

The Electric Capacitated Covering Tour Problem

Lucas Porto Maziero

Instituto de Computação - Universidade Estadual de Campinas
Av. Albert Einstein 1251, 13083-852, Campinas, SP
lucas.maziero@ic.unicamp.br

Rafael Kendy Arakaki

Instituto de Computação - Universidade Estadual de Campinas
Av. Albert Einstein 1251, 13083-852, Campinas, SP
rafael.arakaki@ic.unicamp.br

Matheus Diógenes Andrade

Instituto de Computação - Universidade Estadual de Campinas
Av. Albert Einstein 1251, 13083-852, Campinas, SP
matheusdiogenesandrade@gmail.com

Fábio Luiz Usberti

Instituto de Computação - Universidade Estadual de Campinas
Av. Albert Einstein 1251, 13083-852, Campinas, SP
fusberti@ic.unicamp.br

ABSTRACT

This paper proposes a new problem based on the classic Vehicle Routing Problem (VRP). The problem is called Electric Capacitated Covering Tour Problem (ECCTP) and model a situation where customers can be serviced remotely by covering and only electric vehicles are employed. The ECCTP integrates two research areas: the covering problems and the green logistics. The problem is modeled by Mixed Integer Linear Program (MILP) and solved by a constructive heuristic. A set of benchmark instances is proposed and computational experiments were conducted.

KEYWORDS. Electric capacitated covering tour problem; Green vehicle routing problem; Green logistics;

1. Introduction

The Vehicle Routing Problem (VRP) is a problem where a set of customers must be serviced by a heterogeneous fleet of vehicles with limited capacity. In this problem, all customers need to be visited and serviced. The objective is to minimize the total cost of vehicle routes. Many variants of the VRP were studied in the literature [Golden, 1975].

In other research direction, green logistics have emerged like a discipline focused on the planning and operation of logistic systems with high energy efficiency and low levels of carbon emission [Leggieri e Haouari, 2017].

Among VRP variants dealing with green logistics, there is the Green Vehicle Routing Problem (G-VRP). The G-VRP has additional challenges associated with operating a fleet of Alternative Fuel Vehicles (AFV) and incorporating stops at Alternative Fuel Stations (AFS) [Erdoğan e Miller-Hooks, 2012]. One of the biggest challenges modeled by G-VRP is the reduced number of alternative fuel stations and the low autonomy of alternative fuel vehicles.

VRP variants model applications where additional constraints related to scheduling, budget, fleet heterogeneity and, others are considered. [Allahyari et al., 2015] studies a VRP variant where some customers may be located in difficult access regions and therefore they can be serviced by covering of neighboring customers. In this context, the Capacitated Covering Tour Problem (CCTP) was proposed. In the CCTP, a customer can be serviced by any vehicle which transits inside a pre-established neighborhood radius of that customer.

The present work aims to link aspects of vehicle routing by covering and green logistics. The Electric Capacitated Covering Tour Problem (ECCTP) is a vehicle routing problem where a homogeneous fleet of battery electric vehicles are used to service the demands of customers by covering.

We introduce a new VRP variant called ECCTP and presented a mathematical formulation for the problem. A heuristic method is proposed for ECCTP. Furthermore, benchmark instances for the ECCTP were created based on a set of public benchmark instances originally proposed for the VRP. Computational experiments were conducted on benchmark instances.

Section 2 describes some literature papers related to this work. In Section 3 the ECCTP is formally defined using a mathematical formulation for the problem. Section 4 describes the constructive heuristic solution method proposed for the ECCTP. Section 5 contains the details and result of computational experiments. Finally, Section 6 presents the paper conclusion.

2. Literature review

The Covering Salesman Problem (CSP), proposed by [Current e Schilling, 1989], addresses the following question: considering a set of sites scattered in the plane that must be covered by a single vehicle tour and knowing that each site covers some of its neighbors, what is the minimum length of an enclosed vehicle tour in which all sites are covered? More formally, given an undirected graph, the CSP objective is to find the shortest Hamiltonian cycle on a subset of vertices that covers the graph. The special case where each vertex covers strictly itself is the Travelling Salesman Problem (TSP) [Applegate et al., 2007], which follows that CSP is also NP-hard.

Several variants of CSP were also investigated. [Gendreau et al., 1997] studied the Covering Tour Problem (CTP), a variant of CSP. Let $G = (V \cup W, E)$ be an undirected graph, where $V \cup W$ is the set of vertices and E is the set of edges. Vertex v_0 is the depot, V is the set of vertices that can be visited, $T \subseteq V$ is the set of vertices that must be visited ($v_0 \in T$), and W is the set of vertices that must be covered but cannot be visited. The goal of the CTP is to determine a minimum length tour that visits a subset of vertices $S \subseteq V$ such that $T \subseteq S$ and each vertex of W is covered

by some vertex in S . The authors proposed heuristics and a branch-and-cut algorithm to solve the CTP.

There are works in the literature that tackle the geometric version of CSP. In this version, a compact region of the plane containing each vertex is specified as a neighborhood set. The goal is to find a minimum length tour that starts from a depot and intercepts all neighborhood sets, thus covering all its corresponding vertices. Approximation algorithms, heuristics, and methodologies based on Integer Linear Programming (ILP) were developed for this version ([Dumitrescu e Mitchell, 2003], [Gulczynski et al., 2006], [Dong et al., 2007], [Shuttleworth et al., 2008], [Behdani e Smith, 2014], [Coutinho et al., 2016]).

Some works in the literature address variants of CTP considering multiple vehicles. [Hachicha et al., 2000] introduced the Multi-Vehicle Covering Routing Problem m -CTP. Given a graph $G = (V \cup W, E)$, where $V \cup W$ is the set of vertices and E the set of edges. Vertex v_0 is a depot at which are based m identical vehicles, V is the set of vertices that can be visited, $T \subseteq V$ is the set of vertices that must be visited ($v_0 \in T$) and W is the set of vertices that must be covered but cannot be visited. The objective of m -CTP is to determine m routes of minimum total length satisfying following restrictions:

- there are at most m vehicle routes and each route starts and ends at vertex v_0 ;
- each vertex of T is visited exactly once, while each vertex of $V \setminus T$ is visited at most once;
- each vertex of W must be covered by a route in the sense that it must lie within a preset distance c of a vertex of V belonging to a route (assuming that v_0 does not cover all vertices of W);
- the number of vertices (excluding v_0) in each route is limited by value p ;
- the length of each route cannot exceed a value of q .

In the work of [Naji-Azimi et al., 2012] was studied an application relating to the location of distribution centers, which provides humanitarian aid to affected people throughout a disaster area. In these situations, support teams are unable to visit each affected area. Therefore, people need to go to a particular distribution center to obtain survival items, provided that these centers are not too far away. The locations of the distribution centers are defined for a car fleet departing from a depot. This application can be modeled through m -CTP. The authors proposed an ILP formulation and developed a heuristic able to produce good solutions, even for large size instances.

[Hachicha et al., 2000] developed an ILP formulation and a branch-and-cut algorithm for a special case of m -CTP, where the restrictions of the length of the routes are relaxed, i.e. $q = +\infty$. The authors also developed a genetic algorithm that provides heuristic solutions for m -CTP. [Ha et al., 2013] addressed this same special case of m -CTP. The authors developed a branch-and-cut algorithm and a metaheuristic based on the evolutionary local search method. The computational experiments were performed on a set of instances derived from the TSPLIB library and showed the efficiency of the proposed methods. [Kammoun et al., 2017] used the Variable Neighborhood Search (VNS) metaheuristic based on the Variable Neighborhood Descent (VND) method for solving m -CTP. The computational experiments showed that the VNS metaheuristic outperformed the results obtained by [Ha et al., 2013].

[Erdoğan e Miller-Hooks, 2012] propose the G-VRP and, develop two constructive heuristics, the modified heuristic of Clarke and Wright savings and, the algorithm of clustering based on density, and propose a custom improvement technique.

Erdoğan e Miller-Hooks [2012] execute both proposed algorithms and the exact method on two instance sets, a group of small instances which was randomly generated, and the other group of instances originated from a textile supply company located at Virginia in the EU.

[Çağrı Koç e Karaoglan, 2016] introduce a new efficient approach to solve the G-VRP, this approach uses the Simulated Annealing heuristic based on the Branch and Cut algorithm, which combines many valid inequalities derived from the literature to improve the lower bound. That approach uses the Simulated Annealing heuristic to improve the initial solution and find upper bounds during the search process. [Çağrı Koç e Karaoglan, 2016] evaluate the approach in terms of the number of test instances solved optimally and the computational time used to find the best solution. The computational results have shown that 22 of 40 instances with 20 customers was solved optimally into a pre-defined time limit.

In the approach of [Çağrı Koç e Karaoglan, 2016], a new mathematical formulation with fewer variables and restrictions, and without network augment, is proposed for the problem. [Çağrı Koç e Karaoglan, 2016] make an analysis of the formulation proposed by [Erdoğan e Miller-Hooks, 2012], and conclude: “The G-VRP formulation proposed by [Erdoğan e Miller-Hooks, 2012] require the network augment due to the dummy vertices used to represent multiple visits to an AFS. In the worst case, the number of dummy vertices for each AFS can be equals to the number of customers, what increase the total number of vertices in the network from $n + s + 1$ to $n + ns + 1$. Although this is acceptable for small instances, the method requires much time to achieve a solution for big instances, once the number of binary variables depends on the number of dummy vertices. Knowing that, the enumeration tree size in any MILP solver grow up exponentially based in the number of binary variables of a problem and, the memory requirement of any software grow up with the number of variables and restrictions, then is expected that the proposed formulation with network augments has an worst outcome than the formulation without network augmentation, in terms of Linear Programming (LP) relaxation, Central Process Unit (CPU), and etc.”.

The new formulation of [Çağrı Koç e Karaoglan, 2016] counts with the restrictions of capacity and sub-cycle elimination proposed by [Miller et al., 1960] to the Traveling Salesman Problem (TSP) and adapted by [Kulkarni e Bhavne, 1985] to the Capacitated VRP (CVRP).

3. Problem Description

Consider a complete graph $G(V \cup W \cup F \cup \{v_0\}, E)$, where V , W and F are three disjoint sets of vertices: V are traffic vertices; W are customer vertices; and F are recharge stations vertices. The $v_0 \in V$ is the depot vertex and E is the set of edges. The edges $(i, j) \in E$ have costs $c_{ij} \geq 0$. In the vertex depot v_0 , there is a set of M homogeneous vehicles with capacity Q and energy autonomy of β . For each vertex $w \in W$ is associated a demand $d_w \geq 0$. For each vertex $v \in (V \cup W) \setminus \{v_0\}$, $D(v)$ is the set of vertices that are covered by v such that $D(v) \cap F = \emptyset$ and for each vertex $w \in W$, $C(w)$ is the set of vertices that cover w . It is considered that $w \in C(w)$, $\forall w \in W$. The vehicles can visit vertices of $V \cup W \cup F$ and all vertices of W must be served by coverage or visit. The goal of ECCTP consists of defining a set of up to M minimum length routes satisfying the following constraints:

- Each route begins and ends at the vertex depot v_0 ;
- Each vertex $v \in (V \cup W)$ is visited at most by one route;
- Each vertex $f \in F$ can be visited by several routes;
- Each vertex $w \in W$ must be covered by at least one route, i.e., it must be serviced through some vertex $v \in C(w)$ that is visited by some route;

- The demand d_w of a vertex $w \in W$ is served exclusively by one route;
- The demand of a vertex $w \in W$ can be served by a route k if there is a vertex v visited by route k such that $v \in C(w)$ or w is visited by route k ;
- Given the set of vertices $R \subseteq (V \cup W)$ visited by a route k , the total demand of this route is given by the sum of the demands of the vertices served by route k ;
- The total demand of the route must not exceed the capacity Q of the vehicle;
- The energy autonomy β of the vehicle decreases in c_{ij} when it travels along the edge $(i, j) \in E$. It is not possible for the vehicle to visit an edge whose cost is higher than its energy autonomy. When the vehicle visits a vertex recharge station in F , its energy autonomy is restored to the maximum of β units.

Bellow follows the variables used in the MILP formulation for the ECCTP:

- x_{ij} = number of times edge $(v_i, v_j) \in E$ is visited in the solution.
- $y_i^k = \begin{cases} 1, & \text{if vehicle } k \in M \text{ visit } v_i \in V \cup W \cup \{v_0\} \\ 0, & \text{otherwise} \end{cases}$
- w_f = number of visits in $v_f \in F$;
- $z_i^k = \begin{cases} 1, & \text{if vehicle } k \in M \text{ attend the demand of } v_i \in W \\ 0, & \text{otherwise} \end{cases}$
- e_j = vehicle energy level in $v_j \in V \cup W \cup F \cup \{v_0\}$;
- p = number of non-empty routes in solution ($p \leq M$).

Bellow follows the functions and constants used in the MILP formulation for the ECCTP:

- $\delta^+(i)$ = directed edges which *leave* from vertex $v_i \in V \cup W \cup F \cup \{v_0\}$;
- $\delta^-(i)$ = directed edges which *enter* in a vertex $v_i \in V \cup W \cup F \cup \{v_0\}$;
- $\delta(i) = \delta^+(i) \cup \delta^-(i)$;
- $M_{LB} = \lceil \sum_{i \in W} d_i / Q \rceil$ is a lower bound for a feasible number of vehicles;
- M = maximum number of vehicles allowed.

The objective function (1) minimize the routes total cost, the constraints (2) and (3) implies that the depot must belong to all non-empty routes and limit the number of routes. Constraints (4) guarantee that customer or traffic nodes are visited by at most one vehicle. The constraints (5) ensure that every vertex $v_i \in W$ must have its demand attended, and the constraints (6) implies that for every vertex $v_i \in (V \cup W) \setminus \{v_0\}$ visited by a route k , there are two incident edges at this vertex. The constraints (7) implies that the number of times which a recharge station is visited must be equals to the number of entering edges at the station and to the number of leaving edges at the station. The constraints (8) and (9) ensure that an edge $(v_i, v_j) \notin \delta(v_0)$ belongs to a route k

if the vertices v_i and v_j belongs to the route k . The constraints (10) ensure that, for every vertex $v_r \in W$ whose demand d_r is attend by vehicle $k \in M$, exists at least one vertex $v_i \in (v_r)$ visited by the same vehicle k , where $C(v_r)$ is the neighborhood of v_r including the v_r . The constraints (11), ensure that the total attended demand by a vehicle must not exceed its capacity D . The constraints (12), is the subcycle restriction, which assures solution connectivity. More specifically, it state that given a set of nodes S without depot node v_0 , summing the node degrees of all nodes in S ($v_i \in S$), but excluding one node (v_i^*), the number of edges must be less than enough to form a cycle in S that is disconnected of v_0 . The constraints (13) restore the vehicle energy level when vehicle visiting a depot or a recharge station. The constraints (14) updates the vehicle energy level at each $v \in V \cup W$ visited. The constraints (15) does not allow vehicles to go to stations or depot without the required energy. The constraints (16) define the minimum and maximum energy level. The constraints (17-23) define the variables domain.

$$\text{Min } \sum_{(i,j) \in E} c_{ij} x_{ij} \quad (1)$$

s.a.

$$\sum_{k \in M} y_{v_0}^k = p \quad (2)$$

$$M_{LB} \leq p \leq M \quad (3)$$

$$\sum_{k \in M} y_i^k \leq 1 \quad \forall v_i \in (V \cup W) \quad (4)$$

$$\sum_{k \in M} z_i^k = 1 \quad \forall v_i \in W \quad (5)$$

$$\sum_{j: (i,j) \in \delta^+(i)} x_{ij} = \sum_{j: (j,i) \in \delta^-(i)} x_{ji} = \sum_{k \in M} y_i^k \quad \forall v_i \in V \cup W \cup \{v_0\} \quad (6)$$

$$\sum_{j: (f,j) \in \delta^+(f)} x_{fj} = \sum_{j: (j,f) \in \delta^-(f)} x_{jf} = w_f \quad \forall v_f \in F \quad (7)$$

$$y_i^k \geq x_{ij} + y_j^k - 1 \quad \forall (v_i, v_j) \in E : v_i, v_j \in (V \cup W), k \in M \quad (8)$$

$$y_j^k \geq x_{ij} + y_i^k - 1 \quad \forall (v_i, v_j) \in E : v_i, v_j \in (V \cup W), k \in M \quad (9)$$

$$\sum_{i \in C(v_r)} y_i^k \geq z_r^k \quad \forall v_r \in W, k \in M \quad (10)$$

$$\sum_{i \in W} d_i z_i^k \leq Q \quad \forall k \in M \quad (11)$$

$$\sum_{v_i \in S} \sum_{v_j \in S} x_{ij} \leq \sum_{k \in M} \sum_{v_i \in S \setminus (\{v_i^*\} \cup F)} y_i^k + \sum_{v_i \in S \cap F} w_i$$

$$\forall S \subseteq (V \cup W \cup F) \setminus \{v_0\}, \forall v_i^* \in S \cap (V \cup W) \quad (12)$$

$$e_f = \beta \quad \forall v_f \in F \cup \{v_0\} \quad (13)$$

$$e_j \leq e_i - c_{ij}x_{ij} + \beta(1 - x_{ij}) \quad \forall v_j \in V \cup W, \forall v_i \in V \cup W \cup F \cup \{v_0\} \quad (14)$$

$$c_{ij}x_{ij} \leq e_i \quad \forall v_i \in V \cup W \cup F \cup \{v_0\}, \forall v_j \in F \cup \{v_0\} \quad (15)$$

$$0 \leq e_i \leq \beta \quad \forall v_i \in V \cup W \quad (16)$$

$$x_{ij} \in \{0, 1\} \quad \forall (v_i, v_j) \in E : v_i \in V \cup W \text{ or } v_j \in V \cup W \quad (17)$$

$$x_{ij} \in \mathbb{Z}^+ \quad \forall (v_i, v_j) \in E : v_i, v_j \in F \cup \{v_0\} \quad (18)$$

$$y_i^k \in \{0, 1\} \quad \forall v_i \in V \cup W, \forall k \in M \quad (19)$$

$$w_f \in \mathbb{Z}^+ \quad \forall v_f \in F \quad (20)$$

$$z_i^k \in \{0, 1\} \quad \forall v_i \in W, \forall k \in M \quad (21)$$

$$e_j \in \mathbb{R}^+ \quad \forall v_j \in V \cup W \cup F \cup \{v_0\} \quad (22)$$

$$p \in \mathbb{Z}^+ \quad (23)$$

4. Constructive heuristic

The Constructive Heuristic (CH) proposed for ECCTP that is a greedy heuristic that guarantees to construct a feasible solution if there is any feasible solution for the instance. The constructive heuristic is composed of a initialization phase followed by a construct phase. CH is deterministic unless there are ties in the order given by the structures initialized in initialization routine.

4.1. Initialization procedure

The initialization phase of the CH is composed of computing four data structures that will be used throughout the algorithm:

- $R(u)$ for each $u \in (V \cup W)$ is an ordered (by distance, non-decreasing) list of nodes closest to the node u .
- $C(u)$ for each $u \in W$ is a list of customer and traffic nodes that cover node u .
- F_u for each $u \in (V \cup W)$ is the closest recharge station $F_u \in F$ to node u .
- $SP_{(u,v)}$ for each $u, v \in F$ contains a minimum cost feasible path from u to v using only recharge stations as intermediary nodes. This path is feasible regarding vehicle energy autonomy when traveling between nodes. The entry $SP_{(u,v)}$ contains only one path.

Initializing this data is important from the aspect of computational efficiency, since these data structures need to be initialized only once and then can be used throughout the algorithm for any number of iterations of the constructive heuristic.

4.2. Heuristic pseudocode

The constructive heuristic is given by Algorithm 1. The CH is a greedy heuristic that is guaranteed to construct a feasible solution if there is any feasible solution for the instance. CH is a deterministic procedure unless there are ties in the order given by the data structures computed in initialization routine (Section 4.1).

Algorithm 1 starts by initializing an empty route with the vehicle at the depot node. In a greedy and iterative fashion, it searches for the closest customer (w) that has not yet been serviced and that demand does not exceed the residual vehicle capacity (rv_c). After selecting the customer w to be serviced the CH heuristically decides if the customer will be visited or serviced by covering through a traffic node.

Data: Graph $G(V \cup W \cup F, E)$; cost function c ; demand function d ; covering functions C, D ; depot node v_0 ; parameters of vehicle demand capacity Q and energy capacity β .

Result: Heuristic solution sol for input instance.

```

1 Initialize structures  $R, C, F$  and  $SP$ ; // initialization phase (Section 4.1)
2  $sol \leftarrow \emptyset$ ;  $route \leftarrow \emptyset$ ;  $u \leftarrow v_0$ ;  $rvc \leftarrow Q$ ;  $ea \leftarrow \beta$ ;
3 while Exists a customer  $w \in W$  not serviced do
4     Search closest  $w \in W$  in order of  $R[u]$  such that  $w$  is not serviced yet and  $d_w \leq rvc$ ;
5     if  $w \neq \emptyset$  then
6         if  $ea < \beta$  and  $ea < c_{uw} + c_{w,F_w}$  then
7              $route \leftarrow route \cup \{F[u]\}$ ; // vehicle goes to closest station and is recharged
8              $ea \leftarrow \beta$ ;  $u \leftarrow F[u]$ ;
9         end
10        else if  $ea \geq c_{uw} + c_{w,F_w}$  then
11            Search closest traffic node  $v \in V, v \in C[w]$  such that  $v$  is not visited yet;
12            if  $v$  is found and  $c_{uv} \leq c_{uw}$  and  $ea \geq c_{uv} + c_{v,F_v}$  then
13                // vehicle goes to  $v$  and service  $w$  by distance
14                 $route \leftarrow route \cup \{v\}$ ;  $ea \leftarrow ea - c_{uv}$ ;
15                 $u \leftarrow v$ ;  $rvc \leftarrow rvc - d_w$ ; //  $v$  is visited;  $w$  is serviced.
16            end
17            else
18                // vehicle goes to  $w$  and service it
19                 $route \leftarrow route \cup \{w\}$ ;  $ea \leftarrow ea - c_{uw}$ ;
20                 $u \leftarrow w$ ;  $rvc \leftarrow rvc - d_w$ ; //  $w$  is serviced.
21            end
22        end
23        else
24            // vehicle goes to  $w$  using intermediary recharge stations and service it
25             $route \leftarrow route \cup \{SP_{(u,F[w])}\} \cup \{w\}$ ;  $ea \leftarrow \beta - c_{F[w]w}$ ;
26             $u \leftarrow w$ ;  $rvc \leftarrow rvc - d_w$ ; //  $w$  is serviced.
27        end
28    end
29    else
30        if  $ea \geq c_{uv_0}$  then
31             $route \leftarrow route \cup \{v_0\}$ ;
32        end
33        else
34             $route \leftarrow route \cup \{F[u]\} \cup \{SP_{(F[u],F[v_0])}\} \cup \{v_0\}$ ;
35        end
36         $sol \leftarrow sol \cup route$ ; // vehicle goes to depot and route is finished
37         $u \leftarrow v_0$ ;  $rvc \leftarrow Q$ ;  $ea \leftarrow \beta$ ;  $route \leftarrow \emptyset$ ;
38    end
39 end
40 return  $sol$ 

```

Algorithm 1: Constructive Heuristic (CH) for ECCTP.

The constructed solution is feasible because at each step the heuristic does not visit a node $v \in V \cup W$ such that it will be without sufficient energy (ea) to then recharge to the station ($F[v]$) closest to node v . Also, when the vehicle is in low energy it visits the closest recharge station. If necessary to visit a long distance node w the vehicle uses the feasible shortest path $SP_{(u,w)}$ containing only stations as intermediary nodes. The vehicle returns to the depot when its capacity is not sufficient to service any customer. If there is still demand remaining then a new vehicle is employed and the process is repeated. Finally, the set of routes constructed (feasible solution) is the algorithm output.

5. Computational Experiments

The computational experiments for ECCTP aim to generate a set of benchmark instances for the problem and solve these instances using the proposed heuristic method. Because ECCTP is a new problem, we created a set of benchmark instances that is adapted from a set of benchmarking instances from CVRP (Section 5.1). Since no methods have been published to compare our methods with, the aim of the experiments is to verify how the performance and cost of the proposed method solutions are compared against each other with varying parameters of demand capacity, vehicle energy capacity, covering distances threshold and instance sizes.

The proposed method was implemented in C++ language and compiled by g++ v. 5.4.1 with “-O2” option on an Intel i7-6700k 4.0 GHz processor with 16 GB of RAM and Linux 64-bit operating system.

5.1. Benchmark of Instances

The set of instances for the ECCTP was generated from the set of instances for the CVRP created in the work of [Uchoa et al., 2017]. The CVRP instances used to generate the instances for ECCTP were $X-n101-k25$, $X-n200-k36$, $X-n303-k21$, $X-n401-k29$, $X-n502-k39$, $X-n613-k62$, $X-n701-k44$, $X-n801-k40$, $X-n916-k207$ and $X-n1001-k43$. These instances are named in the format $X-nA-kB$, where A represents the $n + 1$ vertices (including the depot vertex) and B the minimum number of vehicles needed to serve the demand for all vertices, calculated by the Bin Packing Problem.

To generate ECCTP instances, four parameters were used:

- $|W|$: number of vertices with demand;
- $|F|$: number of vertices which are recharge station;
- β : vehicle energy autonomy;
- c : distance of coverage.

The $|W|$ parameter varied in 40% and 60% of n , while the $|F|$ parameter varied in 10% and 15% of n . The vertices with demand and the vertices that are recharge station were chosen randomly, with uniform distribution, from the set of vertices of the instance of the CVRP. It is worth noting that a vertex of the ECCTP instance cannot have demand and be a recharge station, simultaneously. The $n - |W| - |F|$ remaining vertices are considered traffic vertices.

Consider d_G the graph diameter of the CVRP instance. The β parameter varied in 40%, 60%, and 80% of d_G . The vertices with demand are always within a maximum distance $\frac{\beta}{2}$ from some vertex recharge station. In addition, there is at least one vertex recharge station that is within a maximum distance β from the vertex depot and every vertex recharge station is within a maximum distance β from another vertex recharge station. Therefore, the vehicles can travel through all the recharging stations and all vertices with demand can be served.

For each vertex $v \in (V \cup W) \setminus \{v_0\}$, the set $D(v)$, which represents the vertices covered by v , is formed by all traffic and customers vertices that are within a maximum distance c from v . The c parameter varied in 100 and 200.

For each instance of the CVRP used to generate the instances for the ECCTP, all possible combinations of the $|W|$, $|F|$, β and c parameters were considered, resulting in 24 instances generated for the ECCTP. In all, 240 instances for the ECCTP were generated. It is worth noting that the depot vertex v_0 and the capacity Q of the vehicles of the ECCTP instances are the same as that of the CVRP instances.

The instances for the ECCTP are named in the format $X-nA-eaD-fE-wH-cI$, where A represents the $n + 1$ vertices (including the depot vertex), D represents the energy autonomy of the vehicle, E the number of vertices that are recharge station, H the number of vertices that have demand and I the distance of coverage.

5.2. Experiment results

Table 1 shows the experiment results for the proposed heuristic method. The column “Instance name” presents the instance name. The column “Solution Cost” is the upper bound achieved with the heuristic and the column CPU Time is the time spent by the heuristic. The heuristic method was run for all benchmark of instances, but Table 1 shows the results for only 24 instances.

Instance name	Solution Cost	CPU Time
n101-ea1012-f10-w40-c200	33978	0
n101-ea506-f10-w40-c200	48724	0
n200-ea1003-f19-w119-c100	66972	0
n200-ea1003-f19-w119-c200	61107	0.01
n303-ea454-f30-w120-c100	23076	0.01
n303-ea454-f30-w120-c200	20005	0.01
n401-ea1109-f40-w160-c100	72688	0.02
n401-ea1109-f40-w160-c200	69014	0.02
n502-ea482-f50-w200-c100	76093	0.03
n502-ea482-f50-w200-c200	74945	0.03
n613-ea1120-f61-w244-c100	80437	0.05
n613-ea840-f91-w367-c200	57407	0.05
n701-ea1098-f105-w280-c100	82774	0.07
n701-ea1098-f105-w280-c200	71419	0.07
n801-ea1113-f120-w320-c100	61589	0.09
n801-ea1113-f120-w320-c200	54696	0.09
n916-ea1108-f137-w366-c100	360154	0.12
n916-ea1108-f137-w366-c200	341981	0.12
n1001-ea1088-f100-w600-c100	69799	0.11
n1001-ea1088-f150-w600-c100	68507	0.12

Tabela 1: Results of Computational Experiments.

From the results in Table 1 one can observe that instances with increased values of covering threshold c displays for substantially lower solution costs (comparing 3rd and 4th lines, or 6th and 7th). In the other hand, the impact of having a greater number of recharge stations (can be observed comparing the last two lines) shows that having more recharge stations allowed for a better solution cost but the difference is not as big as for the parameter c .

The difference of energy capacity is also interesting (comparing 1st and 2nd lines), where the method obtained a solution with 30% less cost when the vehicle energy capacity is doubled. This can be useful for example to understand how meaningful is the impact of having electric cars with better batteries.

Regarding computational processing time, it is observable that for all instances the processing time is very short. However it increases substantially as the number of nodes are increased, for example for $n = 200$ the CPU time is ≈ 0.01 and for $n = 1001$ it is increased to ≈ 0.12 (an approximate increase of 12 times in processing time for a 5 times number of nodes growth).

6. Conclusion

The ECCTP joins two different NP-hard combinatorial optimization problems, the problem of vehicle routing with covering and the electric vehicles routing employed by Green VRP. Our work proposed a benchmark of instances and a constructive heuristic for the ECCTP.

The computational experiments performed with the constructive heuristic show that the method obtained feasible solutions for the proposed benchmark of instances in very short computational time. Also, the parameters of covering functions and energy capacity were shown to have a substantial impact on the solution costs produced by the proposed heuristic.

Future studies for the ECCTP consist of implementation of the exact model presented in this work. The development of a metaheuristic for the problem is also an interesting research topic.

Referências

- Allahyari, S., Salari, M., e Vigo, D. (2015). A hybrid metaheuristic algorithm for the multi-depot covering tour vehicle routing problem. *European Journal of Operational Research*, 242(3):756–768.
- Applegate, D. L., Bixby, R. E., Chvatal, V., e Cook, W. J. (2007). *The Traveling Salesman Problem: A Computational Study (Princeton Series in Applied Mathematics)*. Princeton University Press, Princeton, NJ, USA. ISBN 0691129932, 9780691129938.
- Behdani, B. e Smith, J. C. (2014). An integer-programming-based approach to the close-enough traveling salesman problem. *INFORMS Journal on Computing*, 26(3):415–432.
- Coutinho, W. P., Nascimento, R. Q. d., Pessoa, A. A., e Subramanian, A. (2016). A branch-and-bound algorithm for the close-enough traveling salesman problem. *INFORMS Journal on Computing*, 28(4):752–765.
- Current, J. R. e Schilling, D. A. (1989). The covering salesman problem. *Transportation science*, 23(3):208–213.
- Dong, J., Yang, N., e Chen, M. (2007). Heuristic approaches for a tsp variant: The automatic meter reading shortest tour problem. In *Extending the Horizons: Advances in Computing, Optimization, and Decision Technologies*, p. 145–163. Springer.
- Dumitrescu, A. e Mitchell, J. S. (2003). Approximation algorithms for tsp with neighborhoods in the plane. *Journal of Algorithms*, 48(1):135–159.
- Erdoğan, S. e Miller-Hooks, E. (2012). A green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 48(1):100–114. ISSN 1366-5545. URL <http://www.sciencedirect.com/science/article/pii/S1366554511001062>. Select Papers from the 19th International Symposium on Transportation and Traffic Theory.
- Gendreau, M., Laporte, G., e Semet, F. (1997). The covering tour problem. *Operations Research*, 45(4):568–576.
- Golden, B. L. (1975). Vehicle routing problems. Technical report, M.I.T. Operations Research Center, Cambridge. URL <http://www.dtic.mil/dtic/tr/fulltext/u2/a013639.pdf>.
- Gulczynski, D. J., Heath, J. W., e Price, C. C. (2006). The close enough traveling salesman problem: A discussion of several heuristics. In *Perspectives in Operations Research*, p. 271–283. Springer.

- Ha, M. H., Bostel, N., Langevin, A., e Rousseau, L.-M. (2013). An exact algorithm and a metaheuristic for the multi-vehicle covering tour problem with a constraint on the number of vertices. *European Journal of Operational Research*, 226(2):211–220.
- Hachicha, M., Hodgson, M. J., Laporte, G., e Semet, F. (2000). Heuristics for the multi-vehicle covering tour problem. *Computers & Operations Research*, 27(1):29–42.
- Kammoun, M., Derbel, H., Ratli, M., e Jarboui, B. (2017). An integration of mixed vnd and vns: the case of the multivehicle covering tour problem. *International Transactions in Operational Research*, 24(3):663–679.
- Kulkarni, R. e Bhawe, P. (1985). Integer programming formulations of vehicle routing problems. *European Journal of Operational Research*, 20(1):58 – 67. ISSN 0377-2217. URL <http://www.sciencedirect.com/science/article/pii/037722178590284X>.
- Leggieri, V. e Haouari, M. (2017). A practical solution approach for the green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 104:97 – 112. ISSN 1366-5545. URL <http://www.sciencedirect.com/science/article/pii/S1366554517302880>.
- Miller, C. E., Tucker, A. W., e Zemlin, R. A. (1960). Integer programming formulation of traveling salesman problems. *J. ACM*, 7(4):326–329. ISSN 0004-5411. URL <http://doi.acm.org/10.1145/321043.321046>.
- Naji-Azimi, Z., Renaud, J., Ruiz, A., e Salari, M. (2012). A covering tour approach to the location of satellite distribution centers to supply humanitarian aid. *European Journal of Operational Research*, 222(3):596–605.
- Shuttleworth, R., Golden, B. L., Smith, S., e Wasil, E. (2008). Advances in meter reading: Heuristic solution of the close enough traveling salesman problem over a street network. In *The Vehicle Routing Problem: Latest Advances and New Challenges*, p. 487–501. Springer.
- Uchoa, E., Pecin, D., Pessoa, A., Poggi, M., Vidal, T., e Subramanian, A. (2017). New benchmark instances for the capacitated vehicle routing problem. *European Journal of Operational Research*, 257(3):845–858.
- Çağrı Koç e Karaoglan, I. (2016). The green vehicle routing problem: A heuristic based exact solution approach. *Applied Soft Computing*, 39:154 – 164. ISSN 1568-4946. URL <http://www.sciencedirect.com/science/article/pii/S1568494615007085>.