract: Computational intelligence for optimization techniques has many applications in our day to day lives. One of the applications is vehicle routing which helps to manage available resources efficiently and effectively. In this study, the techniques were used to find the best shortest routes trucks can travel to collect trash in Lisbon area within their weight capacity.

Keywords: *Capacitated vehicle routing problem, Genetic algorithm, Multi-objective Optimization*

1. Introduction

Computational intelligence for optimization can be applied in many different aspects of our day-to-day lives for the purpose of solving complex problems within a shorter period. In result, benefits such as cost reduction, time saving as well as resources management can be attained.

In this project, we assume weekly trash collection for 50 neighborhoods of Lisbon metropolitan area whereas the district owns 5 trucks to go around all the neighborhoods starting from the dump area and returning to the same. We also assume that each truck has the capacity of 40 tons.

Capacitated vehicle routing problem

The capacitated vehicle routing problem (CVRP) is a vehicle routing problem in which vehicles with limited carrying capacity need to pick up or deliver items at various locations. The problem is to pick up or deliver the items for the least cost, while never exceeding the capacity of the vehicles.

Based on the definition, our problem classifies as a CVRP problem. We assume that least cost in our problem equates to shortest (total) distance travelled by all trucks with all trash collected.

Data description

Data used in this project is 51 geographic points in Lisbon metropolitan area, where 50 of them are the pickup points namely; 'Alfama', 'Bairro Alto', 'Chiado', 'Campo de Ourique', 'Areeiro', 'Telheiras', 'Restelo', 'Chelas', 'Príncipe Real', 'Alta de Lisboa', 'Moorish Quartear', 'Bairro Padre Cruz', 'Xabregas', 'Bairro de casas económica', 'Bairro de Boavista', 'Bairro da Cruz Vermelha', 'Cais do Sodré', 'Quinta do Lambert', 'Galinheiras', 'Musgueira', 'Olivais Sul', 'Bairro de Horta Nova', 'Bairro do Rego', 'Madre Deus', 'Bairro Azul', 'Bairro Grandella', 'Bairro de Liberdade', 'Alvito', 'Bairro de Angola', 'Olivais Norte', 'Calhariz de Benfica', 'Alto da Ajuda', 'Bairro das Amendoeiras', 'Condado', 'Quinta das Lavadeiras', 'Bairro da Flamenga', 'Bairro das Calvanas', 'Palma de Baixo', 'Jardim da Burra', 'Olaias', 'Quinta de Santa Clara', 'Bairro São João', 'Quinta do Pisany', 'Campolide', 'Teresinhas', 'Santa Maria dos Olivais', 'Casal das Cruzes', 'Bairro Dona Leonor', 'Ameixoeira', 'Quinta do Olival' and the 'Dump_area' where the trucks originate and return to dump the trash and park. Each pickup point contains between 1 and 6 tons of trash.

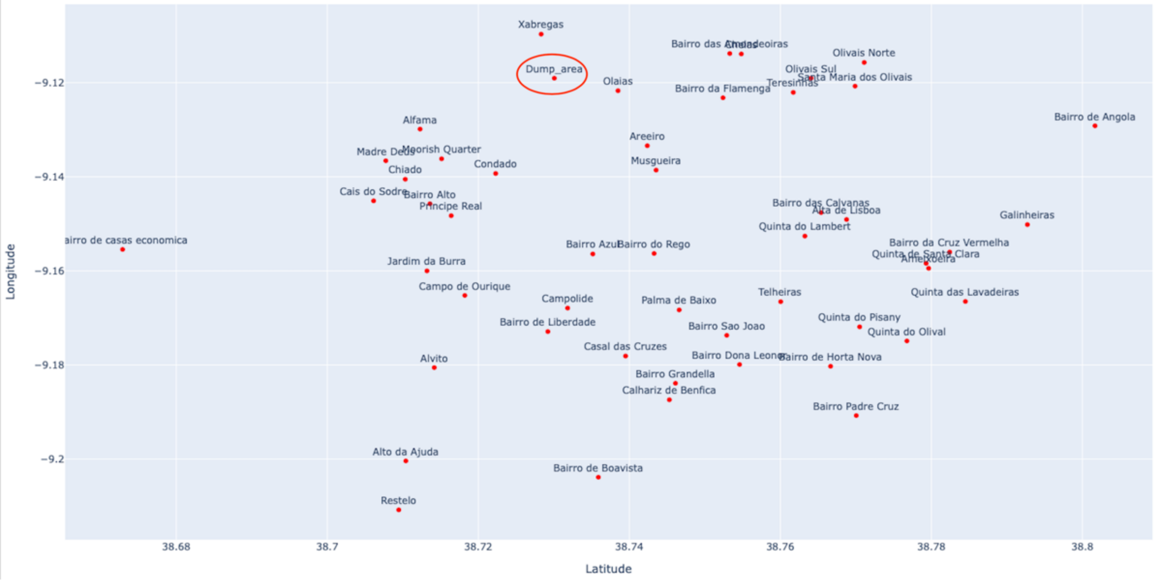


Figure 1: Pickup points

Problem formalization

The objective of an optimization problem is to find the best solution(s) to a problem in a (huge) set of possible alternative solutions. A set of possible solutions and the fitness function ***(P, f)*** that measures the best solution must be defined.

In our problem, the individual solution (*N*) is defined as a set of 5 trucks completing trash collection per week whereas each truck collects from 10 pickup points. A set of solutions (*P)* is different kinds of routes each set (of 5) takes, while the fitness function measures the distance each truck travelled from the dump area and back and sums the total distance travelled by all the 5 trucks. It also takes into account that each the trucks must not exceed its capacity of 40 tons.

Individual (*N*) = [[truck 1 route], [truck 2 route], [truck 3 route], [truck 4 route], [truck 5 route]]

Population (*P*) = [$N_1, N_2, N_3, ..., N_i$]

Fitness function (*f*) = Minimum distance travelled by N where weight collected (Nw) equals total weight

2. Methodology

The methodology used in our CVRP problem, is genetic algorithm by following the process as shown on figure 2.

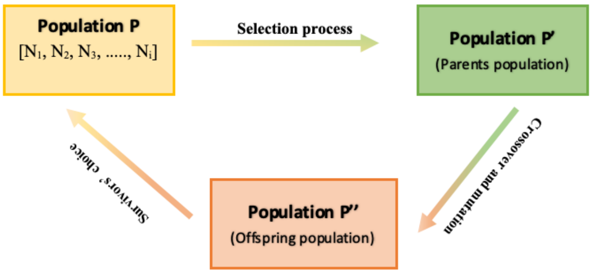
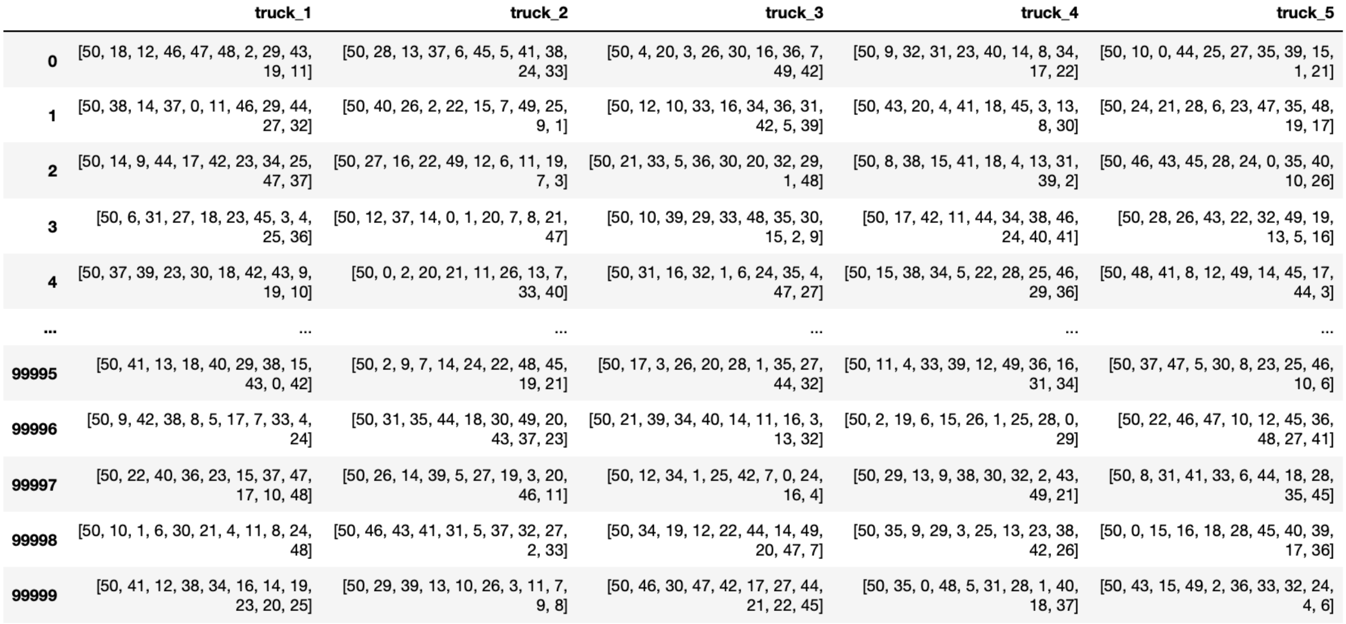


Figure 2: Genetic algorithm process

Population *P*

Population P was created by shuffling the list of 50 pickup points 1000 times and for each shuffle a set of 50 points containing 5 sets of 10 points was created. The process created the total of 1000 individuals. As seen on figure 3, each truck starts at point 50 which is the DUMP_AREA. Each row represents an individual **N,** and the columns are each truck trip per individual.



	truck_1	truck_2	truck_3	truck_4	truck_5
0	[50, 18, 12, 46, 47, 48, 2, 29, 43, 19, 11]	[50, 28, 13, 37, 6, 45, 5, 41, 38, 24, 33]	[50, 4, 20, 3, 26, 30, 16, 36, 7, 49, 42]	[50, 9, 32, 31, 23, 40, 14, 8, 34, 17, 22]	[50, 10, 0, 44, 25, 27, 35, 39, 15, 1, 21]
1	[50, 38, 14, 37, 0, 11, 46, 29, 44, 27, 32]	[50, 40, 26, 2, 22, 15, 7, 49, 25, 9, 1]	[50, 12, 10, 33, 16, 34, 36, 31, 42, 5, 39]	[50, 43, 20, 4, 41, 18, 45, 3, 13, 8, 30]	[50, 24, 21, 28, 6, 23, 47, 35, 48, 19, 17]
2	[50, 14, 9, 44, 17, 42, 23, 34, 25, 47, 37]	[50, 27, 16, 22, 49, 12, 6, 11, 19, 7, 3]	[50, 21, 33, 5, 36, 30, 20, 32, 29, 1, 48]	[50, 8, 38, 15, 41, 18, 4, 13, 31, 39, 2]	[50, 46, 43, 45, 28, 24, 0, 35, 40, 10, 26]
3	[50, 6, 31, 27, 18, 23, 45, 3, 4, 25, 36]	[50, 12, 37, 14, 0, 1, 20, 7, 8, 21, 47]	[50, 10, 39, 29, 33, 48, 35, 30, 15, 2, 9]	[50, 17, 42, 11, 44, 34, 38, 46, 24, 40, 41]	[50, 28, 26, 43, 22, 32, 49, 19, 13, 5, 16]
4	[50, 37, 39, 23, 30, 18, 42, 43, 9, 19, 10]	[50, 0, 2, 20, 21, 11, 26, 13, 7, 33, 40]	[50, 31, 16, 32, 1, 6, 24, 35, 4, 47, 27]	[50, 15, 38, 34, 5, 22, 28, 25, 46, 29, 36]	[50, 48, 41, 8, 12, 49, 14, 45, 17, 44, 3]
...	...	...	...	...	...
99995	[50, 41, 13, 18, 40, 29, 38, 15, 43, 0, 42]	[50, 2, 9, 7, 14, 24, 22, 48, 45, 19, 21]	[50, 17, 3, 26, 20, 28, 1, 35, 27, 44, 32]	[50, 11, 4, 33, 39, 12, 49, 36, 16, 31, 34]	[50, 37, 47, 5, 30, 8, 23, 25, 46, 10, 6]
99996	[50, 9, 42, 38, 8, 5, 17, 7, 33, 4, 24]	[50, 31, 35, 44, 18, 30, 49, 20, 43, 37, 23]	[50, 21, 39, 34, 40, 14, 11, 16, 3, 13, 32]	[50, 2, 19, 6, 15, 26, 1, 25, 28, 0, 29]	[50, 22, 46, 47, 10, 12, 45, 36, 48, 27, 41]
99997	[50, 22, 40, 36, 23, 15, 37, 47, 17, 10, 48]	[50, 26, 14, 39, 5, 27, 19, 3, 20, 46, 11]	[50, 12, 34, 1, 25, 42, 7, 0, 24, 16, 4]	[50, 29, 13, 9, 38, 30, 32, 2, 43, 49, 21]	[50, 8, 31, 41, 33, 6, 44, 18, 28, 35, 45]
99998	[50, 10, 1, 6, 30, 21, 4, 11, 8, 24, 48]	[50, 46, 43, 41, 31, 5, 37, 32, 27, 2, 33]	[50, 34, 19, 12, 22, 44, 14, 49, 20, 47, 7]	[50, 35, 9, 29, 3, 25, 13, 23, 38, 42, 26]	[50, 0, 15, 16, 18, 28, 45, 40, 39, 17, 36]
99999	[50, 41, 12, 38, 34, 16, 14, 19, 23, 20, 25]	[50, 29, 39, 13, 10, 26, 3, 11, 7, 9, 8]	[50, 46, 30, 47, 42, 17, 27, 44, 21, 22, 45]	[50, 35, 0, 48, 5, 31, 28, 1, 40, 18, 37]	[50, 43, 15, 49, 2, 36, 33, 32, 24, 4, 6]

Figure 3: Population P

Fitness *f*

A distance matrix was created by calculating distances from each pickup point to the other. The matrix was then used to calculate the distance travelled by each individual taking into account the trucks capacity.

For each truck, distance and weight carried are calculated and if a truck exceeds its capacity, then it equals 0. When the total weight collected by all trucks (per individual) is calculated, if there was any truck that exceeded capacity then the total individual weight will be less than total weight to be collected and hence the solution will be dimmed as a 'n*on-admissible solution*'.

The solutions with less weight than the weight to be collected is then filtered out before the selection process.

Selection process of parents'

Individuals were selected by two different methods, tournament selection method whereas a sample of 20 individuals is randomly selected from the population and by roulette wheel selection where the probability of of an individual being picked is by dividing the total population fitness and individual fitness. Two individuals are picked (independently) by selecting an individual with the best fitness which are then used as parents for the variation processes (crossover and mutation). Note that, each trip contains the DUMP_AREA point 50.

NOTE: Before crossover and mutation, the point (50) is removed from the trips and each individual is flattened and so, instead of crossing over truck by truck, we crossover individual by individual and then dividing the trips back to 5.

The elite individuals are also selected whereas the individual with the overall best fitness from the entire population (not sample) is added in the new population without performing any variation procedure.

Crossover and mutation

The variation processes are performed probabistically with different rates. The rates control the variation processes such that the individuals have more diversity. The selected pairs of parents are then crossed over to produce offsprings and then mutated probabilistically whereas some individuals are only crossed over and not mutated and vice versa while some go through both processed and some none of the processes. The offsprings then replace the new population and the evolution process starts again for 100 generations.

3. Results and Discussion

The 50 pickup points were randomly shuffled to create different solutions/individuals. Each individual was then divided by the number of trucks (in our case 5) of the same capacity (40 tons). Each truck is meant to collect from 10 points and starting and ending at the same point (point 11) which in our case was labelled as 50. Fitness was calculated on each individual whereas the individuals with the best fitness were passed in the new population (in case of elitism) and the rest of the individuals were crossed between eachother pairwise to create new individuals. The individuals were then mutated and appended to the new population. The fitness for each individual was calculated again and the best individual was returned for each generation process.

The evolution process yielded different results based on different population sizes, variation methods and rates and with elitism or without.

From the results, the individual that cycle crossover and inversion mutation variation methods were applied got the best fitness. The table below shows the best routes and their fitness scores.

Variation	Routes	Fitness score
Tournament selection Cycle crossover (0.7) Swap mutation (0.3) Elitism = True	[50, 14, 0, 19, 29, 28, 5, 43, 10, 35, 25] [50, 42, 24, 21, 9, 34, 45, 18, 49, 38, 1] [50, 26, 3, 30, 41, 46, 36, 8, 2, 33, 13] [50, 12, 39, 11, 27, 31, 6, 16, 23, 20, 37] [50, 4, 22, 15, 17, 48, 32, 40, 47, 7, 44]	207.69
Roulette wheel selection Cycle crossover (0.7) Swap mutation (0.3) Elitism = False	[50, 32, 36, 7, 19, 47, 43, 38, 40, 9, 15] [50, 4, 30, 37, 24, 28, 11, 5, 26, 23, 39] [50, 20, 41, 31, 46, 25, 27, 34, 48, 17, 21] [50, 44, 29, 1, 6, 3, 10, 12, 18, 49, 22] [50, 33, 8, 2, 42, 45, 35, 13, 0, 16, 14]	209.33
Tournament selection **Cycle crossover (0.7)** **Inversion mutation (0.3)** **Elitism = True**	**[50, 1, 30, 17, 34, 4, 36, 45, 0, 23, 44]** **[50, 5, 42, 25, 37, 48, 28, 20, 40, 18, 8]** **[50, 33, 2, 16, 38, 47, 41, 19, 3, 14, 10]** **[50, 27, 26, 39, 32, 35, 43, 24, 6, 31, 13]** **[50, 7, 9, 11, 15, 49, 46, 21, 22, 29, 12]**	**186.52**
Roulette wheel selection Cycle crossover (0.7) Inversion mutation (0.3) Elitism = False	[50, 49, 21, 22, 43, 24, 39, 35, 5, 30, 19] [50, 12, 31, 46, 18, 28, 32, 15, 2, 8, 20] [50, 14, 17, 7, 4, 42, 48, 47, 10, 0, 38] [50, 37, 41, 26, 3, 9, 11, 40, 45, 44, 34] [50, 13, 1, 6, 27, 16, 23, 29, 36, 25, 33]	205.76
Tournament selection Single point crossover (0.9) Swap mutation (0.3) Elitism = True	[50, 19, 22, 39, 21, 14, 41, 26, 8, 10, 45] [50, 44, 32, 36, 9, 1, 38, 16, 2, 48, 33] [50, 0, 23, 20, 18, 46, 5, 17, 40, 28, 36] [50, 29, 12, 1, 0, 3, 49, 46, 48, 18, 40] [50, 19, 37, 9, 35, 33, 5, 34, 14, 31, 24]	191.1
Roulette wheel selection Single point crossover (0.9) Swap mutation (0.3) Elitism = False	[50, 21, 22, 39, 19, 14, 41, 26, 8, 44, 45] [50, 44, 32, 36, 9, 1, 38, 16, 2, 48, 33] [50, 0, 23, 20, 18, 46, 5, 17, 40, 11, 25] [50, 15, 35, 10, 38, 48, 46, 37, 36, 34, 43] [50, 31, 33, 10, 1, 23, 39, 15, 40, 25, 41]	193.65
Tournament selection Single point crossover (0.9) Inversion mutation (0.3) Elitism = True	[50, 18, 36, 48, 0, 37, 22, 44, 21, 41, 12] [50, 16, 47, 19, 4, 2, 43, 3, 30, 24, 33] [50, 1, 13, 17, 5, 42, 11, 22, 24, 39, 12] [50, 17, 49, 28, 48, 29, 34, 18, 35, 16, 0] [50, 10, 31, 36, 46, 33, 15, 44, 19, 41, 23]	197.57
Roulette wheel selection Single point crossover (0.9) Inversion mutation (0.3) Elitism = False	[50, 21, 22, 39, 19, 14, 41, 26, 8, 10, 45] [50, 44, 32, 36, 9, 1, 38, 16, 2, 48, 33] [50, 0, 23, 20, 18, 46, 5, 17, 40, 11, 25] [50, 15, 42, 29, 7, 24, 27, 31, 35, 34, 49] [50, 39, 25, 2, 10, 9, 37, 22, 28, 18, 48]	197.96

Diagram 4 is the illustration of the best trip where each truck is represented by a different color. Plotting the results help realize how complex it is to guess the best trip. The trucks take complex routes and totally unpredictable.

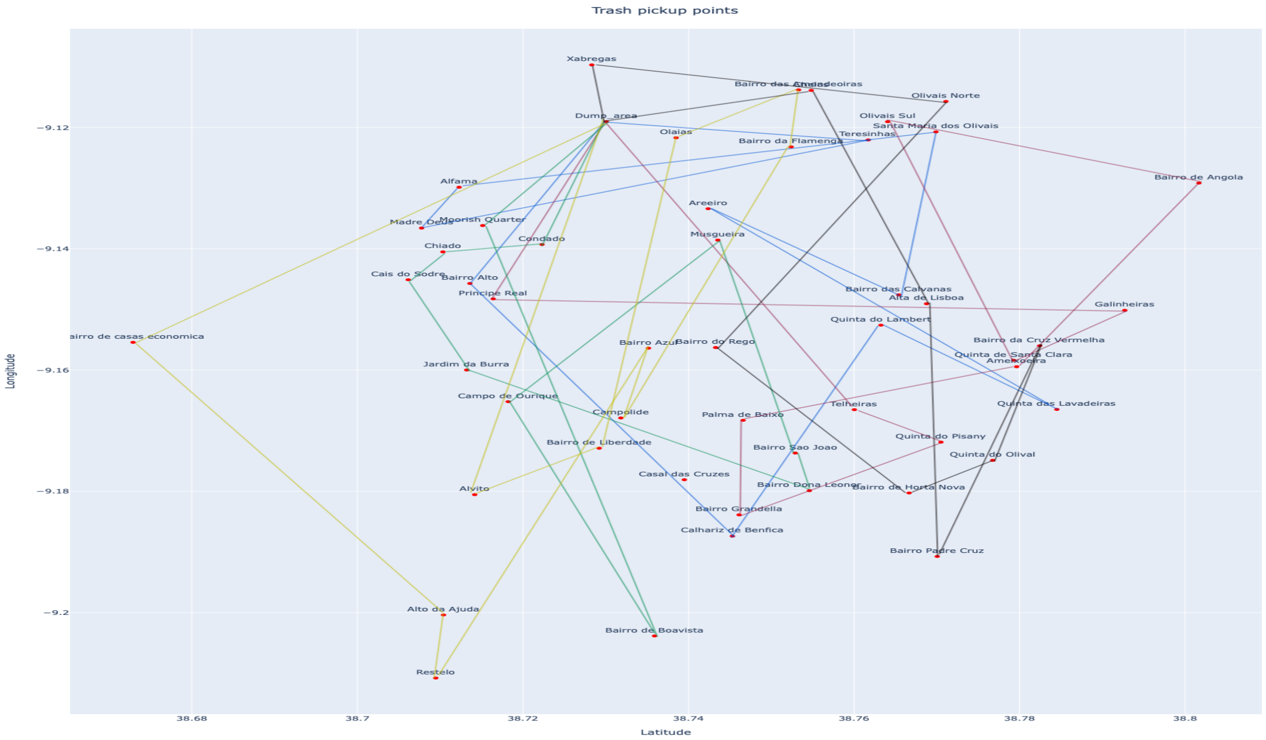


Figure 4: Shortest distance trips, each truck represented by color.

4. Conclusion

In this project computational intelligence techniques for optimization were applied to a trash collection routing problem. The problem was classified as a capacitated vehicle routing problem and the fitness functions were developed accordingly. The solutions were then optimized using the evolution process by applying selections and variations techniques to improve the fitness of the individuals. Cycle crossover and inversion mutation with the application of elitism gave the best solution with the fitness of 186.52. The solution was then plotted on the graph to visualize the trucks routes.

For future work, the project will be implemented in a sense that the algorithm can pick the number of trucks with different capacities.