E-ROUTE: AN ELECTRIC VEHICLE ROUTING PROBLEM

Juan Pablo Giraldo R Universidad
EAFIT Colombia
Jpgiraldor@eafit.edu.co

Juan Felipe Londoño G Universidad
EAFIT Colombia
jflondonog@eafit.edu.co

Mauricio Toro
Universidad Eafit
Colombia
mtorobe@eafit.edu.co

ABSTRACT

A very big problem that companies such as as Amazon, FedEx, UPS and many others had was called the travelling salesman problem. The problem consisted in being able to go to all the needed destinations while spending the least amount of gas(energy), distance and time. The problem has already been solved but in our case we have been presented with a similar problem with certain changes. The problem consists in a delivery company that needs to deliver packages to clients. The delivery trucks are electric meaning they need to charge the batteries taking into account that in order to charge them it takes a long time compared to gasoline engines. Also the routes taken by the truck have a maximum amount of time limit.

We are proposing a solution using a greedy technique completely designed by us. From the beginning we new it would not be the most optimal solution compared to others. Regardless we decided to try. No results were captured due t the fact that the code is unfinished but with other implmentations found, we were able to conclude that greedy is the least optimal compared to others. To conclued, to find the perfect solution to this problem is hard but close solutions can be found. Comparing with other methods, we found that Tabu Search is sometimes the most optimal but in other occasions intra or inter route search can be better but in all cases the greedy implementation came in last.

TSP, CVRP, Greedy, Electric Vehicle Routing Problem

Software and its engineering---Software notations and tools---General programming languages---Language features---Objects

Software and its engineering---Software notations and tools---General programming languages---Language features---Operation between objects

Theory of computation → Design and analysis of algorithms → Graph algorithms analysis → Shortest paths

1. INTRODUCTION

In the near future it an be said that most vehicles will rely solely on energy. The world is heading towards electric vehicles. Due to this delivery companies will encounter different problems to the already existing ones. One of them is a variation of TSP (travelling salesman problem) which will search the shortest route possible in order to deliver all the packages. Also it's important to take into account that in order to charge the batteries it takes more time than refueling a petrol car. In our derivation of the problem the battery capacity will be lineal, the route between two different places will be straight and there is an unlimited amount of trucks.

2. PROBLEM

The problem to solve as said in the abstract is to find the best way to travel the different destinations taking in to account that there is a maximum amount of time to deliver the packages, that the vehicles are electric and need to be recharged and it takes a noticeable amount of time. Also there is an unlimited amount of trucks to deliver the package.

3. RELATED WORK

3.1 TSP(Travelling salesman problem) The travelling salesman problem also known as TSP, is problem which was first defined in the 1800s by the irish and british mathematicians W.R. Hamilton and Thomas Kirkman. The basis of the problem is to travel to all the needed destinations while spending the least amount of time, money, distance or other factors.This problem is seen as a difficult problem to solve because if it is possible to break the problem into smaller ones those would be as complex as the first one. The solution is to find the least expensive route(most optimal) and verify that there is not one that is better.

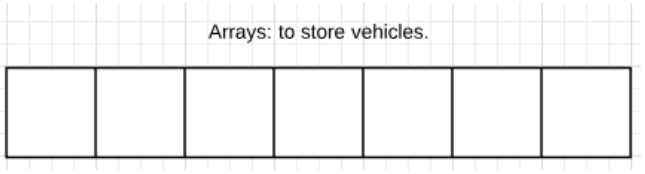
3.2 Dijkstra's Algorithm Dijkstra also known as the shortest path algorithm is an algorithm for finding the shortest route between two nodes, given a start and end vertex.The subjacent idea in this algorithm consists of exploring all the possible paths that begin in a certain vertex and selecting the shortest one. When found the algorithm stops and returns the shortest path, not necessarily the path but the shortest distance. It doesn't work in weighted graphs with negative values because with this it could show longer routes because it would subtract the values.

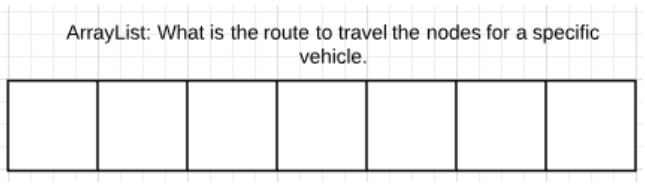
3.3 **CVRP(Capacitated vehicle routing problem)** CVRP consists of adding to all the vehicles a standard capacity of the traditional VRP(Vehicle routing problem) in a way to satisfy the clients' needs. Its objective is to minimize the amount of vehicles and time satisfying the demand and not exceeding the vehicle capacity. The problem is symmetrical when the distance from i to j and j to i iis the same, if not it's asymmetrical.

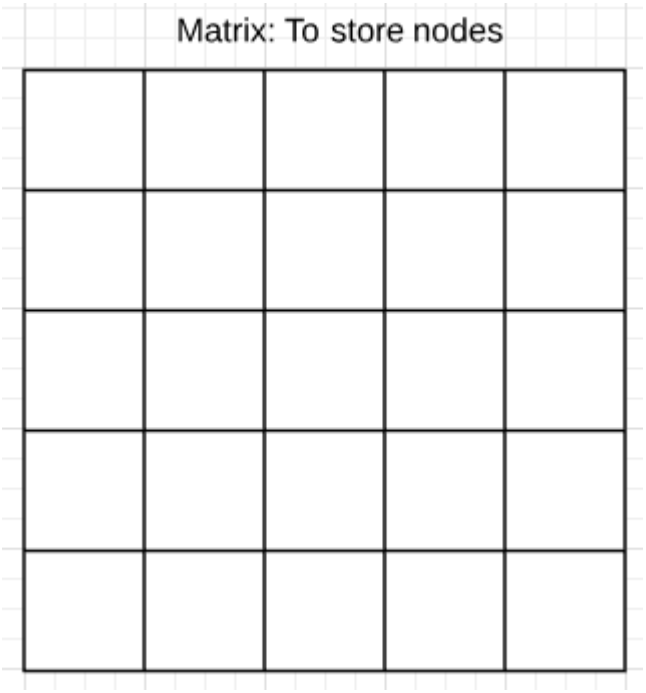
3.4 VPRB (The vehicle routing problem with backhaul) VPRB (The vehicle routing problem with backhaul), considers that the vehicle delivers all the cargo that it began with. The authors model a unified problem of delivery clients(Linehaul) and of return(Backhaul). The problem is solved using the heuristic of local search. It uses a metaheuristic algorithm named Tabu search to solve the routing problem whom itself divides into subroutes, one for the delivery clients and another for th return in order to obtain the least expensive solution.

4. GREEDY-TYPE IMPLEMENTATION

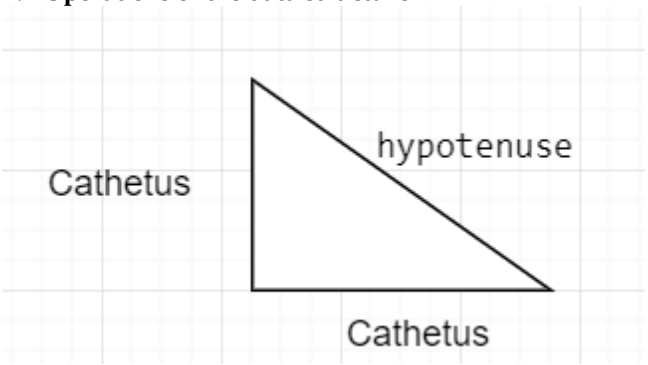
4.1 Data Structure







4.2 Operations of the data structure



CalculateDistance: Applies the Pythagorean Theorem. First get the positions in X and Y for the two nodes. Subtract the b-a to obtain the value of the side of the triangle known as cathetus

$$E = calculateDistance/\text{watss per hour}$$

CalculateEnergy: It uses the method of calculating the distance and that divides the consumption

4.3 Design criteria of the data structure

The matrix was a good way to relate all the nodes with all the nodes because its two dimensional, ArrayList because it has infinite size and Arrays because it is easy to use to access any position

4.4 Complexity analysis

Methods	Complexity
CreateStation	O(1)
CreateDepot	O(1)
CreateClient	O(1)
CalculateDistance	O(n)
CalculateEnergy	O(n)
Complexity total:	$O(n^2)$

Table 1: Table to report complexity analysis without adding the battery, therefore, it can be more.

5. CONCLUSIONS

We had problems with the implementation of the algorithm, for example, we proposed as future work to correct, improve the algorithm and make a graphical interface.

It is important to start working with electric cars because in the future they will be the ones who order the stop, because it helps the environment. And what more than the problem posed: “The problem to solve as said in the abstract is to find the best way to travel the different destinations taking in to account that there is a maximum amount of time to deliver the packages, that the vehicles are electric and need to be recharged and it takes a noticeable amount of time. Also there is an unlimited amount of trucks to deliver the package”.

5.1 Future work

For future work, we would like correct the algorithm, optimize more and plot the routes of the cars

ACKNOWLEDGEMENTS

We would like to thank EAFIT university for helping us become excellent professionals. Our classmates and everyone who helped this project become possible.

REFERENCES

Reference sourced using ACM reference format. Read ACM guidelines in http://bit.ly/2pZnE5g

As an example, consider this two references:

1.Adobe Acrobat Reader 7, Be sure that the references sections text is Ragged Right, Not Justified. http://www.adobe.com/products/acrobat/.

2. Fischer, G. and Nakakoji, K. Amplifying designers’ creativity with domainoriented design environments. in Dartnall, T. ed. Artificial Intelligence and Creativity: An Interdisciplinary Approach, Kluwer Academic Publishers, Dordrecht, 1994, 343-364.

Definition of CVRP through the Chu-Beasley algorithm. http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/4000/5196O75.pdf?sequence=1

Travelling salesman problem - https://en.wikipedia.org/wiki/Travelling_salesman_problem

Dijkstra algorithm - https://es.wikipedia.org/wiki/Algoritmo_de_Dijkstra