Optimization for last mile delivery with electric vehicles

Alexandre Thibaud Ventura Gallon

Valencia & Rome, 2019

Acknowledgements

Agradecer a mis padres y hermanas, haberme apoyado desde que empezó esta aventura que han sido los estudios universitarios hace ocho años. También por permitirme realizar los intercambios que me han traído hasta aquí.
Agradecer, por supuesto, a Giuseppe, que desde marzo de 2016 depositó su confianza en mí, y que sin su ayuda esto no hubiese sido posible, ni ahora ni tres años atrás.
También a Julien por aceptar ayudarme desde España.
Y finalmente acordarme, que no agradecer, de lo sucedido ese 7 de septiembre de 2018, que en cierta forma cambió de forma radical los siguientes meses, y de esos largos e incontables momentos de aviones y esperas en hospitales donde este trabajo ha sido también desarrollado.

I want to thank my parents and sisters for supporting me since this adventure started eight years ago, also for allowing me to do these student exchanges that have brought me to this point.
I also want to thank Giuseppe, since March 2016 he gave me his trust and help. Without him, this work would have not been possible.
Also thank Julien for his help from Spain.
And finally, remember, but not to thank, the accident happened on September 7th, 2018, that changed radically my life for the following months and those long and frequent moments of flights and hospitals where this work has also been developed.

'Grazie (2016), Gracias (2017), Thank you (2018), Grazie (2019)'

Contents

List of Tables

List of Figures

1 Abstract

Electric vehicles powered with batteries are already playing a significant role in the transport and distribution of goods. Their characteristics differ respect to petrol vehicles, such as cruising range, recharging time, energy regeneration when braking. . .
This is why new models have been proposed to adapt the requirements of this type of vehicles.

The model considered is similar and an adaptation of the Vehicle Routing Problem with Pickup and Delivery (VRPPD) of G. Desaulniers, J. Desrosiers, A. Erdmann, M. M. Solomon and F. Soumis. Nevertheless, these authors did not consider electric vehicles and their constraints, but consider time windows, which are not required to solve the studied problem. As the complexity of the problem was high, they present different ways ideas in order to reduce the search space. They suggest to use clustering algorithms proposed by F. Cullen, J. Jarvis and D. Ratfliff to group nodes and simplify the problem. Some meta-heuristics are also proposed by this aspect will be discuss in a following chapter and deeper as a future approaches to keep developing the work done in this thesis.
The original VRP considers a set of identical vehicles, based at a central depot. R. Baldacci, M. Battarra and D. Vigo proposed a model of an heterogeneous fleet of vehicles, similar to the one proposed, as different types of vehicles with different capacities and autonomy of batteries are allowed.
Schneider, Stenger and Goeke made the Electric VRPTW with Recharging Stations adapting the Green VRP to Electric Vehiles (EV) and adding the time windows constraints. The model proposed will be really similar to this one but removing time windows, which simplifies the problem. The recharging problem complicates the problem as the recharging time depends on the battery level. They also proposed some meta-heuristics to solve the problem that will be exposed later. Crevier, Cordeau and Laporte, and Tarantilis, Zachariadis and Kiranoudis have presented approaches where depot can be visited between customers to restock in order to satisfy demand and capacity constraints. Nevertheless, for the studied problem, as the depot is far away from the deliveries zone, restock will be assumed as two different tours and then modelled in two independent orders, which will also simplify the problem formulation. They also consider multi depot, topic that will be discuss in the future approaches section

and its convenience studied. G. Hiermann, J.Puchinger, S.Ropke and R.F.Hartl at their work, propsed a formulation where a heterogeneous electric fleet routing problem is modelled and solved. They also propose meta-heuristics to solve the problem for large instances. The proposed model of this thesis will be similar to the previous one, but once again without considering time windows.
Finally, the work done by A. Felipe, M.T. Ortuño, G.Righini and G.A.Tirado will be also considered as they proposed a VRP with electric vehicles and allowing partial recharges and several recharge technologies. They also propose heuristics to solve the problem. For the studied case, two types (more can be added) of recharging are allowed, electricity coming from the photovoltaic panels and the one coming from the Italian electric network. Furthermore, partial recharging is allowed as vehicle can just charge electricity to complete the deliveries and return to depot, always satisfying the minimum level of battery.

2 Introduction

ZEDL (Zero Emissions Distribution Logistics) project aim is 'to implement a new logistic model with zero CO_2 emissions, with perspectives of economic saving, in the 'Zone a Traffico Limitato (ZTL) di ROMA' with an integrated use of renewable energies source and innovative technologies and warehouses equipped with photovoltaic panels and recharging stations to supply electric vehicles'.

The projects objective is to deliver from 200 to 1000 orders per day in the ZTL, with warehouses located between 10-15 Km far from the city centre. At this moment the warehouse is located in the North-East of Rome.

The objective is to have a fleet of 20 electric vehicles (so far just 4 are performing deliveries). These electric vehicles have a capacity of 2,5 T and an autonomy at dull load of 140 Km. Their recharging time is 7-8h.

The company was using to plan the routes of deliveries an algorithm that did not give the optimal route. The purpose of this work has been to create a model, that taking into account a list of nodes to visit, the load of deliveries required to serve and the level of battery of the vehicles, was able to give better results than the previous model.

In the next pages that model will be presented and explained. Furthermore several cases of study will be simulated in order to prove the convenience of using the proposed model. Some changes in the parameters will be also made in order to see how the model behaves and finally, some topics will be presented as future approaches to be considered and developed by the reader.

3 Model

3.1 Sets

- I set of nodes
- A set of arcs
- K set of vehicles
- P set of orders
- T set of recharging technologies

3.2 Parameters

- $c_{i,j}$ distance between nodes i, j
- $b_{i,j}$ battery consumption between nodes i, j
- c_k maximum battery capacity for vehicle k
- $cr_{i,t}$ recharging cost at node i with technology t
- s_k node source for vehicle k
- $t_k$node target for vehicle k
- $rt_{i,k,t}$ recharging time for vehicle k at node i with technology t
- H time that the driver can drive without stopping
- F fixed cost per stop

3.3 Variables

- $x_{i,j,k} = \begin{cases} 1 & \text{if arc } i,\ j \text{ is traveled by } k \\ 0 & \text{otherwise} \end{cases}$

- $y_{p,k} = \begin{cases} 1 & \text{if order } p \text{ is loaded to vehicle } k \\ 0 & \text{otherwise} \end{cases}$

- $a_{i,k}$: arrival time for vehicle k at node i.

- z_k : number of stops required by vehicle k.

- $L_{i,j,k}$: load of vehicle k when traveling arc i, j.

- $ca_{i,k}$: level of battery of vehicle k when arriving at node i.

- $cl_{i,k}$: level of battery of vehicle k when leaving at node i.

- $cc_{i,k,t}$: amount of battery charged by vehicle k at node i with technology t.

3.4 Objective Function

$$\min \sum_{i,j \in A, k \in K} c_{i,j} x_{i,j,k} + F \sum_{k \in K} z_k + \sum_{i \in I, k \in K, t \in T} cc_{i,k,t} cr_{i,t} + \sum_{i \in I, k \in K} \frac{-cl_{i,k}}{M}$$

3.5 Constraints

1. $\sum_{k \in K} y_{p,k} = 1 \qquad \forall\ p \in P$

2. $\sum_{j \in \delta^+(s(k))} x_{s(k),j,k} = 1 \qquad \forall\ k \in K$

3. $\sum_{j \in \delta^-(t(k))} x_{i,t(k),k} = 1 \qquad \forall\ k \in K$

4. $\sum_{j \in \delta^-(i)} x_{j,i,k} = \sum_{j \in \delta^+(i)} x_{i,j,k} \qquad \forall\ k \in K, i \in N \backslash \{s(k), t(k)\}$

5. $a_{i,k} + rt_{i,k,t} - a_{j,k} \leq (1 - x_{i,j,k})M \qquad \forall (i,j) \in A, k \in K, t \in T$

6. $a_{l(p),k} \leq a_{m(p),k} + M(1 - y_{p,k}) \qquad \forall p \in P, k \in K, t \in T$

7. $z_k \geq \frac{a_{t,k}}{H} - 1 \qquad \forall k \in K$

8. $\sum_{j \in \delta^+(l(p))} x_{l(p),j,k} \geq y_{p,k} \qquad \forall p \in P,\ k \in K$

9. $\sum_{j \in \delta^-(m(p))} x_{i,m(p),k} \geq y_{p,k} \qquad \forall p \in P,\ k \in K$

10. $\frac{L_{i,j,k}}{C_k} \leq x_{i,j,k} \qquad \forall (i,j) \in A, k \in K$

11. $\sum_{i \in \delta^-(i)} L_{i',i,k} \geq \sum_{p \in P: m(p)=i} d_p y_{p,k}$ $\forall i \in N,\ k \in K$

12. $cl_{s(k),k} = c_k$ $\forall k \in K$

13. $c_k x_{i,j,k} + b_{i,j} x_{i,j,k} + ca_{j,k} - cl_{i,k} \leq c_k$ $\forall (i,j) \in A,\ k \in K$

14. $\sum_{t \in T} cc_{i,k,t} + ca_{i,k} - cl_{i,k} = 0$ $\forall i \in N,\ k \in K$

15. $x_{i,j,k} \in \{0,1\}$ $\forall (i,j) \in A, k \in K;$

    $y_{p,k} \in \{0,1\}$ $\forall p \in P, k \in K;$

    $a_{i,k} \geq 0$ $\forall i \in I, k \in K;$

    $z_k \geq 0$ $\forall k \in K;$

    $L_{i,j,k} \geq 0$ $\forall (i,j) \in A, k \in K;$

    $c_k \geq ca_{i.k} \geq 0,3c_k$ $\forall i \in I, k \in K;$

    $c_k \geq cl_{i.k} \geq 0,3c_k$ $\forall i \in I, k \in K;$

    $cc_{i.k,t} \geq 0$ $\forall i \in I, k \in K, t \in T$

3.6 Model explanation

This model has been based on a multi-commodity and vehicle routing based model for long haul shipment.
Nevertheless, it has been adapted to the last mile delivery and some electric constraints have been added in order to adapt the previous model to the requirements of this thesis.

Furthermore, it will shown that the scope of the model is wider than the required for the problem and it is able to make some calculations that are not required at this point, but they will be explained in the following chapters as future scenarios may need to consider them.

The objective function is divided into four terms. The first is related to the traveled distance. The second term, takes into account a fixed cost that is related to the number of stops. The third term takes into account how much battery is recharged at every node and the cost of it. The fourth and last term considers the level of battery.
It could seem that different magnitudes are being minimized (distance, monetary cost and battery level), but in the end, all of them depend on the distance, so it is important be careful and to write properly the units of some parameters in order to have coherence.

Regarding to the constraints, (1) ensures that every order p is assigned to a certain vehicle k. (2) ensures that all used vehicles leave source node and (3) that all arrive to the depot node. Constraint (4) is a continuity constraint, which says that all the vehicles arriving to a certain node, have to leave this node (it does not apply to depot and target as they have to be considered in a different way as previously has been explained).
Constraint (5) is related with arriving times. It the arcs *(i,j)* is traveled, time at j has to be equal or higher than time at i plus the recharging time spent at this node. Constraint (6) ensures that time at the delivery node is equal or higher than at the origin node. (7) is related with the number of stops needed. (8) ensures that a vehicle leaves the origin node from an order and (9) that a vehicle arrives to a delivery node for any order. (10) is a capacity constraint, which says that the load in a vehicle cannot exceed the maximum capacity and (11) guarantees that the demand of a certain delivery node is satisfied with the

load of the vehicle that does the delivery.
The following constraints are related with the electric considerations. (12) ensures that the vehicle leaves the source node fully charged. (13) ensures a continuity in the battery of the vehicle. (14) ensures that battery when arriving to a node plus battery charged at this node is equal to the battery when leaving. Finally (15) shows the nature of variables.

Our case of study does not require to consider the number of stops of the driver. In the long haul shipment drivers must to do a stop every certain time, in order to allow them to rest.
As we do not need to use this parameter we can remove its contribution to the objective function making the cost of the stop (F) equal to 0. We can also make the number of stops (z_k) equal to zero giving to the parameter (H) a big vale M.
This model could be applied to countries where the legislation dictates than stops every a certain period of time is mandatory, just modifying these parameters.

Furthermore, the problem has been simulated just allowing recharging at the depot node. But the model is ready to allow recharging at every node of the network if possible. This approach would be explained in a following chapter, as a future developing topic.
The last term of the objective function has as objective to show the real battery level (the constraints make continuity in the battery possible). As its contribution on the objective function is desired to be as low as possible, it is divided by a big term so it does not affect to the final value but makes this continuity possible.
In order to make the total distance more precise, a summation of the overall path will be done, as well as the total battery consumption (even if it is dependent on the distance traveled).
Finally, has been observed than in real deliveries, sometimes capacity of the vehicle was exceed. This model does not allow to surpass the capacity, but an approach where surplus of load would be possible and penalized could be considered.

4 Distances

A set of 2625 nodes, with their longitude and latitude, representing all the different locations where goods can be delivered has been given.
In order to calculate the distance between them, different approaches have been considered.
Due to the big amount of data available, it was not possible to calculate all the combinations between nodes. There are different ways to calculate the amount of combinations:

$$N(x) = x^2$$

Nevertheless, this formula considers the distance between a node and itself, which does not make sense for the proposed problem, so there is some waste of computational time to make these calculus.
The following formula does not take into account the previous combination.

$$N(x) = x^2 x - (x - 1)$$

Furthermore, it does not consider combinations between source node and target node, as they are same location. The algorithm has also been programmed to calculate also combinations where source and target are different.
For the studied scenario the amount of possible combinations would be:

$$N(2625) = 2625^2 - 2625 - 2624 = 6.885.376$$

As the program also calculates battery consumption for each distance, the amount of calculus is 13.770.752.
Orders are much smaller, between 10 and 20 visited nodes. This would mean:

$$N(20) = 20^2 - 20 - 19 = 361$$

As it can be observed, it does not make sense to work with such a big networks because computational time is being wasted and just $0,00524\%$ of the combinations would be used.

Once the problem dimensions have been defined, it is time to consider different ways to approach the distance problem.

In the first place, Euclidean distance was considered.
Being points $P_1(x_1, y_1)$ and $P_2(x_1, y_2)$, distance between them is given by the formula:

$$d_{12} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

This distance is not really accurate for points in cities, as the path followed by the vehicle is not a straight line. This is why a new approach was considered.

Taxicab metric (also known as Manhattan distance) calculates the sum of the absolute difference of Cartesian coordinates between two points. Being points $P_1(x_1, y_1)$ and $P_2(x_1, y_2)$, distance between them is given by the formula:

$$d_{12} = |x_1 - x_2| + |y_1 - y_2|$$

This distance is higher than Euclidean distance and most of the times is more accurate. In the next picture a comparison between both approaches is shown.

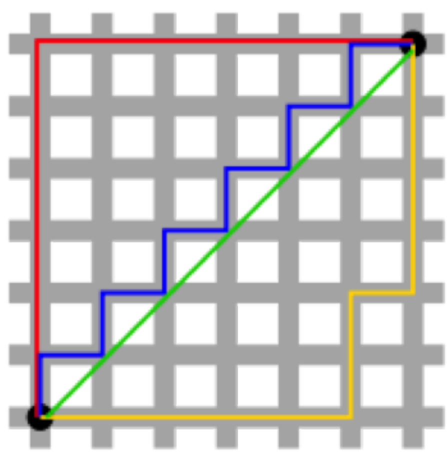

Figure 1: Manhattan distance

Green line corresponds to Euclidean distance, which is the shorter. Blue, red and yellow are different ways to connect both points with the same length.

Even if taxicab metric approach seems to be correct, a deeper study on the networks has shown that there was a big gap between distances given by the formula and real path done by trucks.
The reason behind that difference is that the warehouse from which deliveries are done is quite far from the city. So it does not make sense to consider Manhattan distance as it increases a lot the real distance.
Using Euclidean distance until the city borderline and then the taxicab metric inside the city would be a more accurate way to calculate the distance, but the problem would be to define properly where this borderline is exactly location.

A last approach has been used and it is considered the most accurate of all of them.
Haversine formula determines the distance in a sphere between two points given their longitude and latitude.

$$d = 2r * \arcsin \sqrt{sin^2 \frac{\phi_2 - \phi_1}{2} + cos(\phi_1) cos(\phi_2) sin^2 \frac{\lambda_2 - \lambda_1}{2}}$$

where:
- ϕ_1, ϕ_2 are latitude of points 1 and 2,
- λ_1, λ_2 are latitude of points 1 and 2.

Is used in navigation but it is also useful to measure the distance between points that are far from each other. Similar to Euclidean distance as it gives a straight line between two points, this one also takes into account the sphere shape so gives better results. Moreover, the problem of accuracy for distance in the city centre still happening. This is why a more exhaustive and last study has been done.

Fifty pairs of nodes have been studied in order to compare their Haversine distance with the real driven distance. For determining the driven distance, a website that uses Google Maps database, has been used to make the comparison.

From the study several deductions are made. This fifty pairs can be divided in two main groups. 24 of them have the warehouse (depot node) in the pair and 26 of them are just a connection between nodes randomly selected from the 2625 locations.

This division is important because a factor that multiplies the Haversine distance is going to be calculated and it is going to be different for each one of these groups.

For the first group (the ones that have depot node) and studying the relation between the divisions of the real distance by Haversine distance an index is obtained. This index has a range between [1,384 – 2,099], with an average of 1,6855. To know if that index can be accepted as good, standard deviation of the set is calculated. The result is 0,22, which is a good value.
For the second group, same procedure is done. Now the range is wider [1,227 – 6,398] with average of 3,087. Standard deviation is much higher 1,6514.
This cannot be accepted as good so a deeper study of these pairs of nodes has to be done. The reason behind these big differences is that certain nodes that are in the city centre of Rome, the area where most of the touristic monuments are located, give a bigger index than the rest of the nodes. This is because in the city centre driven distances are much bigger than Haversine distance.
So a third group of nodes has been identified and will be treated in a different way. These nodes (will be called in advance city centre nodes) belong to areas located in neighbourhood with postal codes between 00184 and 00189.
Removing from the first set (the one with depot node) these city centre nodes and making all the calculations again, a new range is determined [1,479 – 1,823] with average 1,633. The standard deviation is now 0,1154. This is again considered as good and will be the factor that multiplies the Haversine distance for pairs that contain depot node.

For the second set (nodes that do not belong to post codes between 00184 and 00189), same procedure is done, and a new range is determined [1,716 – 3,531] as well as a new average of 2,424. The standard deviation of this set is now 0,628, which is considered as good.

Finally, for the set of nodes belonging to the city centre areas, the range is wide [1,227 – 6,398] and the average 3,751, with a standard deviation of 2,081. It is true that it could seem a really bad idea to accept this average as the index multiplying Haversine distance for those nodes. Nevertheless, as in the city centre distances are much smaller that pairs belonging to other sets, that value is going to be accepted, as the maximum error in the distance is 4km (next error is just 2km).

The purpose of all of these calculations is to give a starting point to simulate the model and finding the optimal route (or a close one to the optimal). Once deliveries are performed and real distances are known and recorded by the GPS located in each vehicle, they will replace the approximate distances calculated with the approach explained.

5 Cases of application

5.1 Introduction

Several orders done by the drivers have been simulated in order to show how an improvement in the route could be achieved.

In this section results will be shown and explained to prove the convenience of using this algorithm. In all of them is possible to observe a change in the sequence of visited nodes that produces the reduction of the total distances as a consequence of this new path.

The first four cases will focus on proving the benefit of using this algorithm instead of using the one that the company had previously.

The next two cases will analyze how the order consolidation affects to distance reduction and computational time.

To continue, the following two cases will simulate again two of the previous scenarios with real distances. Will be possible now that a change in the sequence is produced, but always, the result given by this model is better than the one given by the previous algorithm.

The next case will modify the location of the warehouse to see how the model behaves.

Finally, the last case will show a relation between nodes and computational time.

5.2 Case 1

The next deliver correspond to the order identified with code '18064' done by vehicle MODEC3 on 3/4/18.

Driver's path	Simulated Path
77	77
2	1
8	7
6	5
4	3
5	4
1	6
3	8
7	2
77	77

Table 1: Case 1: visited nodes sequence

Driver	57,82 Km
Simulation	52,98 Km

Table 2: Case 1: distances

As it can be observed, route done by the simulation is shorter that path followed by the driver. Actually, it has improved an 8,385%.
Furthermore, the computational time used to solve this instance has been 0,36s for 8 orders.

5.3 Case 2

The next deliver correspond to the order identified with code '18068' done by vehicle MODEC2 on 3/4/18.

Driver's path	Simulated Path
77	77
9	13
10	15
11	12
14	16
16	10
13	11
15	14
12	9
77	77

Table 3: Case 2: visited nodes sequence

Driver	49,36 Km
Simulation	47,89 Km

Table 4: Case 2: distances

In this instance, the improvement achieved is lower than the previous instances, but still being a reduction of the total distance, 2,99%.
The computational time spent is 0,16s for 8 orders.

5.4 Case 3

The next deliver correspond to the order identified with code '18304' done by vehicle MODEC3 on 4/4/18.

Driver's path	Simulated Path
77	77
37	37
28	28
40	39
39	40
34	27
31	33
38	31
30	38
29	30
35	36
20	32
25	24
17	23
21	18
22	22
18	21
23	17
19	19
26	26
24	29
36	25
32	35
33	20
27	34
77	77

Table 5: Case 3: visited nodes sequence

Driver	66,85 Km
Simulation	61,48 Km

Table 6: Case 3: distances

There is also an improvement in this instance. Now the improvement has been 8,024%.

Nevertheless, as the number of orders has increased, the computational time has also increased. It has been $3s$. Compare to the previous cases is almost 10 times bigger, while the number of orders is just 3 times bigger. This gives an idea about how big computational time could be for bigger instances, where meta-heuristics could be required to solve the problem. But this point will be explained deeper in following sections.

5.5 Case 4

The next deliver correspond to the order identified with code '18305' done by vehicle MODEC6 on 4/4/18.

Driver's path	Simulated Path
77	77
42	55
6	49
50	41
48	57
43	45
46	47
52	54
44	53
51	56
56	51
53	52
54	44
47	48
45	46
57	43
41	50
55	6
49	42
77	77

Table 7: Case 4: visited nodes sequence

Driver	57,8 Km
Simulation	56,29 Km

Table 8: Case 4: distances

The improvement for this instance is 2,6%.
The computational time to simulate this instance of 18 orders was 0,45s.

5.6 Case 5

Studying again the order of case 3, '18304', and focusing on the path done by the simulation, benefits of order consolidation are going to be shown.
The instance had 24 orders. If it is divided into two instances of 12 orders each:

First division
77
37
40
39
28
34
20
35
25
29
30
28
31
77

Second division
77
17
21
22
18
23
19
26
24
36
32
33
27
77

Table 9: Case 5: visited nodes sequence

First division	45,41 Km
Second division	55,36 Km

Table 10: Case 5: distances

The sum of the two instances gives a total distance of 100,77 Km, while the consolidated instance was 61,48 Km. This means an increase of 39,99%.
Regarding to computational time, for the first division took 0,36s to complete the simulation. For the second division, 0,37s. The sum is 0,73s. Compared to the previous time that was 3,00s, means a reduction of 2,27s which is a 310,95%.
As observed, there is a trade-off between computational time and distance reduction due to order consolidation.
Once again, use of meta-heuristics for big instances would be beneficial.

5.7 Case 6

Finally, to prove the consistence of the argument explained in the previous case, another order consolidation will be simulated. Now, order '18305' will be divided into two instances of 9 orders each.

First division
77
42
6
50
43
46
48
44
52
51
77

Second division
77
55
49
41
57
45
47
54
53
56
77

Table 11: Case 6:visited nodes sequence

First division	44,05 Km
Second division	50,29 Km

Table 12: Case 6: distances

The sum of the two instances gives a total distance of 94,34 Km, while the consolidated instance was 56,29 Km. This means an increase of 67,59%. Regarding to computational time, for the first division took 0,19s to complete the simulation. For the second division, 0,18s. The sum is 0,37s. Compared to the previous time that was 0,72s, means a reduction of 0,35s which is a 51,14%. As observed, there is a trade-off between computational time and distance reduction due to order consolidation.

5.8 Case 7

Order '18064' has been simulated again but now considering real distances.
As it can be observed, the sequence has changed respect to the path done by the driver. Once again, the result given by the model is better than the one given by the algorithm previously.

Driver's path	Simulated Path
77	77
2	2
8	8
6	5
4	6
5	1
1	4
3	3
7	7
77	77

Table 13: Case 7: visited nodes sequence

Driver	44,42 Km
Simulation	43,44 Km

Table 14: Case 7: distances

Distance has been reduced, 0,98 Km, which means an improvement of 2,21%. If we compare the sequence obtained using assumed distances (at case 1) with real ones, it is possible to observe that the sequence has also changed. This is because as they were multiplied by a factor, the result of these multiplications sometimes was higher than real distances and sometimes lower, producing a different optimal sequence.
Nevertheless, the model, as said before, gives better results no matter if assumed distances or real distances are being used.

5.9 Case 8

Order '18305' has been also simulated again with real distances. Same conclusions as the previous case can be extracted.

Driver's path	Simulated Path
77	77
42	6
6	48
50	46
48	44
43	52
46	51
52	53
44	56
51	54
56	47
53	45
54	41
47	57
45	55
57	49
41	43
55	50
49	42
77	77

Table 15: Case 8: visited nodes sequence

Driver	40,65 Km
Simulation	39,08 Km

Table 16: Case 8: distances

The improvement in the total distance has been 1,57 Km, which means a reduction of 3,85%.

5.10 Conclusions of cases 1-8

The substitution of the previous algorithm by the model explained at this thesis is justified with an improvement of the total distance travelled by the vehicle independently of distances used (assumed or real).
Furthermore, benefits of orders consolidation have also been proved, but there is a trade off between distances reduction an increase of computational time.

5.11 Case 9

Next case is going to study the modification of the depot distance in order to study how the model behaves.
The reason of considering an increase of the distance is because is interesting to see how much the distance increases when increasing the depot distances.
The order selected to complete these simulations is going to be order '18305', using real distances. Depot distance is going to be multiplied by different factor and them and their distances are shown in the next table.
Sequences are not relevant now as the only objective is to study total distance increase.

Factor	Distances
1	39,08
1,1	41,81
1,2	44,54
1,3	47,28
1,4	50,01
1,5	52,74
1,75	59,56
2	69,39

Table 17: Case 9: Multiplying factor and distances

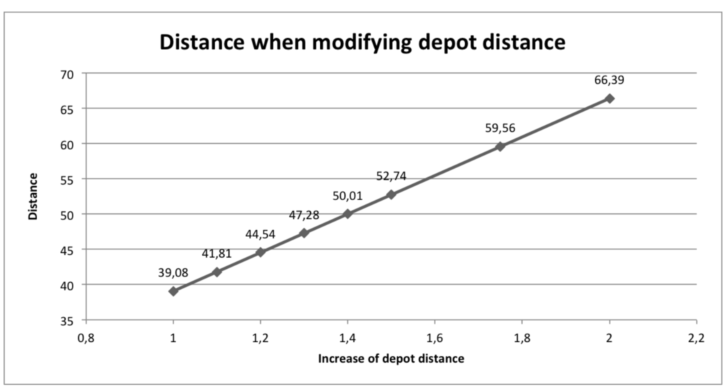

Figure 2: Distance increased when changing depot distance

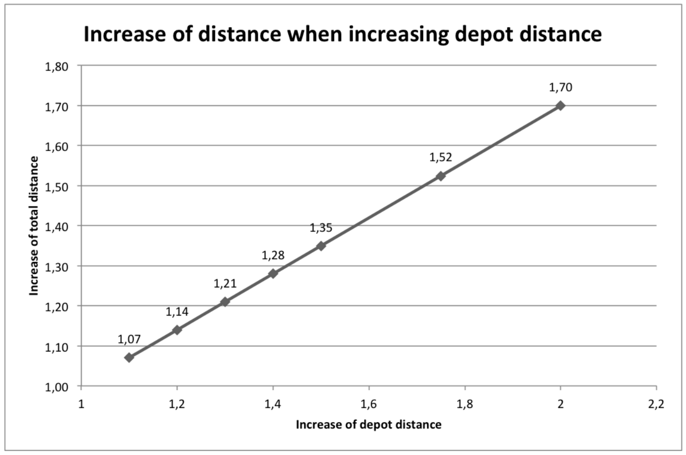

Figure 3: Proportion increased when changing depot distance

These types of scenarios could help the company to consider a relocation. A further warehouse locations would imply longer distances, higher battery consumption and higher cost. Nevertheless, further distance would mean cheaper warehouse renting cost, so several studies can be carried out in order to determine the convenience of the change.

5.12 case 10

Different orders have been simulated in order to study the relation between nodes and computational time. Results are shown graphically:

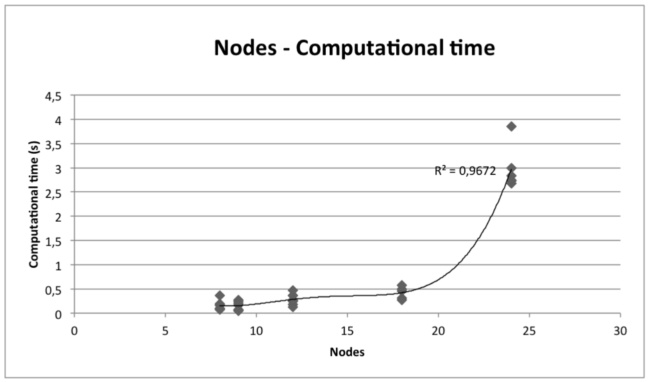

Figure 4: Computational time depending on scenario dimensions

As it can be observed, orders with similar number of nodes produce different results, even if all of them have similar computational time.
As said before, an increase of the number of nodes produces higher computational times. Once again, meta-heuristics will be required to solve bigger instances. The time increases exponentially.

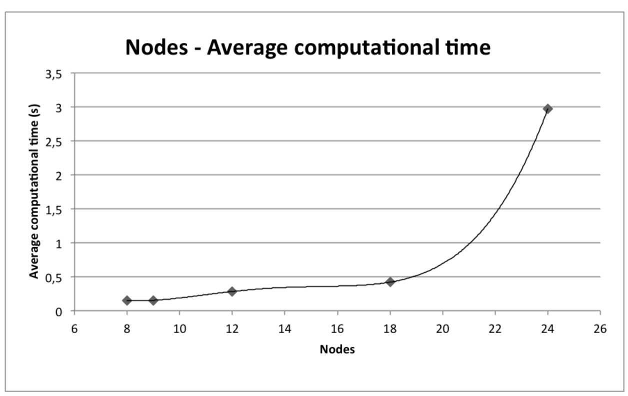

Figure 5: Average time for different scenarios

The equation of the curve can be obtained and forecasts of computational time can be made.

6 Future approaches

6.1 Meta-heuristics application

As the studied problem is a modification of the Vehicle Routing Problem (VRP) and this one is known as a NP-hard problem, it will also be a NP-hard problem. As explained in previous sections, the increase of the number of orders produces a huge increase on computational time. This is why for these type of instances meta-heuristics must be used.
It is not the purpose of this thesis to explain these meta-heuristics but to show which ones have been used to solve problems with similar characteristics.

Bent and Van Hentenryck at their work *'A two-stage hybrid for pickup and delivery'* vehicle routing problems with time windows' proposed to use Large Neighbourhood Search (LNS) and Simulated Annealing (SA).

Schenider et Al, at their work *'The Electric Vehicle Routing Problem with Time Windows and Recharging Stations'* consider that the best way to solve these problems is to apply Variable Neighbourhood Search (VNS) with Tabu Search (TS).

Hiermann et Al at *'The Electric Fleet Size and Mix Vehicle Routing Problem With Time Windows and Recharging Stations'* determine that the best way of obtaining a good solution is Large Neighbourhood Search (LNS) and an embedded local search and labelling procedure for intensification.

As it can be observed, different theories and meta-heuristics are applied to solve problems with similar.
Investigating the suitability of applying certain meta-heuristics is left as an approach for future ways of continuing this thesis.

6.2 Real distances

As explained in previous sections, distances have been calculated in an approximate way, but they do not correspond to real ones.
They are proportional (multiplied by a factor) to the real ones, so their accuracy could modify the sequence of visited nodes.
Nevertheless, as a consequence of order consolidation, longer deliveries will be computed and remaining battery and its minimum level could suppose a constraint limiting the vehicle range.

For the previous simulated orders, battery level was not a constraint due to the minimum level was never reach. But it could be a problem in the future.
It is true that the factor considered previously has been determined after an exhaustive study and most of the distances are longer than real ones.
Most of them, but not all, so it would be possible to have an instance with a combination of nodes where the simulation gives shorter distance than the real done by the vehicle.
This problem could happen mainly for orders that have lot of deliveries at the city center of Rome, where variability of the multiplying factor was higher that other neighbourhoods of the city.
If for this instance the remaining battery is achieving its minimum, some constraints could be violated in real instances but not in the simulation.

Use of off-line maps allow to obtain the real distance. As the objective of this work was mainly to prove the convenience of using this algorithm and not to calculate the exact distance, calculation of proper distances is proposed as a future approach.
The benefits of using these distances will avoid to compute orders feasible just in the simulation but not in real instances.

6.3 Recharging stations

So far, the company just allows a recharging station where the warehouse is located. Nevertheless, the model is prepared and has also been tested (even if these results have not been showed as they do not give any relevant information for the purpose of the thesis) allowing recharge at every node.

The convenience of recharging at any node (or some of them in the city center) allow to simulate longer orders without returning to the main warehouse, which is far from city center, where most of the orders are delivered.
Once again, benefits of order consolidation would be applied. The convenience of locating these new recharging stations and studying the trade-off between order consolidation and recharging time (during this time vehicle could not deliver order and its efficiency would be reduce) is also left as a future approach that could be developed.

6.4 Distance calculation integration

The model has as an input the orders and distances between them. This input is calculated by another independent model. Considering the integration of this model (or a similar one that allow the calculation of distances between a combination of given nodes) will allow to reduce the input data just to nodes that are desired to visit.

This integration has not been performed as it inclusion will suppose an increase of computational time. For big order, meta-heuristics could be also required just to calculate distance combinations.
This is why this integration is also left as a future way of developing this thesis.

6.5 Warehouse relocation convenience

In case 9, warehouse distances has been modified. As exposed briefly before, this produce an increase in the overall cost as more battery is consumed. Nevertheless, renting decreases with distance to city centre.

There is then a trade off between increased delivery cost and reduction of renting. Its minimum would be the optimal solution. This work has not considered this study but leaves it for the reader as a future consideration to develop.

6.6 Warehouse decentralization

Decentralization has positive and negative effects. It is know that a central warehouse obtains benefits of aggregated demand and risk pooling. Nevertheless, there is an increase in transport cost.

Meanwhile, decentralization reduces transport cost but has an increase of stock cost, as safety stock must be higher.

The model has been programmed in order to allow different depot and target sources, so a deeper analysis could be carried out in order to determine the convenience of decentralization.

7 References

1. G.Desaulniers, J.Desrosiers, A.Erdmann, M. M. Solomon and F. Soumis (2002), VRP with pick up and delivery.

2. F.Cullen, J.Jarvis and D.Ratfliff (1981), Set partitioning based heuristics for interactive routing

3. M.Schneider, A.Stenger and D.Goeke (2014). *Electric Vehicle Routing Problem with Time Windows and Recharging Stations.*

4. R. Baldacci, M.Battarra and D.Vigo,(2008). Routing a heterogeneous fleet of vehicles.

5. B.Crevier, J.F.Cordeau and G. Laporte (2007), The multi-depot vehicle routing problem with inter-depot routes.

6. C.D.Tarantilis, E.E.Zachariadis,and C.T.Kiranoudis (2008). A hybrid guided local search for the vehicle-routing problem with intermediate.

7. R.Bent and P.Van Hentenryck (2004), A two-stage hybrid algorithm for pickup and delivery vehicle routing problems with time windows.

8. A. Felipe, M.T. Ortuño, G.Righini and G.A.Tirado (2014), A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges

9. G. Hiermann, J.Puchinger, S.Ropke and R.F.Hartl (2016). The Electric Fleet Size and Mix Vehicle Routing Problem with Time Windows and Recharging Stations.

10. G.Stecca and V.Contente (2018). A multi-commodity and vechilce routing based model for long haul shipment.

11. C.Lozzi (2018). *Progetto ZEDL: analisi e miglioramento della distribuzione ultimo miglio con veicoli elettrici.*