

A heuristic algorithm for the Multi-Depot Electric Location-Routing Problem with Time Windows, Battery Swapping and Partial Recharging

Victor Hugo Vidigal Corrêa

Universidade Federal de Viçosa

Av. Peter Henry Rolfs, s/n - Campus Universitário, Viçosa - MG, 36570-900

victor.vidigal@ufv.com

André Gustavo dos Santos

Universidade Federal de Viçosa

Av. Peter Henry Rolfs, s/n - Campus Universitário, Viçosa - MG, 36570-900

andreufv@gmail.com

RESUMO

Veículos elétricos se tornaram uma tendência nos últimos anos e um rápido avanço em suas tecnologias pode ser visto diariamente. Como toda tecnologia em ascensão, implementar uma infraestrutura boa e confiável é um passo crucial em sua adoção. Técnicas de otimização podem ser usadas para reduzir custos e torná-la mais atraente para empresas de logística aderirem ao uso desse meio de transporte. Neste artigo, apresentamos um algoritmo para o problema de roteamento e localização de veículos elétricos com múltiplos depósitos, janelas de tempo, troca de bateria e recarga parcial. O algoritmo proposto é composto por uma metaheurística Simulated Annealing com a busca local realizada por uma metaheurística GRASP-VNS. Comparamos os resultados deste algoritmo com o modelo matemático resolvido via Gurobi. Enquanto Gurobi foi capaz de resolver apenas instâncias de pequeno porte, o algoritmo apresentado foi capaz de resolver todos os tamanhos de instância e até obter alguns ótimos globais.

PALAVRAS CHAVE. Localização e Roteamento, Veículos Elétricos, Metaheurísticas.

Tópicos: Logística e Transporte, Metaheurísticas.

ABSTRACT

Electric vehicles has become a trend in the last few years and a quick advancement on its technologies can be seen daily. As every rising technology, implementing a good and reliable infrastructure is a crucial step on its adoption. Optimization techniques can be used in order to reduce costs and make it more appealing to logistic companies to adhere to the use of this mean of transportation. In this paper we present a novel algorithm for the Multi-Depot Electric Location-Routing Problem with Time Windows, Battery Swapping and Partial Recharging. The proposed algorithm is composed by a Simulated Annealing metaheuristic with the local search conducted by a GRASP-VNS metaheuristic. We compared the algorithm's results to the mathematical model solved via Gurobi. While Gurobi was able to solve only small sized instances the presented algorithm was able to solve all instances sizes and even obtaining some global optima.

KEYWORDS. Location-Routing. Electric Vehicles. Metaheuristic.

Paper topics: Logistics and Transport, Metaheuristics.

1. Introduction

With the concerns on the limitations and disadvantages of the non-renewable sources of energy, such as petroleum, researches for alternative technologies independent on them has been growing and having an increase in popularity since the beginning of the years 2000 [Høyer, 2008]. One example of those alternative technologies is the electric vehicles that, although was invented in the beginning of the 19th century, only in recent years became more popular and an increasing presence of them in our lives can be seen across the world.

There are many reasons for the yet small market share of electric vehicles, despite the benefits they can bring to society over internal combustion ones according to Carley et al. [2013]. In one hand, there are the internal combustion vehicles that in order to be refueled, millions of gas stations are available in every part of the world. In the other hand, electric vehicles require a special power outlet to be charged and those are not common to be found, mainly in the countryside or even on medium sized cities. Other disadvantage of electric vehicles is the battery recharging time, which affects directly the vehicle autonomy. While the fastest recharge type is done using the quick charge outlet, that can provide up to 80% of battery capacity in 40 minutes, and internal combustion vehicle can be refueled in less than 5 minutes. As stated by Cecere et al. [2018], after the price, the vehicle's range is the most important factor in the decision individuals make to purchase an electric vehicle. Good charging infrastructures can affect directly the vehicles' range by providing alternatives ways to recharge batteries, hence a more efficient distribution of the infrastructure's components throughout the space is vital to the technology's prosperity.

The environmental aspect is also very important. The combustion of fossil fuels produces greenhouse gases (GHG) and they have a major role in the global warming. This happens due to the greenhouse effect, in which the GHGs help the atmosphere to absorb most of the radiation coming from the sun and causing a significantly increase in the global average temperature. The consequences of global warming are severe as stated by the Natural Resources Defense Council Denchak [2019] and can cause a huge environmental and economical loss for the entire world.

Many studies have shown the effects of internal combustion vehicles in the global emission of greenhouse gases in different countries. According to Zhao et al. [2019], EV deployment has a better long-term de-carbonization effect, while fuel economy regulations shows a better result in the short term. During the 2015 United Nations Climate Change Conference, China government agreed to achieve its emission peak of CO₂ before the year 2030. By adopting a deployment plan of electric vehicles together with fuel economy regulations, by the year 2026, China can reach its peak of gases emissions, reaching their goal. In Europe, the transportation sector is responsible for 26% of the total amount of CO₂ emissions, according to the Annual European Union greenhouse gas inventory [EEA, 2020], a 6% increase relatively to 2014. India's transportation sector is the third most GHG emitter in the country according to Singh et al. [2019], and due to the country's huge economical growth, CO₂ emissions by the transportation sector have an exponential growth rate.

For the best of our knowledge there are two, non-excluding, approaches to overcome the difficulties with the electric vehicle network infrastructure problems. The first one is to install a set of recharging stations in several points in the network. This allow vehicles to recharge batteries without spending lots of time and energy moving to a recharging station. In this regard a more effective option would be to install those stations in the places that the vehicles have to head to do some operation and use the operations' time to recharge partially the battery. The second approach is to install battery swap stations instead or with recharging stations. Swap stations can take up to 10 minutes to replace an electric vehicle battery for a fully charged one. They have this very good advantage, however they are much more expensive to be built and must be sited carefully.

Finding a good balance between swap and recharging stations location and the vehicles routes is a difficult task. This problem was modeled as a mixed integer linear programming model called Electric Location-Routing Problem (ELRP) in which many variants were studied involving many characteristics such as: time windows, different types of facility to sit, vehicle capacity, etc. The ELRP consists on a combinatory optimization problem in which a fleet of electrical vehicles must have their routes defined in a way that they will serve a set of customers alongside their travel. Also, one has to determine the locations where a set of facilities will be installed where the vehicles can recharge or replace their batteries so they can finish their routes. This is a combination of two others well known optimization problems, the facility location and vehicle routing problems. Due to their classification as a NP-Hard problem, the ELRP is also in this category and there is no known algorithm able to solve it in a polynomial time.

Our goal is to propose an algorithm for the Multi-Depot Electric Vehicle Location-Routing Problem with Time Windows, Battery Swapping and Partial Recharging and also apply our algorithm to large-sized real world data. This paper is organized as follows: in Section 2 we present a brief review on the literature related to electric vehicle infrastructures; in Section 3 we present the problem definition and a mathematical model for it; in Section 4 we present the developed algorithm; in Section 5 we present the test instances elaborated for the experiments with the model and algorithms and we also show the results obtained with the experiments; in Section 6 we discuss the results and present some insights for future works.

2. Literature review

An electric vehicle infrastructure can be composed by a range of elements going from battery recharging stations (BRS) to special road lanes that allow cars to have their batteries charged wirelessly while driving. However, most of the studies consider either the use of battery swap stations (BSS), BRS or a combination of both. A slightly different approach was made by Cui et al. [2018]. They considered a special type of BRS that is placed in the truck body, so they can move the station to another place according to the daily traffic flow. This vehicle is called mobile charging vehicle (MCV) and it is requested by customers via a mobile app and must be drove to the scheduled place. The MCV can also support multiple types of charges. The authors present a mathematical formulation for the problem based on the single-depot ELRP. The objective is to minimize the total travelled distance and determine the vehicles' route, the MCV's allocation and the type of charger in each station; while meeting the time-windows, battery charge and vehicle capacity constraints.

Li-Ying e Yuan-Bin [2015] elaborate a mixed integer linear programming model and an algorithm for the Multiple Charging Station Location-Routing Problem with Time Window of Electric Vehicle (EV-MCS-LRPTW). Besides the usual constraints and features of the ELRP like time windows, vehicle battery and vehicle loading capacities constraints, they also consider four types of charging stations: slow charging station (SCS), fast charging station (FCS), super-fast charging station (SFCS), and the BSS. The model's objective function is to minimize the total cost including the cost of siting stations, the cost of electricity to recharge batteries and the drivers wage. The designed algorithm is an Adaptive Variable Neighborhood Search mixed with the Tabu Search algorithm. The results show that the proposed algorithm can find near optimal solutions in the smaller sets of instances and provide convincing results in moderate run time on the largest sets. As future works, they pointed out to extend the problem to multiple depots, to consider a mixed fleet of electric and internal combustion vehicles and also to consider public charging stations.

Schiffer e Walther [2017] define a version of the ELRP in which time windows and partial recharging are considered. Time windows constraints are very common in practical applications and partial recharging allow the electric vehicle to use service time to recharge battery, increas-

ing its autonomy. The authors present a novel mixed integer programming model called Electrical Location-Routing Problem with Time Windows and Partial Recharging (ELRP-TWPR). As suggestion for future works the authors highlight the implementation of heuristic solution method in order to solve large instances for real world problem, the application of the proposed model in practical case studies and extend the model for heterogeneous or mixed fleet.

A mixed integer programming model for the ELRP with simultaneously BSS and BRS is presented by Paz et al. [2018]. The authors present, in fact, three models of which two are for the BSS and BRS individually and a third one for BSS and BRS mixed. The model's objective function is to minimize the total traveled distance. In their methodology they use a set of dummy nodes representing duplicated stations, so they can be visited multiple times, being one extra visit per dummy node set. As future works they suggested the design of solutions strategies for large-scale instances and application on real case data.

3. Problem definition

The problem is a variant of the ELRP called the Multi-Depot Electric Vehicle Location-Routing Problem with Time Windows, Battery Swapping and Partial Recharging and is defined as follow. A set of customers, each one with time window and demand parameters, must have their demand supplied by an electric vehicle. A set of vehicles must depart from a depot, perform a route delivering goods and return to the same initial depot. The vehicle has a limited autonomy based on its battery capacity and to address this problem two measures can be taken: (i) use the service time to perform a recharge in the customer's BRS or (ii) go to a BSS and have its battery swapped for a fully charged one. When at the customer, the vehicle can use the service time to give its battery a charge but shouldn't stay longer than that. Also, the places where the depots will be sited and the amount of it must be decided. To address the problem of combining a short-term and long-term problem we used a common approach for location-routing problems, that consists in amortizing the installation costs in a daily base, based on the expected use duration of the facility. For example, if the depot siting cost is one million dollars and it is expected to be used for ten years, the amortized cost will be \$273.97, this represents the daily cost of the depot.

The goal is to determine where to open the depots and sit battery swap stations, the routes the vehicles should travel delivering goods and determine a recharge plan by defining when and where the vehicle has to recharge or swap the battery; while minimizing the total cost composed by the BSS and BRS installation cost, the vehicle cost, the drivers wage, depot siting cost and energy cost.

3.1. Mathematical model

We used the linear programming model presented by Paz et al. [2018] as a base to our research, since they have already created a formulation that incorporate all characteristics we needed. However, we did some modifications, because the original model's objective function was to minimize the total travelled distance and we want to minimize the operational cost. To do so, we had to add a couple more decision variables and implement a few more constraints. The model is presented below with all the modifications done. Also, the original model allowed vehicles to recharge in BRS more time then the service time and allowed the vehicles to revisit BRS vertexes. We believe that in practical terms, this is not a viable option because the customers who provide BRS may have others goods to receive, hence the BRS might be occupied. So we did this modification, not allowing vehicles to revisit BRS and to recharge only during service time. The model is presented below:

Sets:

UD_0, UD_1	Vertices of depots for the dispatch and arrival of vehicles.
C	Customer vertices.
R	Special vertices where BSS can be sited.
SK_i	Dummy vertices associated with a vertex i .
Union sets	$UD = UD_0 \cup UD_1, V = C \cup R \cup SK_i, S = V \cup UD$ $V_0 = V \cup UD_0, V_1 = V \cup UD_1, V_{01} = V \cup UD_0, C_0 = C \cup UD_0$

Parameters:

id_i	Depot identification number, which is the same for both its dispatch and arrival vertices, $i \in UD$.
di_{ij}	Minimum distance between $i \in V_0$ and $j \in V_1$.
td_{ij}	Travel time between $i \in V_0$ and $j \in V_1, td_{ij} = di_{ij}/v, v$ is the vehicle speed.
s_i	Service time required at vertex $i \in C$.
e_i, l_i	Earliest and latest time, respectively, of the time window of vertex $i \in V_{01}$.
p_i	Demand of vertex $i \in V_0$.
D	Freight capacity of the vehicle.
Q	Battery capacity.
ct	Battery swapping time.
c	Consumption rate of energy per distance unit.
O	Maximum number of recharging points to locate.
M	Very large positive number.
r	Energy charging rate per unit of time.
dc, sc	Cost to site a single depot and a single BSS, respectively.
v	Vehicle speed.
vc	Vehicle cost.
dw	Driver's wage by distance unit driven.
rec, sec	Energy cost in BRS and BSS, respectively.

Decision variables:

x_{ij}	Binary, equal to 1 if a vehicle moves from $i \in V_0$ to $j \in V_1 i \neq j$, 0 otherwise.
y_i	Binary, equal to 1 if at vertex $i \in R$ a BSS is located, 0 otherwise.
z_i	Binary, equal to 1 if at vertex $i \in UD_0$ a depot is located, 0 otherwise.
t_i	Moment of time in which a vehicle arrives at the vertex $i \in V_{01}$.
d_i	Amount of freight left when the vehicle arrives at vertex $i \in V_{01}$.
q_i	Amount of energy left in the battery when the vehicle arrives at vertex $i \in V_{01}$.
u_i	Identification number of the depot from which vehicle serving $i \in V_1$ departs.
w_i	Amount of energy to be charged at vertex $i \in C$
v_i	Amount of energy to be charged at vertex $i \in R \cup SK_{h \in R}$

$$\min \sum_{i \in UD_0} z_i \cdot dc + \sum_{i \in R} y_i \cdot sc + \sum_{i \in UD_0} \sum_{j \in V_1 | i \neq j} x_{ij} \cdot vc + \sum_{i \in V_0} \sum_{j \in V_1 | i \neq j} x_{ij} \cdot di_{ij} \cdot dw + \sum_{i \in C \cup SK_c} rec \cdot w_i + \sum_{i \in R} sec \cdot v_i \quad (1)$$

$$\text{s. t.} \quad \sum_{j \in V_{+1} | i \neq j} x_{ij} = 1 \quad \forall i \in C \quad (2)$$

$$\sum_{j \in V_{+1} | i \neq j} x_{ij} \leq 1 \quad \forall i \in \{R \cup S\} \quad (3)$$

$$\sum_{i \in V_{+1} | i \neq j} x_{ji} = \sum_{i \in V_0 | i \neq j} x_{ij} \quad \forall j \in V \quad (4)$$

$$\sum_{i \in \{C \cup R\}} y_i \leq O \quad \forall i \in R \quad (5)$$

$$\sum_{j \in V_0 | i \neq j} x_{ji} \leq M \cdot y_i \quad \forall i \in R \quad (6)$$

$$\sum_{j \in V_0 | h \neq j} x_{jh} \leq M \cdot y_i \quad \forall i \in R, h \in SK_i \quad (7)$$

$$t_j \geq t_i + (td_{ij} + s_i) \cdot x_{ij} - M \cdot (1 - x_{ij}) \quad \forall i \in C_0, j \in V_1 | i \neq j \quad (8)$$

$$t_j \geq t_i + td_{ij} \cdot x_{ij} + ct \cdot x_{ij} - M \cdot (1 - x_{ij}) \quad \forall i \in \{R \cup SK_{h \in R}\}, j \in V_1 | i \neq j \quad (9)$$

$$t_j \geq t_i + td_{ij} \cdot x_{ij} + w_i/r - M \cdot (1 - x_{ij}) \quad \forall i \in \{C \cup SK_{h \in C}\}, j \in V_1 | i \neq j \quad (10)$$

$$e_i \leq t_i \leq l_i \quad \forall i \in V_{01} \quad (11)$$

$$q_j \leq Q - c \cdot di_{ij} \cdot x_{ij} + M \cdot (1 - x_{ij}) \quad \forall i \in \{R \cup SK_{h \in R}\}, j \in V_1 | i \neq j \quad (12)$$

$$q_j \leq q_i - c \cdot di_{ij} \cdot x_{ij} + M \cdot (1 - x_{ij}) \quad \forall i \in UD_0, j \in V_1 | i \neq j \quad (13)$$

$$q_j \leq q_i + w_i - c \cdot di_{ij} \cdot x_{ij} + M \cdot (1 - x_{ij}) \quad \forall i \in \{C \cup SK_{h \in C}\}, j \in V_1 | i \neq j \quad (14)$$

$$q_i + w_i \leq Q \quad \forall i \in \{C \cup SK_{h \in C}\} \quad (15)$$

$$q_i \leq Q \quad \forall i \in UD_0 \quad (16)$$

$$d_j \leq d_i - p_i \cdot x_{ij} + D \cdot (1 - x_{ij}) \quad \forall i \in V_0, j \in V_1 | i \neq j \quad (17)$$

$$d_i \leq D \quad \forall i \in UD_0 \quad (18)$$

$$u_j \geq x_{ij} \cdot id_i - M \cdot (1 - x_{ij}) \quad \forall i \in UD_0, j \in V \quad (19)$$

$$u_j \leq x_{ij} \cdot id_i + M \cdot (1 - x_{ij}) \quad \forall i \in UD_0, j \in V \quad (20)$$

$$u_j \geq u_i - M \cdot (1 - x_{ij}) \quad \forall i \in V, j \in V_1 | i \neq j \quad (21)$$

$$u_j \leq u_i + M \cdot (1 - x_{ij}) \quad \forall i \in V, j \in V_1 | i \neq j \quad (22)$$

$$u_j = id_j \quad \forall j \in UD_1 \quad (23)$$

$$w_i \leq r \cdot s_i \cdot y_i \quad \forall i \in C \quad (24)$$

$$w_h \leq r \cdot s_i \cdot y_i \quad \forall i \in C, h \in SK_i \quad (25)$$

$$x_{ij} = 0 \quad \forall i \in V \setminus \{S\}, j \in SK_i | i \neq j \quad (26)$$

$$x_{ji} = 0 \quad \forall j \in V \setminus \{S\}, i \in SK_j | i \neq j \quad (27)$$

$$\sum_{j \in V_1} x_{ij} \leq |V_1| \cdot z_i \quad \forall i \in UD_0 \quad (28)$$

$$v_i = Q \cdot \sum_{j \in V_1} x_{ij} - q_i \cdot \sum_{j \in V_1} x_{ij} \quad \forall i \in R \cup SK_R \quad (29)$$

$$x_{ij} \in \{0; 1\} \quad \forall i \in V_0, j \in V_1 | i \neq j \quad (30)$$

$$y_i \in \{0; 1\} \quad \forall i \in R \quad (31)$$

$$z_i \in \{0; 1\} \quad \forall i \in UD_0 \quad (32)$$

$$t_i, d_i, q_i \in \mathbb{R}^+ \quad \forall i \in V_{01} \quad (33)$$

$$u_i \in \mathbb{R}^+ \quad \forall i \in V_1 \quad (34)$$

$$w_i \in \mathbb{R}^+ \quad \forall i \in C \quad (35)$$

$$v_i \in \mathbb{R}^+ \quad \forall i \in R \cup SK_{h \in R} \quad (36)$$

The Objective Function 1 represents the total cost of the electric vehicle infrastructure and is divided in 6 parcels: depot installation cost, BSS installation cost, vehicle cost, drivers wage, energy cost in the BRS and BSS. The Constraints 2, 3 and 4 impose the continuity of the routes.

Constraints 5 imposes a limit on the number of BSS to be allocated. Constraints 6 and 7 ensure that a route will only use sited BSS. Constraints 8, 9, 10 and 11 guarantee the time windows being respected. Constraints 12, 13, 14, 15 and 16 control the vehicles' battery level. Constraints 17 and 18 control the vehicle load. Constraints 19, 20, 21, 22 and 23 impose that each route start and finish in the same depot. Constraints 24 and 25 guarantee that the vehicle will be partially charged during the service time. Constraints 26 and 27 block the vehicle to move from a node to a dummy node associated to it. Constraints 28 compute the amount of depots used. Constraints 29 control over the amount of battery that will be recharged in the BSS after the vehicle swap batteries, this constraint set is quadratic but can be easily linearized. Constraints 30, 31, 32, 33, 34, 35 and 36 ensure the decision variables' integrity.

4. Algorithms

In order to address our problem to medium and large sized instances we developed a heuristic based on the Simulated Annealing (SA) algorithm with a local search guided by a Greedy Randomized Adaptive Search (GRASP) and Variable Neighborhood Search (VNS) algorithms. We now describe the solution representation, the neighborhood structures and the algorithm itself.

4.1. Solution representation

Two solution representations are used in our proposed algorithm. First, the solution is represented by a permutation of customers and then, in local search level, the solution is represented in a more complete way, by having all customers designated to routes, depots opened, BSS sited and recharging plan computed. We refer by this representation as the complete solution representation, and to the first one as permutation representation. Figure 1 illustrates the customers permutation $\{11, 10, 5, 8, 6, 7, 9\}$ mapped into the complete solution where two routes are defined. The vertex 0 and 12 correspond to the depot, where 0 is the departure vertex and 12 is the arrival vertex; and vertexes 1, 2 and 3 are the BSSs. Each component in this representation have a set of parameters composed by: vehicle battery level, vehicle load, arrival time and quantity of energy recharged.

0	11	2	10	5	1	8	6	12
0	7	3	9	12				

Figure 1: The permutation $\{11, 10, 5, 8, 6, 7, 9\}$ transformed into the complete solution.

The customers permutation is mapped into the complete solution by a greed algorithm, divided in three steps. First, we build what we call pseudo routes, they are routes only with customers where no BSS nor recharging plan are set. To do so, we create the first empty route and the customers are sequentially inserted until the vehicle capacity or time windows constraints are violated. We considered time windows constraints at this point just to further minimize the number of time windows violated to be addressed in further steps. If there is any customers not placed, we repeat this procedure with a new empty route and the procedure goes until all customers have a designated route. During the pseudo route generation, in order to be able to compute the parameters; such as arrival time, we choose a temporary depot as being the closest one from the first node that will be inserted. The next two procedures are responsible to optimize the depots sited (*addDepots*) and to generate the recharging plan by siting BSSs and defining where the vehicle must recharge or swap battery (*addStations*).

The *addDepots* procedure is responsible for, with all routes pre-generated, to determine a better depot combination by considering the first and last customer of each route. The procedure works in a greed manner where sum of deviation for each depot is computed from the route's

first and last customers and chose the depots with the minimum deviations sum. At this point, at maximum, one depot is chosen for the entire solution, this depot will supply all the customer, but since we don't have the recharging plan, we don't know yet if the chosen depot will be feasible for each route, this problem is addressed further.

The *addStations* procedure, illustrated by the Algorithm 1, is responsible for the recharging plan. To do so, the procedure computes sequentially all route's parameters (arrival time, battery level, vehicle load, etc), until it finds an unfeasibility point that can be caused by time window or battery level. At this stage the vehicle uses every BRS available to recharge the battery in a greed manner, once reaching a BSS we get the the amount of battery remaining and subtract from previous BRS recharging. Unfeasibilities due to time windows constraints are treated by removing the customer from the current route and pushing it to the end of the next route; if the current route is the last, a new route is created to fit the customer. If the unfeasibility is caused by the battery level we have to determine where to install a BSS so the vehicle can keep travelling. To chose where to install the BSS, the procedure trace back the route looking for the first viable BSS, this search is done by looking to every arc and fitting the minimal detour BSS until reaching a previous sited BSS or the depot. This procedure can be implemented in a greed manner where is chosen the first viable interval with the minimum deviation BSS, or in a semi-greed manner, creating a candidate list of intervals and randomly picking one to receive the BSS.

Once the permutation mapping algorithm is finished, we have a solution with all parameters and recharging plan computed. This is the complete representation that is further used in a local search algorithm.

Algorithm 1: addStations procedure

Input : A set of routes RT
Output: The complete representation (a set of routes RT with the recharge plan)
for route r in RT **do**
 for vertex v in r **do**
 Compute v parameters;
 if v is unfeasible to time windows **then**
 Remove v from r ;
 Insert v in $r + 1$;
 if v is unfeasible to battery level **then**
 Choose a route interval to receive a BSS;
 Insert BSS with minimal detour in the interval;
 Set the inserted BSS as the current vertex;

4.2. Neighborhood structures

As we used two solution representations, we developed two sets of neighborhood structures. One for the permutation representation, used in the SA algorithm and another for the routes representation, used by the local search. Next we describe the structures used.

4.2.1. Permutation neighborhood structures

Two structures are used in the permutation domain, they are the Partial Inversion and the Customer Swap. The Partial Inversion takes part of the permutation and invert its sequence, its implementation is exactly to the classic 2opt neighborhood. The Customer Swap just swap the position of two customers in the permutation.

4.2.2. Complete solution neighborhood structures

In the complete solution domain, 5 structures are used: 2opt, Route Split, Route Junction, Shift Customer and BSS Replacement. Those neighborhood structures, excluding the BSS Replacement, are a subset of structures used by Hof et al. [2017] to solve a similar problem to electric vehicle infrastructures, we just added the BSS Replacement to fulfil the multi-depot characteristic in our current problem. The 2opt is a neighborhood structure widely used in many routing problem and consists in swap two arcs in the route, making a route segment inverted. The Route Split structure splits a route in two smaller ones, while the Route Junction structure takes two routes and combine in one, if feasible. Shift Customer simply shift, forward or backward, a customer by a given number of positions inside the route. The BSS Replacement structure selects a BSS being used in the solution and replaces it for another BSS that is also already being used.

4.3. Local search

The local search is conducted by a GRASP algorithm and is based on the complete representation. In order to generate the greed randomised solutions, we use a slightly different version of the permutation mapping algorithm, where we modify the *addStations* procedure to add the battery swap stations based on a restricted list of candidates. The restricted list of candidates is generated during the step where the algorithm trace back the route to find a proper place the install a BSS. Instead of choosing the first viable place to sit a BSS, the list is generated by adding all intervals where the leaving vertex has a battery level up to a threshold of 30 % of total battery level left, and then a random member of the list is chosen and inserted in the route. By doing so, we allow the vehicle to travel to a BSS a little before it would by the greed criteria and the solutions become more diverse.

The GRASP's local search is conducted by a Basic Variable Neighborhood Search Algorithm. This step is crucial, due to the limitations on the permutation mapping algorithm, as it is not able to generate every solution in the solution space. The VNS also deal with the complete solution representation and is composed by a shake stage and a local search procedure. In the shake stage, the solution suffers a modification in order to escape of the local optima and it has various levels of intensity that are defined by the use of different neighborhood structures. All structures defined in the Section 4.2.2 are used in this step, in a random order. The local search is conducted by a Hill-Descent algorithm with fist improvement.

4.4. Metaheuristic

The proposed algorithm is based on the Simulated Annealing metaheuristic. The core element of this algorithm, is the allowance of movement to worsen solutions and, by doing so, the algorithm is capable to escape of local optima solutions and have more chances of approximation of the global optima. In order to control the algorithm convergence, the SA set a temperature variable that as the algorithm runs, decreases and worsening solutions will be accepted according to a probability: the lower is the temperature, the lower is this probability.

The initial solution is generated by the permutation mapping algorithm that is applied to a customers permutation sorted by the due date in ascending order. Next, the algorithm continues iteratively as follow: a neighborhood solution is generated with the Partial Inversion neighborhood structure, a local search is done and the new solution is evaluated and accepted or not according to the current temperature. In the local search step, two algorithm are used, depending on the current temperature level. The option for this strategy is due to the higher GRASP algorithm complexity, that makes this local search be way costly, and also helps to avoid a premature algorithm convergence. At high levels, when the solution are still subject to many worsening movements, a simple Hill-Descent search is used with the Customers Swap structure. At low levels, when the algorithm is approaching the end, the GRASP algorithm is used as local search.

5. Experiments

In this section, results are shown for the mathematical model and the proposed algorithm. Also, in order to demonstrate the effectiveness of our local search approach, we show the results for the SA executing only the Hill-Descent algorithm. The test instances and how they were generated is also shown. The experiments were executed in a machine equipped with a processor Intel(R) Core(TM) i7-7700HQ and 8GB RAM. The model was implemented using the solver Gurobi 9.0 and the maximum runtime was set to 7200 seconds, in which the solver finishes its execution if the gap 0% was not reached until the maximum runtime. The algorithm was implemented in C++ using the Microsoft Visual Studio compiler and compiled in release mode. Even though the model has one quadratic set of constraints, Gurobi was able to solve it without the linearization. The algorithm was executed only once for each instance.

5.1. Instances

The instance set introduced by Demir et al. [2012] for the Pollution-Routing Problem is used as a base to ours. This instance set is composed by a set of cities from the United Kingdom, each one with a given demand, due date, service time and ready time. In order for them to meet all the requirements to be used with our problem, we did some adaptations, such as definition of the vertices that can be used to install the battery swap stations, definition of new depot vertices and definition of parameters related to electric vehicles, BRS and BSS. The instances also contain a distance matrix with real world distances, the vehicle load capacity and speed parameters. Since the original instance set contains more than 170 instances, we decide to use just a few of them and we selected 3 instances of each size.

To define the set of vertices candidates to have a BSS, we developed a maximum coverage location problem mathematical model and the electric vehicles, BSS and BRS parameters values are the same used by Li-Ying e Yuan-Bin [2015]. Our goal with the model was to obtain a more spread out distribution of vertices to avoid some areas to become isolated, due to a long distance between them and the nearest BSS.

5.2. Results

The Table 1 displays the results obtained by the solver and the algorithms. Column 'cost' displays the objective function results as defined by the total cost of the electric vehicle infrastructure, the column 'time' displays the total runtime in seconds and the column 'gap' displays the final gap found by the solver. The SA column refers to the SA algorithm without the GRASP local search. Cells with '-' means that the model was not able to find a solution for the instance.

As we analyze the results in Table 1, we can notice that the solver was able to find only one optimal solution within the time limit set and for most of the instances with 25 customers or more, no solution was found. The SA was able to find one optimal solution, in the instance UK10.01 and match the objective function in the instances UK10.20 and UK15.20. For all instances the solver was not able to find a solution, the SA was able to do so. Comparing results of the SA with the GRASP local search and the SA with the Hill-Descent local search, the benefits of using the second solution representation in the local search is notable by an improvement in the objective function value. Also, it is notable the increase in the final runtime by using another metaheuristic in the local search step. However, we believe the runtime of two hours is reasonable due to the electric vehicle infrastructures long term characteristics.

6. Conclusions

In this paper, we presented a novel algorithm for the Multi-Depot Electric Location-Routing Problem with Time Windows, Battery Swapping and Partial Recharging, that was first introduced by Paz et al. [2018]. The solution is represented by a permutation of customers that

Table 1: Results for the model and algorithms execution.

Instances	Model			SA		SA + GRASP	
	cost	gap (%)	time (S)	cost	time (S)	cost	time (S)
UK10_01	6373.1	0.00	1053	6383.0	0	6373.1	33
UK10_11	6742.0	0.00	2	6790.9	0	6746.1	3
UK10_20	6339.0	93.75	7200	6339.0	0	6339.0	8
UK15_01	6563.2	0.81	7200	6747.5	0	6613.9	171
UK15_11	6528.2	93.98	7200	6664.6	0	6530.6	171
UK15_20	6365.5	99.25	7200	6375.3	1	6365.5	109
UK20_01	6581.2	97.04	7200	6847.6	0	6711.8	302
UK20_11	6594.9	96.80	7200	6802.0	0	6639.5	188
UK20_20	6565.6	98.38	7200	6703.7	1	6629.3	347
UK25_01	6566.8	98.76	7200	6792.1	1	6577.2	1062
UK25_11	6462.7	98.71	7200	6518.4	0	6478.1	488
UK25_20	-	-	7200	6498.0	1	6471.5	912
UK50_01	13408.4	98.84	7200	7662.1	8	6987.9	6503
UK50_11	-	-	7200	7687.7	8	7147.5	7201
UK50_20	-	-	7200	7757.8	13	7265.5	5437
UK75_01	-	-	7200	9242.2	33	8191.8	7201
UK75_11	-	-	7200	7610.9	34	7197.0	7201
UK75_20	-	-	7200	8709.7	38	7998.1	7201
UK100_01	-	-	7200	10051.6	95	8790.5	7201
UK100_11	-	-	7200	9913.5	131	8512.7	7201
UK100_20	-	-	7200	10684.1	104	8882.0	7201
UK150_01	-	-	7200	10509.3	431	9204.2	7201
UK150_11	-	-	7200	12379.6	380	10332.5	7201
UK150_20	-	-	7200	12566.8	382	10415.9	7201
UK200_01	-	-	7200	12512.3	533	10239.4	7201
UK200_11	-	-	7200	12749.0	454	10075.1	7201
UK200_20	-	-	7200	12956.9	467	11376.9	7201

represents the order that each one must be added in the solution. Also, we defined a second solution representation that is composed by the complete solution, that is the permutation after the processing, in which allow us to perform a more precise local search and improve our results. The proposed algorithm was based on the Simulated Annealing metaheuristic and two local search algorithms were used. At higher temperatures a simple Hill-Descent local search is conducted to optimize the runtime and to avoid premature convergence and, at lower temperature, a GRASP algorithm conduct the local search. Also, in the GRASP algorithm, A VNS is used to perform its local search using the complete solution representation to address the permutation's limitation that is not able to represent the entire solution space.

In the result section, we showed the results obtained by the execution of the model, implemented with the solver Gurobi, and the SA algorithm in two variants. The proposed SA algorithm with GRASP local search has been proven to be better then the SA with the simple hill-descent local search in terms of solution cost found and better then the solver because the solver was not able to find solutions in the large-sized instances. As future works, we highlight the strengthen of the formulation, in order to improve the gap and the use of different algorithms to perform the SA local search, in order to improve computational time. Also is we highlight the exploration of other ways to deal with the short-term versus long-term characteristic of location-routing problem, rather then amortizing costs.

Acknowledgments: The authors would like to thanks to *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES)* and *Fundação de Amparo à Pesquisa do Estado de Minas*

Gerais (FAPEMIG) for the support in this project.

References

- Carley, S., Krause, R. M., Lane, B. W., e Graham, J. D. (2013). Intent to purchase a plug-in electric vehicle: A survey of early impressions in large US cities. *Transportation Research Part D: Transport and Environment*, 18(1):39–45. ISSN 13619209.
- Cecere, G., Corrocher, N., e Guerzoni, M. (2018). Price or performance? A probabilistic choice analysis of the intention to buy electric vehicles in European countries. *Energy Policy*, 118 (March):19–32. ISSN 03014215.
- Cui, S., Zhao, H., e Zhang, C. (2018). Multiple types of plug-in charging facilities' location-routing problem with time windows for mobile charging vehicles. *Sustainability (Switzerland)*, 10(8): 2855. ISSN 20711050.
- Demir, E., Bektas, T., e Laporte, G. (2012). An adaptive large neighborhood search heuristic for the Pollution-Routing Problem. *European Journal of Operational Research*, 223(2):346–359. ISSN 03772217.
- Denchak, M. (2019). Are the effects of global warming really that bad? URL <https://www.nrdc.org/stories/are-effects-global-warming-really-bad>. Accessed: 2020-05-21.
- EEA (2020). Annual European Union greenhouse gas inventory 1990–2018 and inventory report 2020. Technical Report May, European Environment Agency, Brussels.
- Hof, J., Schneider, M., e Goeke, D. (2017). Solving the battery swap station location-routing problem with capacitated electric vehicles using an AVNS algorithm for vehicle-routing problems with intermediate stops. *Transportation Research Part B: Methodological*, 97:102–112. ISSN 01912615.
- Høyer, K. G. (2008). The history of alternative fuels in transportation: The case of electric and hybrid cars. *Utilities Policy*, 16(2):63–71. ISSN 09571787.
- Li-Ying, W. e Yuan-Bin, S. (2015). Multiple charging station location-routing problem with time window of electric vehicle. *Journal of Engineering Science and Technology Review*, 8(5):190–201. ISSN 17912377.
- Paz, J. C., Granada-Echeverri, M., e Escobar, J. W. (2018). The multi-depot electric vehicle location routing problem with time windows. *International Journal of Industrial Engineering Computations*, 9(1):123–136. ISSN 19232934.
- Schiffer, M. e Walther, G. (2017). The electric location routing problem with time windows and partial recharging. *European Journal of Operational Research*, 260(3):995–1013. ISSN 0377-2217.
- Singh, N., Mishra, T., e Banerjee, R. (2019). Greenhouse Gas Emissions in India's Road Transport Sector. *Climate Change Signals and Response*, p. 197–209.
- Zhao, F., Liu, F., Liu, Z., e Hao, H. (2019). The correlated impacts of fuel consumption improvements and vehicle electrification on vehicle greenhouse gas emissions in China. *Journal of Cleaner Production*, 207(2019):702–716. ISSN 09596526.