More efficient formulations and valid inequalities for the Green Vehicle Routing Problem

M. Bruglieri[a], S. Mancini[b,*], O. Pisacane[c]

[a]*Dipartimento di Design, Politecnico di Milano*
[b]*Dipartimento di Matematica e Informatica, Universitá di Cagliari*
[c]*Dipartimento di Ingegneria dell'Informazione, Universitá Politecnica delle Marche*

Abstract

The Green Vehicle Routing Problem (G-VRP) aims to efficiently route a fleet of Alternative Fuel Vehicles, based at a common depot, in order to serve a set of customers, minimizing the total travel distance. Because of the limited driving range of these vehicles, intermediate stops at the Alternative Fuel Stations must also be considered. For the G-VRP, we propose two Mixed Integer Linear Programming formulations allowing multiple visits to the stations without introducing dummy copies of them. In the first model, only one visit to a station between two customers or between a customer and the depot is allowed. While, in the second model, two consecutive visits to stations are also permitted. In addition, the two formulations are strengthened through both dominance criteria to a-priori identify the stations that are more efficient to use in each route and valid inequalities, specifically tailored for the G-VRP. Computational results, carried out on benchmark instances, show that our formulations strongly outperform the exact solution approaches presented in the literature. Finally, in order to better investigate the issue of the consecutive refueling stops, a new set of instances is properly generated and significant transport insights are also provided.

Keywords: Alternative Fuel Vehicle, Consecutive Refueling, Efficient Alternative Fuel Station, Valid Inequalities

[*] Corresponding author: simona.mancini@unica.it, simona.mancini@polito.it

1. Introduction

The efficient optimization of the transport activities plays a key role for guaranteeing eco-sustainable solutions. Indeed, the transport sector significantly contributes to generate harmful greenhouse gases (e.g., carbon dioxide CO_2, methane and nitrous oxide). According to a recent study, for example, only in 2010, the transport sector produced greenhouse emissions for 7.0 gigatonnes CO_2 equivalent (Edenhofer et al., 2014). Moreover, all the harmful pollutants emitted have significant effects on the air quality, health and climate (Kontovas and Psaraftis, 2016). In particular, the road transportation represents one of the biggest producer of greenhouse gases and specifically, the road freight transportation contributes for about the 50% (Chapman, 2007). Consequently, many public and private transport agencies start using the Alternative Fuel Vehicles (AFVs) that can guarantee not only a reduction of the harmful emissions but also the respect of recent rules imposed by the European Commission (for more details on the EVs' characteristics, readers are referred to Chapter 6 of (Davis et al., 2015)).

However, the AFVs have a limited driving range and therefore, some stops at the Alternative Fuel Stations (AFSs) may be needed during a trip. On the other hand, the AFSs are currently not widespread across the road network and therefore, such stops should be also a priori planned in order to avoid drivers ending up being stuck along their routes. This happens when an AFV has not enough fuel to reach even the closest AFS.

It is also worth noting that, during a trip, the stops at the AFSs contributes to increase the total distance traveled by an AFV due to the detours for reaching the AFSs. The problem of planning the routes of a fleet of AFVs to serve a set of customers has been introduced in the literature by the seminal work (Erdoğan and Miller-Hooks, 2012) under the name of Green Vehicle Routing Problem (G-VRP). Every AFV leaves fully refueled a common depot and must return to it at the end of its route without exceeding a maximum allowed duration. In order to consider the limited driving range of the AFVs, during each route, stops at AFSs are also planned where the tank is assumed to be always fully replenished. The G-VRP aims to minimize the total travel distance, by assuming a fixed refueling time at the AFSs that does not depend on the remaining fuel in the AFVs' tank. Moreover, the fuel consumption of an AFV is assumed linearly proportional to the traveled distance.

In the literature, the G-VRP is represented on a complete and directed graph in which the set of nodes includes the customers, the AFSs and the depot. In order to allow visiting AFSs more than once, dummy copies of them are introduced to model each route as an elementary cycle and to avoid that routes may share AFSs nodes. The presence of these dummy nodes increases the number of binary variables in the Mixed Integer Linear Programming (MILP) formulation of the problem making its solution harder. This approach is used in most of the papers dealing with the G-VRP.

A first attempt to get rid of dummy nodes is made in (Koç and Karaoglan, 2016) where a three-index formulation of the G-VRP is introduced, based on binary variables y_{ijk} to discriminate if AFS k is used to go from customer i to customer j. However, this formulation does not correctly model the G-VRP. Indeed, the optimal solution of the G-VRP may require an AFV to visit two or more AFSs consecutively. An example of this kind is shown in Figure 1, with two customers, represented by nodes 1 and 2, respectively, two AFSs, s_1 and s_2, and the depot, node 0. The numbers besides the edges indicate the distances in miles between their endpoints. Assuming a tank capacity equal to 60 gallons and a fuel consumption rate of 0.2 gallons per mile as in the instances of (Erdoğan and Miller-Hooks, 2012) (i.e., a fully refueled AFV can travel 300 miles), this instance also satisfies their hypothesis that each customer is reachable by an AFV leaving the depot and returning to it with at most one visit to the AFS. For this instance, the optimal solution of the G-VRP consists in the single route $0, 1, s_1, s_2, 2, 0$ with a total traveled distance of 512 miles. If consecutive visits to AFSs are not allowed, the optimal solution would worsen since it would be made up of the two routes $0, 1, 0$ and $0, 2, 0$ with a total traveled distance of 600 miles.

It is worth noting that this kind of solution can be detected through the original formulation of (Erdoğan and Miller-Hooks, 2012) (since the graph considered is complete) but not by that proposed in (Koç and Karaoglan, 2016). Moreover, the possibility of visiting two or more AFSs consecutively has been also considered in the heuristic solution approaches proposed by (Felipe et al., 2014; Montoya et al., 2016). Indeed, there are several real situations where this may happen: for instance, when a product (e.g., truffle in Alba, near Turin, in Italy, fine wines, etc.) can be produced only in a specific place and must be delivered by AFVs to customers that are all concentrated in another specific place (e.g., the luxury restaurants of the Côte d'Azur, in France) which is so far from the former that more than one refueling along the travel is required.

Place Figure 1 about here

Our paper explicitly focuses on the G-VRP with multiple and consecutive visits to AFSs, without introducing dummy copies of them. More specifically, we propose a first MILP formulation of the G-VRP in which only one visit to AFS is allowed between two customers or between a customer and the depot or vice versa. Moreover, we introduce a second MILP formulation to model the possibility of having two consecutive visits to AFSs. Such a model is proven to be general since, under mild conditions, at most two consecutive visits are shown to be necessary in an optimal solution. Moreover, to further decrease the number of binary variables of the model, we design a method for a priori identifying an efficient set of AFSs, i.e., a set of AFSs that may be actually used in an optimal solution, between either each pair of customers or a customer and the depot (or vice versa). We strengthen the formulation by means of valid inequalities. We also discuss sufficient conditions to ensure that no more than two stops at the AFSs between two customers are required in an optimal solution. We test our model on the benchmark instances of (Erdoğan and Miller-Hooks, 2012) showing that it outperforms all other known exact approaches for the G-VRP, even the Branch and Cut method proposed in (Koç and Karaoglan, 2016). We introduce a new set of instances where consecutive refueling stops at AFSs are actually required in the optimal solutions. Numerical results, carried out on this new set, show that our model outperforms all other solution approaches. Finally, we analyze the optimal solutions in terms of some significant transport indicators (e.g., the percentage value of the detour for refueling respect to the distance traveled for serving the customers) to deduce some insights on the use of AFVs.

The rest of this paper is organized as follows. Section 2 reviews the main contributions on the G-VRP, proposed in the literature. Section 3 yields the statement of the problem together with its assumptions. Sections 4 and 5 describe the new MILP formulations of the G-VRP and the algorithms to compute the efficient set of AFSs for the cases of one visit and of two consecutive visits to them, respectively. Section 6 introduces some valid inequalities to strengthen both the models. Section 7 shows the numerical results on the benchmark instances and yields some comparisons with the best known results in the literature. Finally, Section 8 draws some conclusions and remarks some future research directions worthy of investigation.

2. Literature review

The problem of efficiently routing a fleet of vehicles represents a class of optimization problems widely investigated in the literature and known as *Vehicle Routing Problem* (VRP) (see (Toth and Vigo, 2014) for a recent survey on the topic). In the VRPs, the fleet has to serve a set of customers, geographically distributed and it is usually based at a common depot which every vehicle has to leave from and which has to return to. The route traveled by a vehicle handles a subset of customers and respects some constraints, among which the limited cargo capacity (*Capacitated VRP*- CVRP) and the Time Window within which each customer has to be served (CVRPTW). An assumption, common to a great variety of VRPs, is that the vehicle fuel autonomy is sufficient to reach the customers in every route, i.e., each vehicle is characterized by a unlimited driving range.

As already remarked in Section 1, with the aim of reducing the harmful emissions, the variant of VRP with both AFVs and AFSs, i.e., the G-VRP, is introduced in (Erdoğan and Miller-Hooks, 2012), where a MILP formulation is proposed together with a set of benchmark instances (small/medium size up to 20 customers and large size up to 500 customers). The increasing interest in such a problem is proven by a significant number of works in which it has been addressed, as shown in the recent survey (Bektaş et al., 2016). The problem of reducing the harmful emissions is tackled in (Eglese and Bektas, 2014), considering several factors among which the vehicle speed optimization and/or the use of AFVs. Indeed, the G-VRP finds significant real world applications in several fields, e.g., the *Reverse Logistics* and the *Green Logistics* (Sbihi and Eglese, 2007).

Recently, several solution methodologies have been proposed for the G-VRP. For example, in (Felipe et al., 2014), different heuristic approaches have been designed for the VRP with a fleet of electric vehicles. Together with the route planning, both the amount of the energy recharged and the recharging technology used at each station are decision variables.

In (Koç and Karaoglan, 2016), a Branch and Cut approach has been combined with the Simulated Annealing (SA) framework showing that more than half of small/medium-sized benchmark instances of (Erdoğan and Miller-Hooks, 2012) can be optimally solved in reasonable time. However, as remarked in Section 1, this formulation could provide sub-optimal solutions for some instances, since it neglects the possibility of two consecutive refuelings.

A SA based algorithm, together with a mathematical model, is also proposed in (Ćirović et al., 2014) where the fleet is distinguished into environmentally friendly and unfriendly vehicles. Both the approaches are tested on a network simulating the traffic conditions in Belgrade.

Other metaheuristics to solve the G-VRP are also proposed in the following papers. A two-step approach has been developed in (Montoya et al., 2016), where a set of routes is firstly found through a randomized route-first cluster-second heuristic combined with an optimal AFSs insertion procedure and, then, the solution is assembled through a set partitioning model. An ant-based algorithm combined with a fuel consumption model has been built in (Jabbarpour et al., 2015). A tabu search algorithm taking into account costs due to the fuel, the emissions and the vehicle usage, has been designed in (Zhang et al., 2015). Recently, in (Madankumar and Rajendran, 2018), a G-VRP arising in a supply chain for semiconductors has been addressed modeling also specific constraints on the compatibility among the products, the vehicle capacity, the request priorities and the start/completion time, with the aim of minimizing both the total routing cost and the refueling cost.

The G-VRP has been also extended to a bi-objective variant, minimizing both the total travel distance and the CO_2 emissions. For example, in (Jemai et al., 2012), an NSGA-II evolutionary algorithm is proposed for this variant while, in (Tiwari and Chang, 2015), a block recombination approach, assuming the truck load of a vehicle as key factor for reducing the harmful emissions, is presented.

The time-dependent version of the G-VRP has been addressed in ((Qian and Eglese, 2016), (Xiao and Konak, 2016) and (Sharafi and Bashiri, 2016)). In particular, (Qian and Eglese, 2016) considers a G-VRP where the speed along an arc is treated as a decision variable. A column generation based tabu search algorithm is developed and tested on real traffic data taken from the London road network. While, (Xiao and Konak, 2016) assumes a heterogeneous fleet and that the fuel consumption depends on several factors, among which, the payload weights and the travel speeds varying according to the traffic congestion. The problem is solved through an algorithm that hybridizes the mathematical programming with an iterative neighborhood search. A new version of the G-VRP in which the driver's dissatisfaction to perform a tour is modeled through a tour-driver penalty has been introduced in (Sharafi and Bashiri, 2016) and solved with a genetic algorithm.

Finally, the Hybrid VRP, i.e., a particular version of the G-VRP in which vehicles may switch at anytime between alternative and traditional fuel propulsion, has been presented in (Mancini, 2017). For this purpose, the unitary distance cost is much higher for the traditional fuel mode. However, in certain cases, it results more convenient to cover a short distance with traditional fuel than to make long detours to reach the nearest refueling station. This situation frequently occurs in cases in which refueling stations are not widespread across the territory. The author proposes an effective matheuristic based on large neighborhoods implicit exploration.

3. Statement of the problem and assumptions

The G-VRP is defined on a directed complete graph $G = (N, A)$, where the node set N contains both the set of customers I and the depot 0 while the arc set A is given by $A = \{(i, j) : i \in N, j \in N,\ i \neq j\}$. For each customer $i \in I$, the parameter p_i indicates the service time. The set of available AFSs is denoted by F and, $\forall s \in F$, the parameter p_s indicates the refueling time. For each pair (i, j) with $i \in N \cup F$, $j \in N \cup F, i \neq j$, the travel time, t_{ij}, and the travel distance, d_{ij}, to go from i to j, are given. For each AFV of the fleet, the maximum refueling capacity is Q and the fuel consumption rate is r. The vehicle speed v is assumed to be constant and a priori known.

The G-VRP aims to find a set of at most m AFV routes, with minimum total distance, such that: 1) every route starts and ends from/to the depot; 2) every customer i is visited, for a time equal to p_i, by one and only one AFV; 3) each AFV can stop at every AFS s for a full refueling spending a time p_s; 4) the maximum drive range Q/r is never exceeded, after a refuel; 5) the time duration of each route is at most equal to a given quantity T_{max}.

Place Table 1 about here

In Table 1, we summarize all the input parameters of the G-VRP.

In (Erdoğan and Miller-Hooks, 2012), the authors assume that each customer is reachable by an AFV leaving the depot and returning to it with at most one visit to the AFS. Under this hypothesis, an optimal solution may require two consecutive visits to the AFSs, as shown in the example of Fig.1. Moreover, although not directly stated in (Erdoğan and Miller-Hooks, 2012), we verified that their instances satisfy also the property that from every AFS, it is always possible to reach any other AFS. Under this second hypothesis, no more than two consecutive visits to the AFSs can be required in an optimal solution. Indeed, if by contradiction, the optimal solution should visit a sequence of AFSs, $s_1, s_2, \ldots, s_k$, we could replace such a sequence with the pairs of AFSs s_1, s_k obtaining certainly a feasible solution (since s_k is reachable directly from s_1) with a better or equal objective function value (given by a lower or equal travel distance).

In the next two sections, we introduce the mathematical programming formulations for the cases where at most one and at most two consecutive visits to the AFSs can be possible in the optimal solution, respectively.

4. Mathematical Programming Formulation with one visit to AFS

In this section, we describe the Mixed Integer Linear Programming (MILP) formulation for the case where no more than one visit to the AFS is necessary between two customers (as implicitly assumed in (Koç and Karaoglan, 2016)).

The proposed MILP model is based on the computation of the sets $L_{ij}, \forall (i, j) \in A$, of all the feasible AFSs that may be convenient to visit, going from i to j. In Algorithm 1, the steps for computing these sets are described.

Algorithm 1 Computation of the sets L_{ij}

```
Input: sets $A, F$, parameters $Q$, $r$, $d_{ij}$ and $t_{ij}$ $\forall (i, j) \in A$
Output: sets $L_{ij}$;
for $(i, j) \in A$ do
    $L_{ij} := F$;
    $s^* := \arg\min_{s \in F: d_{is} \leq \frac{Q}{r}, d_{sj} \leq \frac{Q}{r}} \{d_{is} + d_{sj}\}$;
    for $s \in F$ do
        if ($d_{is} \geq d_{is^*}$ and $d_{sj} \geq d_{s^*j}$) or $d_{is} > \frac{Q}{r}$ or $d_{sj} > \frac{Q}{r}$ then
            $L_{ij} := L_{ij} \setminus \{s\}$;
        else
            $\hat{t}_{ijs} := t_{is} + t_{sj} - t_{ij}$;
            $\hat{d}_{ijs} := d_{is} + d_{sj} - d_{ij}$;
        end if
    end for
end for
```

More specifically, the best AFS to visit between nodes i and j is the station s^* belonging to the minimum detour, among those that can be reached (according to the feasibility conditions on the remaining fuel), as determined at step 3. If a feasible AFS s is such that $d_{is} < d_{is^*}$ or $d_{sj} < d_{s^*j}$, then it can be advantageous to use it. In fact, if $d_{is} < d_{is^*}$, it can allow refueling an AFV that has not enough fuel to reach s^*. While, in the second case, the AFV can arrive to customer j with a higher fuel level than from s^*, guaranteeing it more driving range (either to reach the depot or to serve more customers). In any other case, AFS s can be removed from L_{ij} (steps 5 and 6). Finally, at steps 8 and 9, the detour time $\hat{t}_{ijs}$ of an AFV to visit the AFS $s \in L_{ij}$ going from i to j and its detour distance $\hat{d}_{ijs}$ are computed, respectively.

We model the G-VRP through the following binary variables: routing variables x_{ij} $\forall (i,j) \in A$, equal to 1 if node j is visited just after node i (directly or through an AFS), 0 otherwise; z_{ijs} $\forall (i,j) \in A, \forall s \in L_{ij}$ equal to 1 if AFS s is employed to go from i to j, 0 otherwise. Moreover, we use the following continuous non negative variables $\forall i \in N$: the residual fuel level y_i of AFV at i and the time τ_i on which i is reached.

The MILP model is detailed in the following:

$$\min \sum_{(i,j)\in A} d_{ij}x_{ij} + \sum_{(i,j)\in A} \sum_{s\in L_{ij}} \hat{d}_{ijs}z_{ijs} \tag{1}$$

s.t.

$$\sum_{s\in L_{ij}} z_{ijs} \le x_{ij} \quad \forall (i,j) \in A \tag{2}$$

$$\sum_{j\in N:\ j\neq i} x_{ij} = 1 \quad \forall i \in I \tag{3}$$

$$\sum_{i\in N:\ i\neq j} x_{ji} = \sum_{i\in N: i\neq j} x_{ij} \quad \forall j \in N \tag{4}$$

$$\sum_{j\in N:\ j\neq 0} x_{0j} \le m \tag{5}$$

$$\sum_{j\in N:\ j\neq 0} x_{j0} \le m \tag{6}$$

$$\tau_j \ge \tau_i + (t_{ij} + p_i)x_{ij} + \sum_{s\in L_{ij}} (\hat{t}_{ijs} + p_s)z_{ijs} - T_{max}(1 - x_{ij}) \forall i \in N, j \in I, i \neq j \tag{7}$$

$$\tau_j \le T_{max} - (t_{j0} + p_j) - \sum_{s\in L_{j0}} (\hat{t}_{j0s} + p_s)z_{j0s}\ \forall j \in I \tag{8}$$

$$y_j \le \sum_{s\in L_{ij}} (Q - r\cdot d_{sj})z_{ijs} + Q(1 - \sum_{s\in L_{ij}} z_{ijs})\ \forall j \in I, i \in I, i \neq j \tag{9}$$

$$y_j \le y_i - r\cdot d_{ij}x_{ij} + 2Q(1 - x_{ij} + \sum_{s\in L_{ij}} z_{ijs}) \quad \forall j \in I, i \in N, i \neq j \tag{10}$$

$$y_i \ge r\cdot d_{i0}(x_{i0} - \sum_{s\in L_{i0}} z_{i0s}) \quad \forall i \in I \tag{11}$$

$$y_i \ge \sum_{s\in L_{ij}} (r\cdot d_{is}z_{ijs}) \quad \forall i \in N, \forall j \in N, i \neq j \tag{12}$$

$$\sum_{s\in L_{i0}} r\cdot d_{s0}z_{i0s} \le Q \quad \forall i \in I \tag{13}$$

$$y_0 = Q \tag{14}$$

$$x_{ij} \in \{0,1\} \quad \forall i \in N, j \in N \tag{15}$$

$$y_i \geq 0 \quad \forall i \in N \tag{16}$$

$$\tau_i \geq 0 \quad \forall i \in N \tag{17}$$

$$z_{ijs} \in \{0,1\} \quad \forall i \in I, j \in I, s \in L_{ij} \tag{18}$$

Objective function (1) minimizes the total travel distance. Between each pair of customers, only one AFS may be visited (2) and each customer is visited exactly once (3). Route continuity is assured by (4). The number of AFVs is limited by (5)-(6). Arrival time at each node is ruled by (7) that also exclude sub-tours. Maximum route duration is enforced by (8). Fuel level, at each node, is ruled by (9)-(10). An AFV, after visiting its last node, must have enough fuel to return to the depot (11). An AFV, refueled after visiting its last customer, must have enough fuel to reach the depot thanks to (12)-(13). Constraints (14) impose that AFVs leave fully refueled the depot. Finally, the variables nature is specified in (15)-(18).

We note that the routing variables x_{ij} of this model are slightly different from those used in the formulation of (Koç and Karaoglan, 2016). Indeed, in the latter, x_{ij} is equal to 1 only if the customer j is visited directly after customer i without refueling at any AFS, while in ours, they are 1 also if the refuel occurs. Moreover, their refueling variables are defined considering the possibility of refueling at any AFS s between two customers i and j. While, in ours, they are defined only in the efficient non-dominated set L_{ij} of AFSs that may be actually used in an optimal solution. This makes our formulation more efficient, in practice.

In Section 7, we give an experimental evidence of this showing that our formulation is able to solve faster than (Koç and Karaoglan, 2016) the benchmark instances of the G-VRP.

5. Mathematical Programming Formulation with two consecutive visits to AFS

In this section, we extend the formulation presented in Section 4 to the case where two consecutive visits to the AFSs can occur in an optimal solution. Variables z_{ijs} are substituted by variables $z_{ijs_1s_2}$ equal to 1 if the pairs of AFSs s_1, s_2 are consecutively visited between the customers (or the depot) i and j, 0 otherwise.

Detours $\hat{t}_{ijs_1s_2}$ and $\hat{d}_{ijs_1s_2}$ are introduced, defined by $\hat{t}_{ijs_1s_2} = t_{is_1} + t_{s_1s_2} + t_{s_2j} - t_{ij}$ and $\hat{d}_{ijs_1s_2} = d_{is_1} + d_{s_1s_2} + d_{s_2j} - d_{ij}$, respectively.

The sets L_{ij} consists in all the pairs of feasible AFSs that may be convenient to visit, going from i to j. At the beginning, set L_{ij} is fixed as $L_{ij} := \{(s_1, s_2) : s_1, s_2 \in F, d_{is_1} \leq Q/r$ and $d_{s_2j} \leq Q/r\}$. Note that L_{ij} also contains pairs of kind (s,s) $\forall s \in F$ to model the use of a single AFS s.

Concerning the dominance criteria to remove stations from L_{ij}, we introduce two types of rules: those where a pair of AFSs is dominated by using a single AFS and those where it is dominated by using another pair of AFSs.

For the first type, a pair $(s_1, s_2) \in L_{ij}$ can be removed from L_{ij} if $d_{s_1j} < d_{s_2j}$ or if $d_{is_2} < d_{is_1}$. Indeed, in the former case, it is convenient to visit only the AFS s_1 while, in the latter case, only s_2. Concerning the second type of dominance rule, a pair $(s_3, s_4) \in L_{ij}$ can be removed from L_{ij} if a pair $(s_1, s_2) \in L_{ij}$ such that $d_{is_1} < d_{is_3}$, $d_{s_2j} < d_{s_4j}$ and $\hat{d}_{ijs_3s_4} > \hat{d}_{ijs_1s_2}$ exists. Indeed, the pair (s_3, s_4) would be not convenient neither to minimize the total travel distance since its detour is longer than that of (s_1, s_2) nor to serve an AFV coming from i with a lower driving range (since $d_{is_1} < d_{is_3}$) nor to arrive to j with a higher driving range (since $d_{s_2j} < d_{s_4j}$).

The pseudocode of the algorithm used to compute the new sets L_{ij}, according to these dominance rules, is reported in Algorithm 2.

The new formulation, where two consecutive visits to the AFS are allowed, is the following:

$$\min \sum_{(i,j)\in A} d_{ij}x_{ij} + \sum_{(i,j)\in A} \sum_{(s_1,s_2)\in L_{ij}} \hat{d}_{ijs_1s_2} z_{ijs_1s_2} \tag{19}$$

s.t.

Algorithm 2 Computation of the sets L_{ij} with double AFSs

```
Input: sets $A, F$, parameters $Q$, $r$, $d_{ij}$ and $t_{ij}$ $\forall (i,j) \in A$
Output: sets $L_{ij}$;
for $(i,j) \in A$ do
    $L_{ij} := \{(s_1,s_2) : s_1, s_2 \in F, d_{is_1} \leq \frac{Q}{r}, d_{s_2 j} \leq \frac{Q}{r}, d_{s_1 j} \geq d_{s_2 j}$ and $d_{is_2} \geq d_{is_1}\}$;
    for $(s_1,s_2) \in L_{ij}$ do
        $\hat{t}_{ijs_1s_2} := t_{is_1} + t_{s_1s_2} + t_{s_2j} - t_{ij}$;
        $\hat{d}_{ijs_1s_2} := d_{is_1} + d_{s_1s_2} + d_{s_2j} - d_{ij}$;
    end for
    for $(s_1,s_2),\ (s_3,s_4) \in L_{ij}$ do
        if $d_{is_1} \leq d_{is_3}$, $d_{s_2j} \leq d_{s_4j}$ and $\hat{d}_{ijs_3s_4} > \hat{d}_{ijs_1s_2}$ then
            $L_{ij} := L_{ij} \setminus \{(s_3,s_4)\}$;
        end if
    end for
end for
```

$$\sum_{(s_1,s_2)\in L_{ij}} z_{ijs_1s_2} \leq x_{ij} \quad \forall\,(i,j) \in A \tag{20}$$

$$\sum_{j\in N:\ j\neq i} x_{ij} = 1 \quad \forall i \,\in\, I \tag{21}$$

$$\sum_{i\in N:\ i\neq j} x_{ji} = \sum_{i\in N: i\neq j} x_{ij} \quad \forall\, j \,\in\, N \tag{22}$$

$$\sum_{j\in N:\ j\neq 0} x_{0j} \leq m \tag{23}$$

$$\sum_{j\in N:\ j\neq 0} x_{j0} \leq m \tag{24}$$

$$\tau_j \geq \tau_i + (t_{ij} + p_i)x_{ij} + \sum_{(s_1,s_2)\in L_{ij}} (\hat{t}_{ijs_1s_2} + p_{s_1} + p_{s_2}) z_{ijs_1s_2} - T_{max}(1 - x_{ij}) \ \forall i \in N, j \in I, i \neq j \tag{25}$$

$$\tau_j \leq T_{max} - (t_{j0} + p_j) - \sum_{(s_1,s_2)\in L_{j0}} (\hat{t}_{j0s_1s_2} + p_{s_1} + p_{s_2}) z_{j0s_1s_2} \ \forall j \in I \tag{26}$$

$$y_j \leq \sum_{(s_1,s_2)\in L_{ij}} (Q - r\cdot d_{s_2j}) z_{ijs_1s_2} + Q(1 - \sum_{(s_1,s_2)\in L_{ij}} z_{ijs_1s_2}) \quad \forall j \in I, i \in I, i \neq j \tag{27}$$

$$y_j \leq y_i - r\cdot d_{ij}x_{ij} + 2Q(1 - x_{ij} + \sum_{(s_1,s_2)\in L_{ij}} z_{ijs_1s_2}) \quad \forall j \in I, i \in N, i \neq j \tag{28}$$

$$y_i \geq r\cdot d_{i0}(x_{i0} - \sum_{(s_1,s_2)\in L_{ij}} z_{i0s_1s_2}) \quad \forall i \,\in\, I \tag{29}$$

$$y_i \geq \sum_{(s_1,s_2)\in L_{ij}} (r\cdot d_{is_1} z_{ijs_1s_2}) \quad \forall (i,j) \in A \tag{30}$$

$$\sum_{(s_1,s_2)\in L_{ij}} r\cdot d_{s_20} z_{i0s_1s_2} \leq Q \quad \forall i \,\in\, I \tag{31}$$

$$y_0 = Q \tag{32}$$

$$x_{ij} \in \{0,1\} \quad \forall (i,j) \in A \tag{33}$$

$$y_i \geq 0 \quad \forall i \in N \tag{34}$$

$$\tau_i \geq 0 \quad \forall i \in N \tag{35}$$

$$z_{ijs_1s_2} \in \{0,1\} \quad \forall i \in I, j \in I, (s_1, s_2) \in L_{ij} \tag{36}$$

Constraints (h) of the new formulation have the same meaning of constraints $(h-18)$ of the formulation of Section 4, for $h = 20, \ldots, 36$.

6. Valid inequalities

In this section, we introduce some valid inequalities to strengthen the formulation of the G-VRP presented in Section 5. These inequalities can be used also in the formulation with one visit to the AFSs, presented in Section 4, since the latter can be reduced to the former considering only the pairs of stations $(s_1, s_2) \in L_{ij}$ with $s_1 = s_2$.

The first kind of valid inequalities is given by (37). It enforces to refuel between a pair of visited customers, i and j, if they are farther than the maximum distance that can be traveled by a completely refueled AFV.

$$\sum_{(s_1,s_2) \in L_{ij}} z_{ijs_1s_2} \geq x_{ij} \quad \forall\, (i,j) \in A : d_{ij} > Q/r \tag{37}$$

Valid inequalities (38) prevent from refueling after the last customer visited if there is enough fuel to reach the depot. Indeed, when the latter condition is satisfied, the right hand side of (38) becomes strictly less than 1, being ε a positive, small enough constant and $y_i - rd_{i0} > 0$. Thus, $z_{i0s_1s_2}$ must be 0.

$$\sum_{(s_1,s_2) \in L_{i0}} z_{i0s_1s_2} \leq 1 - \varepsilon \cdot (y_i - rd_{i0}) \quad \forall i \in V \tag{38}$$

Being $y_i - rd_{i0} \leq Q$, we set ε equals to $1/Q$ to guarantee that the right hand side of (38) is nonnegative.

Let us indicate by $D = \max_{i \in N \cup F, j \in N \cup F : i \neq j} d_{ij}$. Valid inequalities (39) prevent from serving a triplet of customers i, j and k without intermediate refueling if the sum of their distances (i.e., $d_{ij} + d_{jk}$) plus the distance to reach either the nearest station or the depot exceeds the maximum distance that can be covered by a completely refueled vehicle. Indeed, in the latter case, the right hand side of (39) becomes strictly less than 2, (since the first two summations are 0 and the term $(d_{ij} + d_{jk} + \min_{s \in F \cup \{0\}} d_{ks} - y_i/r)/3D$ becomes a positive number smaller than 1. Thus, x_{ij} or x_{jk} must be 0).

$$x_{ij} + x_{jk} \leq 2 + \sum_{(s_1,s_2) \in L_{ij}} z_{ijs_1s_2} + \sum_{(s_1,s_2) \in L_{jk}} z_{jks_1s_2} - \frac{d_{ij} + d_{jk} + \min_{s \in F \cup \{0\}} d_{ks} - y_i/r}{3D} \quad \forall (i,j) \in A, \forall (j,k) \in A \tag{39}$$

In similar way, valid inequalities (40) prevent from serving a triplet of customers i, j and k without intermediate refueling if the sum of their distances plus the distance to either reach the nearest station or the depot exceeds the maximum distance that can be covered within the duty time of the AFV, given by T_{max}.

$$x_{ij} + x_{jk} \leq 2 + \sum_{(s_1,s_2)\in L_{ij}} z_{ijs_1s_2} + \sum_{(s_1,s_2)\in L_{jk}} z_{jks_1s_2} -$$

$$\frac{d_{ij} + d_{jk} + \min_{s\in F\cup\{0\}} d_{ks} - (T_{max} - \tau_i)\cdot v}{3D} \quad \forall (i,j)\in A, \forall (j,k)\in A \tag{40}$$

7. Instances description and numerical results

In this section, we present computational results carried out on instances taken from the literature and we introduce a new set of real-like instances, in which consecutive refuelings actually occur in the optimal solutions.

7.1. Instances from the literature

In this subsection, we compare the performances of the formulation for the G-VRP introduced in Section 4, hereafter named BMP, with that proposed by (Erdoğan and Miller-Hooks, 2012) denoted by EMH and the Branch and Cut proposed by (Koç and Karaoglan, 2016) indicated by B&C. Computational results are carried out on the following four sets of instances introduced in (Erdoğan and Miller-Hooks, 2012):

- **S1 (Uniform customer distribution)**: 10 randomly generated instances of 20 uniformly distributed customers with three different AFS locations.
- **S2 (Clustered customer distribution)**: 10 randomly generated instances of 20 clustered customers with three different AFS locations.
- **S3 (Impact of the spatial AFS configuration)**: 10 instances, half selected from S1 (i.e., 20c3sU2, U4, U6, U8, U10) and half from S2 (i.e., 20c3sC2, C4, C6, C8, C10), with 6 AFS randomly generated locations.
- **S4 (Impact of station density)**: 10 instances, half of which created from one instance of S1, i.e., 20c3sU2 and half from one instance of S2, i.e., 20c3sC2. While, the number of AFSs is gradually increased from 2 to 10 by 2.

Each instance is randomly generated assuming customers, depot and AFSs distributed within a grid of 330 by 300 miles. The depot location is fixed and close to the center of the grid, in all the instance sets. In all the instances, the depot is also an AFS. In the first two sets, S1 and S2, three further AFSs, fixed, are between the depot and the grid boundaries in westerly, northerly and southeasterly directions.

A fuel tank capacity Q equal to 60 gallons and a fuel consumption rate r of 0.2 gallons per mile are considered. The average vehicle speed is assumed equal to 40 miles per hour and the total route duration T_{max} is limited to 11 hours. Service times p_i, at customers $i \in N$, are assumed to be equal to 30 minutes while refueling time p_s at each AFS s is fixed to 15 minutes. Even if it has not been clearly stated in (Erdoğan and Miller-Hooks, 2012), the authors consider also the initial refueling time at the depot. For each node of the network (i.e., customer, depot and AFS), the geographical coordinates are given while, the distance between each pair of nodes is computed by means of the Haversine formula considering an Earth radius of $4,182.449$ miles, as pointed out in (Schneider et al., 2013). Concerning the maximum number of vehicles used (m), since in (Erdoğan and Miller-Hooks, 2012) it is not declared, we fix it equal to the number of vehicles used in the best known solution reported in the literature. Therefore, in set $S1$, m ranges from 4 to 6, in $S2$ from 3 to 9, in $S3$ from 4 to 7 and in $S4$, from 4 to 6.

Both BMP and EMH models have been solved with CPLEX 12.5 on an Intel Core i7-2630QM CPU at 2.00 GHz with 8 GB RAM, while B&C has been solved by (Koç and Karaoglan, 2016) on an Intel Xeon 3.16 GHz equipped with 8 GB RAM (i.e., on a slightly powerful computer), also using CPLEX 12.5 as LP solver. The CPU time limit has been set equal to 3,600 seconds for all the solution methods.

Place Table 2 about here

Place Table 3 about here

Place Table 4 about here

Place Table 5 about here

Numerical comparisons are reported in Tables 2-5, one for each instance set. The tables are organized as follows. The first three columns report the name of the instance, the number of involved customers and the number of AFSs, respectively. The successive three columns report the best feasible solutions found by the three approaches EMH, B&C, and BMP, respectively. The values are emphasized in boldface when the optimality has been certified. The successive three columns report the gap between the best feasible solution and the best lower bound, while computational times, expressed in seconds, are shown in the last three columns. Moreover, in the tables, *no sol* means that no feasible solution is found in the time limit while *N/A* indicates that the GAP is not available for that instance.

On the instance set S1, the BMP formulation gets 7 optimal solutions against the 5 found by B&C, with a lower average computational time ($1,772$ seconds against $2,373$) and an average relative MIP gap of 3.2% against 3.5% of B&C. EMH is not able to solve to optimality any instance within the time limit, obtaining an average relative MIP gap of about 27%. Moreover, on one instance, it is not able even to find a feasible solution. On S2, both BMP and B&C solve to optimality 8 instances. However, BMP requires a slightly lower average computational time (about $1,060$ seconds against $1,293$). EMH is not able to solve to optimality any instance within the time limit, and obtains an average relative MIP gap of about 34%. Moreover, on two instances, it is not able even to find a feasible solution. On S3, BMP detects the optimal solution in 9 instances, while B&C in only 8 instances. BMP obtains both a lower average computational time (about $1,040$ seconds against $1,204$) and a lower average relative MIP gap (0.68% against 1.38%) with respect to B&C. EMH does not find any feasible solution on 5 instances and does not solve any instance to optimality, obtaining an average relative MIP gap of 36%. Finally, on S4, B&C obtains only one optimal solution while BMP solves to optimality 6 instances within a significantly lower average computational time (about $2,164$ seconds against $3,453$) and with a lower average relative MIP gap (3.66% against 5.38%). EMH finds a feasible solution only on 5 instances, with an average relative MIP gap of 30%, and is not able to solve to optimality any instance. Globally, BMP strongly outperforms B&C and EMH in terms of both efficiency and effectiveness.

In order to prove the impact of the dominance rules (to generate the sets L_{ij}, as described in Section 4) on the efficiency, we compare the computational times required to solve the BMP with the dominance rules with those required without them. On S1, we obtain an average CPU time reduction of 13%, on S2 of 11%, on S3 of 14% and on S4 of 16%. Moreover, without the dominance rules, we solve to optimality three instances less (one in S1, one in S2 and one in S4). This is justified by the fact that applying the dominance rules the cardinality of sets L_{ij} significantly decreases compared to the cardinality of all the available AFSs, i.e., $|F|$. Indeed, on S1, the average cardinality of L_{ij} is 2.19 against $|F| = 4$, on S2, it is 2.07 against $|F| = 4$, on S3, it is 3.34 against $|F| = 7$, and finally, on S4, it is 3.52 against an average $|F| = 7$. Therefore, on average, for all the instances, $|L_{ij}|$ is about half of $|F|$.

Moreover, on the same set of instances, we also run the model introduced in Section 5, where two consecutive visits to the AFSs are allowed. In order to always obtain the optimal solution, we increase the CPU time limit of CPLEX to 10,800 seconds, being this second model more time consuming since it requires bigger sets L_{ij}. Indeed, the computational time increases of 11.8% on average. Concerning the optimal solutions, we remark that they always coincide with those obtained by the BMP model. Therefore, we can certify that for the benchmark instances of the G-VRP introduced by (Erdoğan and Miller-Hooks, 2012), consecutive visits to AFSs are not necessary in the optimal solutions found.

As a further analysis, we run the EMH model, on the set S1, with larger CPU time limits, i.e., 12 and 24 hours, respectively. Results are reported in Table 6, where the values in bold indicate cases in which, thanks to the increased time limit, an improvement on the objective function has been obtained. What has been pointed out by this analysis is that the EMH model is not able to solve to optimality any instance, except one, and that the average percentage GAP decreases only slightly, passing from 26.97% in 1 hour to 20.61% in 24 hours.

Place Table 6 about here

7.2. Analysis of the solutions found

For all the instance sets considered in the computational experiments, we have reported some useful information about the optimal solution (or the best solution found, when its optimality cannot be certified). This analysis, detailed in the Tables 7-10, allows us deducing some hints on the usage of AFVs. For each instance, the distance covered to serve the customers (SERVICE), that to reach the AFSs (DETOUR) and the percentage value of DETOUR respect to the total travel distance are reported. Furthermore, for each AFS, the number of times it is visited together with the number of vehicles that use them are indicated. The term *N/A*, used in Table 10, indicates that the corresponding station is not available for that instance.

Place Table 7 about here

Place Table 8 about here

Place Table 9 about here

Place Table 10 about here

The percentage of distance covered to reach an AFS is low in all sets, meaning that the AFSs are well distributed. The average percentage of vehicles that refuel along their route is greater than 40%, meaning that the instances are suitable for the G-VRP model. Indeed, if the percentage of vehicles that requires to be refueled would be very low, there will be no need to explicitly plan the visits to the AFSs. In fact, the problem can be addressed as a standard VRP and the few cases in which a vehicle needs refueling could be a posteriori detected and the infeasibility may be repaired trying to insert a visit to an AFS.

From a further analysis, only in 35% of the cases, the AFS visited is that at minimum detour. This happens when the AFV has not enough fuel to reach the minimum detour station. Therefore, it needs to visit a nearer AFS with a longer detour to reach the next customer. In a real world application, the problem could be solved intensifying the density of AFSs in these zones.

Although the model proposed in (Erdoğan and Miller-Hooks, 2012) allows consecutive visits to AFSs, in the benchmark instances here introduced, they never occur in any optimal solution. For this reason, in the next section, we introduce a new set of instances in order to deeply investigate this issue.

7.3. Real-like instances

In this section, we introduce a new set of instances, named *Cloud*, in which two consecutive refuelings are necessary in the optimal solutions. This set is composed by 10 instances with 15 customers and 8 AFSs, sharing the same layout specified as follows. The depot is located at the center of a square with edge equal to 160 miles. At the same distance from the depot, 80 miles, in each main direction (North, South, East and West), four 'clouds', containing between 2 and 6 customers, are located. The number of customers per cloud varies instance by instance but the total number of customers remains unvaried. In the nearby of each cloud, two AFSs are located toward the two nearest clouds. An example of a cloud instance is reported in Figure 2. This may represent the case in which a depot is located in Milan and the goods has to be delivered to other large cities such as Turin, Bologna, Venice and Bern, and no AFS is available along the highway but all AFSs are located in the nearby of the cities. In this set, AFSs locations are fixed, as well as the total number of customers. While customers' locations vary instance by instance. The number of available vehicles is fixed to 3. Fuel tank capacity, Q, is equal to 30 gallons that, combined with a fuel consumption rate, r, of 0.2, implies a maximum autonomy of the vehicles, D_{max}, equal to 150 miles. Maximum route duration, T_{max}, is fixed to 11 hours and no time for the initial refueling at the depot is considered. Service times p_i, at customers $i \in N$, are assumed to be equal to 5 minutes, while all other parameters are unchanged respect to the values considered in the instance sets S1-S4.

Place Figure 2 about here

Analyzing Figure 2, it can be easily noted that, although feasible solutions without two consecutive refueling stops may exist, optimal solutions will always contain consecutive refueling stops. Indeed, given the instance layout, it is always more convenient to serve two clouds with a single AFV. In fact, if two AFVs were used, each of them would serve one cloud and their total traveled distance would be $4D$, being D the distance between the depot and a cloud (the clouds are equidistant from the depot). While, a single AFV can serve the two clouds with the following route: going from the depot to the first cloud, from it to the second one, after refueling with zero detour at the two AFSs located on the line connecting the two clouds, and then returning to the depot. In this way, the total traveled distance is $2D + \sqrt{2}D < 4D$. Instead, if consecutive refueling stops are not allowed, an AFV must refuel at the station close to the first cloud, then serve one or more customers in the second cloud and then refuel again. The second refueling generates a non-zero detour making the solution sub-optimal.

Place Table 11 about here

Table 11 reports a comparison of the results obtained by the model presented in Section 5, indicated by BMP2, with those of EMH and of KK, giving to all methods a CPU time limit of 43,200 seconds. As one can evince from this table, BMP2 is able to find the optimal solution for all instances, with an average computational time of 825.44 seconds. EMH always reaches the CPU time limit without finding the optimal solution, with an average gap of 62.04%. The optimal solutions obtained by KK are always sub-optimal for these instances, because this model does not allow consecutive visits to AFSs. In this way, the average total distance provided by KK is 591.83 against 582.94 of that found by BMP2.

Place Table 12 about here

In order to show the impact of the valid inequalities presented in Section 6 and of the dominance rules to reduce the cardinality of the sets L_{ij} proposed in Section 5, we compare the results obtained on the Cloud instances by the model without both Valid Inequalities and Dominance Rules, (NO VI- NO DR), with those obtained adding only Dominance Rules (DR), and those obtained adding both of them (VI+DR). As proved by the results reported in Table 12, both VI and DR have a significant impact on the CPU times reduction. In fact, without VI and DR, the average CPU time is equal to 825.44, while, it decreases to 500.79 with the addition of DR and it further decreases to 349.19 adding also VI. We can notice that, applying DR, we can save, on average, about 40% of the computational time. This saving is much greater than that obtained on the EMH instances (14% on average). This is due to the fact that, giving the particular layout of the Cloud instances, DR allow strongly reducing the size of the sets L_{ij} because of the high number of dominated AFSs. While, the effect of DR on the reduction of the L_{ij} size is lower in the EMH instances.

8. Conclusions and future works

In this paper, we proposed two Mixed Integer Linear Programming (MILP) formulations for the Green Vehicle Routing Problem (G-VRP) such that multiple visits to Alternative Fuel Stations (AFSs) are allowed without introducing dummy nodes to model them. In the first formulation, only one visit to an AFS is allowed between either two customers or a customer and the depot. While, in the second formulation, two consecutive visits to AFSs are permitted. The latter formulation is enough general since we proved that, under mild hypothesis, at most two consecutive visits to AFSs are necessary in an optimal solution. Getting rid of dummy nodes, we obtained a strong reduction of the number of binary variables used in the two MILP models compared to the previous mathematical formulations of the literature. A further reduction was obtained introducing some dominance criteria to a-priori identify an efficient set of AFSs. Moreover, the MILP models were strengthened by means of some valid inequalities specifically tailored for the G-VRP.

Computational results, carried out on 40 benchmark instances taken from the literature, showed that both the proposed MILPs models strongly outperform the already existing exact solution approaches and even a Branch and Cut method. Since we noted that, on the benchmark instances, consecutive refueling stops never occurred, we built a new set of instances where this actually happens in the optimal solutions. Also on this set, our second formulation strongly outperforms all known exact methods always detecting the optimal solution. Moreover, we showed how much the valid inequalities and the dominance criteria contribute to reduce the computational time.

Finally, we reported a detailed analysis of the optimal solutions providing, for each instance, some transport indicators, i.e., the average percentage value of the distance to reach the AFSs respect to the total travel distance and for each AFS, the number of times it is visited and by how many vehicles.

Future developments in this field may concern the extension of the proposed MILP formulations to similar optimization problems, e.g., the Electric Vehicle Routing Problem with Time Windows.

References

Bektaş, T., Demir, E., Laporte, G., 2016. Green vehicle routing. In: Green Transportation Logistics. Springer, pp. 243–265.

Chapman, L., 2007. Transport and climate change: a review. Journal of transport geography 15 (5), 354–367.

Ćirović, G., Pamučar, D., Božanić, D., 2014. Green logistic vehicle routing problem: Routing light delivery vehicles in urban areas using a neuro-fuzzy model. Expert Systems with Applications 41 (9), 4245–4258.

Davis, S., Diegel, S., Boundy, R., 2015. Transportation data book. edition 34, ornl-6991. In: Department of Energy, 2015. Center for Transportation Analysis Energy and Transportation Science Division, pp. 1–440.

Edenhofer, O., Pichs-Madruga, R., Sokona, Y., Farahani, E., Kadner, S., Seyboth, K., Adler, A., Baum, I., Brunner, S., Eickemeier, P., et al., 2014. Climate change 2014: Mitigation of climate change. Working group III contribution to the fifth assessment report of the Intergovernmental Panel on Climate Change. UK and New York.

Eglese, R., Bektas, T., 2014. Green vehicle routing. Vehicle Routing: Problems, Methods, and Applications 18, 437.

Erdoğan, S., Miller-Hooks, E., 2012. A green vehicle routing problem. Transportation Research Part E: Logistics and Transportation Review 48 (1), 100–114.

Felipe, Á., Ortuño, M. T., Righini, G., Tirado, G., 2014. A heuristic approach for the green vehicle routing problem with multiple technologies and partial recharges. Transportation Research Part E: Logistics and Transportation Review 71, 111–128.

Jabbarpour, M. R., Noor, R. M., Khokhar, R. H., 2015. Green vehicle traffic routing system using ant-based algorithm. Journal of Network and Computer Applications 58, 294–308.

Jemai, J., Zekri, M., Mellouli, K., 2012. An nsga-ii algorithm for the green vehicle routing problem. In: Evolutionary computation in combinatorial optimization. Springer, pp. 37–48.

Koç, Ç., Karaoglan, I., 2016. The green vehicle routing problem: A heuristic based exact solution approach. Applied Soft Computing 39, 154–164.

Kontovas, C. A., Psaraftis, H. N., 2016. Transportation emissions: Some basics. In: Green Transportation Logistics. Springer, pp. 41–79.

Madankumar, S., Rajendran, C., 2018. Mathematical models for green vehicle routing problems with pickup and delivery: A case of semiconductor supply chain. Computers & Operations Research 89, 183–192.

Mancini, S., 2017. The hybrid vehicle routing problem. Transportation Research Part C: Emerging Technologies 78, 1–12.

Montoya, A., Guéret, C., Mendoza, J. E., Villegas, J. G., 2016. A multi-space sampling heuristic for the green vehicle routing problem. Transportation Research Part C: Emerging Technologies 70, 113–128.

Qian, J., Eglese, R., 2016. Fuel emissions optimization in vehicle routing problems with time-varying speeds. European Journal of Operational Research 248 (3), 840–848.

Sbihi, A., Eglese, R. W., 2007. Combinatorial optimization and green logistics. 4OR 5 (2), 99–116.

Schneider, M., Stenger, A., Goeke, D., 2013. Appendix of the electric vehicle routing problem with time windows and recharging stations.http://www.logistikplanung.tu-darmstadt.de/media/logplanung/publikationen-4/veroeffentlicht/appendix.pdf.

Sharafi, A., Bashiri, M., 2016. Green vehicle routing problem with safety and social concerns. Journal of Optimization in Industrial Engineering 10 (21), 93–100.

Tiwari, A., Chang, P.-C., 2015. A block recombination approach to solve green vehicle routing problem. International Journal of Production Economics 164, 379–387.

Toth, P., Vigo, D., 2014. Vehicle routing: problems, methods, and applications. SIAM.

Xiao, Y., Konak, A., 2016. The heterogeneous green vehicle routing and scheduling problem with time-varying traffic congestion. Transportation Research Part E: Logistics and Transportation Review 88, 146–166.

Zhang, J., Zhao, Y., Xue, W., Li, J., 2015. Vehicle routing problem with fuel consumption and carbon emission. International Journal of Production Economics 170, 234–242.

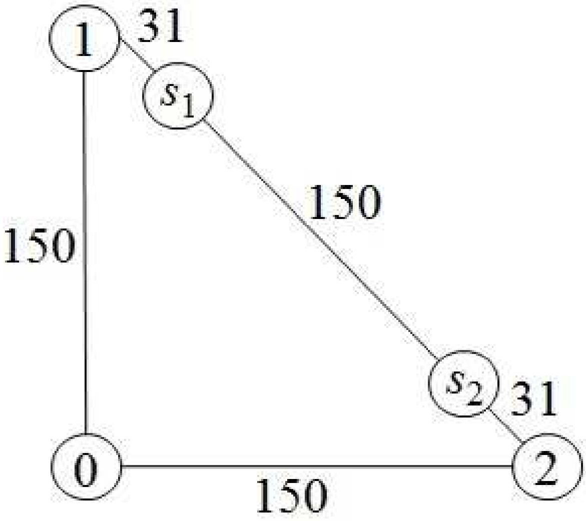

Figure 1:

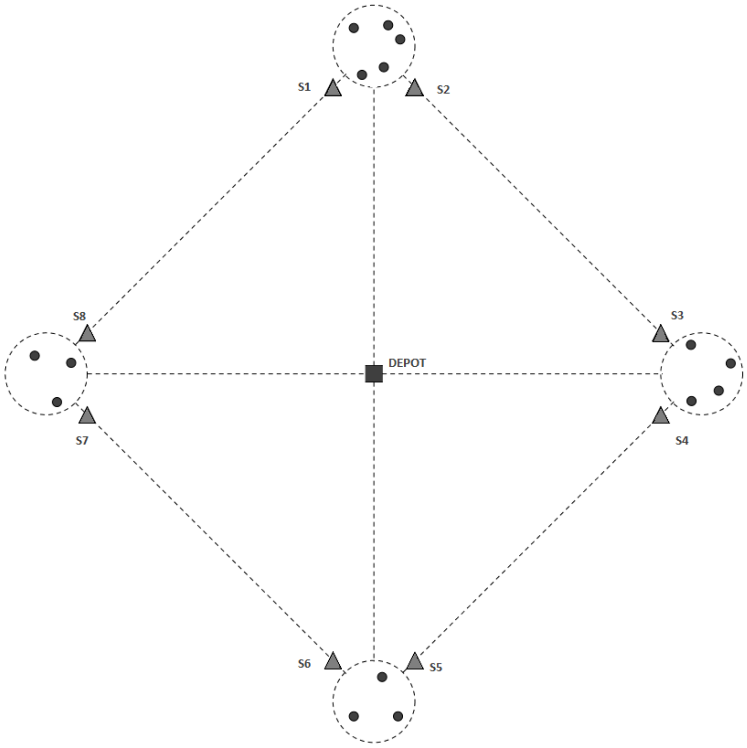

Figure 2:

List of Figure captions:

- Figure 1: An instance of the G-VRP where two consecutive visit to the AFSs are necessary in the optimal solution.
- Figure 2: Cloud instances layout.

Input Parameter	Meaning
I	Set of customers
0	Depot
N	Set of customers plus the depot
F	Set of available AFSs
m	Number of AFVs
v	Vehicle speed
Q	Maximum refueling capacity of each AFV
r	Fuel consumption rate of each AFV
p_i	Service time of each customer i
p_s	Refueling time of each AFS s
d_{ij}	Travel distance for going from node i to node j
t_{ij}	Travel time for going from node i to node j
T_{max}	Maximum route duration
$\hat{t}_{ijs}$	Detour time for going from node i to node j passing from AFS s
$\hat{d}_{ijs}$	Detour distance for going from node i to node j passing from AFS s

Table 1:

INSTANCE	$\|I\|$	$\|F\|$	UB			GAP			CPU (s)		
			EMH	B&C	BMP	EMH	B&C	BMP	EMH	B&C	BMP
20c3sU1	20	4	1,797.51	**1,797.51**	**1,797.51**	29.07	0.00	0.00	3,600	172.3	330.3
20c3sU2	20	4	1,574.82	1,574.82	**1,574.82**	26.66	4.70	0.00	3,600	3,600	1,083.64
20c3sU3	20	4	1,704.46	**1,704.46**	**1,704.46**	29.21	0.00	0.00	3,600	1,789	843.78
20c3sU4	20	4	no sol	1,482.00	**1,482.00**	N/A	9.50	0.00	3,600	3,600	1,842
20c3sU5	20	4	1,689.35	**1,689.35**	1,689.35	20.19	0.00	10.13	3,600	2,165.5	3,600
20c3sU6	20	4	1,618.64	1,618.64	**1,618.64**	21.9	2.70	0.00	3,600	3,600	1,747.53
20c3sU7	20	4	1,713.80	1,713.66	1,713.66	36.32	9.80	6.34	3,600	3,600	3,600
20c3sU8	20	4	1,709.43	**1,706.51**	**1,706.51**	32.61	0.00	0.00	3,600	1,601.3	1,020.26
20c3sU9	20	4	1,708.84	1,708.84	1,708.84	35.99	8.30	15.69	3,600	3,600	3,600
20c3sU10	20	4	1,181.31	**1,181.31**	**1,181.31**	10.75	0.00	0.00	3,600	2.3	55.2
AVG	**20**	**4**	**1633.13**	**1,617.71**	**1,617.71**	**26.97**	**3.50**	**3.22**	**3,600**	**2,373.04**	**1,772.27**

Table 2:

INSTANCE	$\|I\|$	$\|F\|$	UB			GAP			CPU (s)		
			EMH	B&C	BMP	EMH	B&C	BMP	EMH	B&C	BMP
20c3sC1	20	4	1,179.17	1,173.58	**1,173.58**	43.62	7.60	0.00	3,600	3,600	2,223.82
20c3sC2	19	4	no sol	**1,539.95**	**1,539.95**	N/A	0.00	0.00	3,600	1,164.5	612.97
20c3sC3	12	4	880.21	**880.21**	**880.21**	9.87	0.00	0.00	3,600	25.4	8.45
20c3sC4	18	4	1,109.73	1,059.35	1,059.35	38.62	11.00	12.60	3,600	3,600	3,600
20c3sC5	19	4	2,156.02	**2,156.01**	2,156.01	64.16	0.00	35.83	3,600	2,246.4	3,600
20c3sC6	17	4	2,758.14	**2,758.14**	**2,758.14**	46.64	0.00	0.00	3,600	61.6	17.28
20c3sC7	6	4	1,393.99	**1,393.99**	**1,393.99**	10.70	0.00	0.00	3,600	0.1	0.4
20c3sC8	18	4	no sol	**3,139.70**	**3,139.70**	N/A	0.00	0.00	3,600	53.7	14.96
20c3sC9	19	4	1,813.93	**1,799.95**	**1,799.95**	33.19	0.00	0.00	3,600	113.9	90.59
20c3sC10	15	4	2,583.42	**2,583.42**	**2,583.42**	23.01	0.00	0.00	3,600	2,067.5	427.5
AVG	**19**	**4**	**1734.33**	**1848.43**	**1848.43**	**33.73**	**1.86**	**4.84**	**3,600**	**1,293.31**	**1,059.60**

Table 3:

INSTANCE	$\|I\|$	$\|F\|$	UB			GAP			CPU (s)		
			EMH	B&C	BMP	EMH	B&C	BMP	EMH	B&C	BMP
S1_2i6s	20	7	no sol	**1,578.15**	**1,578.15**	N/A	0.00	0.00	3,600	1,626.6	1,775.39
S1_4i6s	20	7	no sol	1,397.27	**1,397.27**	N/A	4.30	0.00	3,600	3,600	782.26
S1_6i6s	20	7	no sol	**1,560.48**	**1,560.48**	N/A	0.00	0.00	3,600	523.50	1,107.05
S1_8i6s	20	7	no sol	**1,692.34**	**1,692.34**	N/A	0.00	0.00	3,600	817.80	1,745.88
S1_10i6s	20	7	1,173.48	**1,173.48**	**1,173.48**	22.69	0.00	0.00	3,600	1.90	98.03
S2_2i6s	20	7	1,645.8	**1,633.09**	**1,633.09**	35.34	0.00	0.00	3,600	66.00	524.8
S2_4i6s	19	7	1,505.06	1,505.06	1,505.06	34.72	9.50	6.82	3,600	3,600	3,600
S2_6i6s	20	7	no sol	**2,431.34**	**2,431.34**	N/A	0.00	0.00	3,600	1,801.00	13.63
S2_8i6s	16	7	2,176.4	**2,158.34**	**2,158.34**	38.79	0.00	0.00	3,600	3.20	5.68
S2_10i6s	16	7	1,621.42	**1,585.43**	**1,585.43**	47.85	0.00	0.00	3,600	1.30	749.36
AVG	**20**	**7**	**1624.43**	**1671.50**	**1671.50**	**35.88**	**1.38**	**0.68**	**3,600**	**1,204.13**	**1,040.21**

Table 4:

INSTANCE	$\|I\|$	$\|F\|$	UB			GAP			CPU (s)		
			EMH	B&C	BMP	EMH	B&C	BMP	EMH	B&C	BMP
S1_4i2s	20	3	1,582.22	1,582.22	**1,582.22**	18.17	6.20	0.00	3,600	3,600	2,340.2
S1_4i4s	20	5	1,460.10	1,460.10	**1,460.10**	31.32	3.80	0.00	3,600	3,600	621.18
S1_4i6s	20	7	no sol	1,397.28	**1,397.28**	N/A	4.30	0.00	3,600	3,600	770.94
S1_4i8s	20	9	no sol	**1,397.28**	**1,397.28**	N/A	0.00	0.00	3,600	2,133.6	815.75
S1_4i10s	20	11	no sol	1,396.02	**1,396.02**	N/A	5.00	0.00	3,600	3,600	570.28
S2_4i2s	18	3	1,059.36	1,059.36	1,059.36	28.72	8.20	12.58	3,600	3,600	3,600
S2_4i4s	19	5	1,496.56	1,446.10	1,446.10	38.38	6.20	7.61	3,600	3,600	3,600
S2_4i6s	20	7	1,505.06	1,434.14	1,505.06	34.81	6.80	6.62	3,600	3,600	3,600
S2_4i8s	20	9	no sol	1,434.14	1,434.14	N/A	6.60	9.78	3,600	3,600	3,600
S2_4i10s	20	11	no sol	1,434.14	**1,434.14**	N/A	6.70	0.00	3,600	3,600	2,121.01
AVG	**20**	**7**	**1420.66**	**1404.08**	**1411.17**	**30.28**	**5.38**	**3.66**	**3,600**	**3,453.36**	**2,163.94**

Table 5:

INSTANCE	EMH CPU limit 1h			EMH CPU limit 12h			EMH CPU limit 24h		
	UB	GAP	CPU (s)	UB	GAP	CPU (s)	UB	GAP	CPU (s)
20c3sU1	1,797.51	29.07	3,600	1,797.51	24.00	43,200	1797.51	20.77	86,400
20c3sU2	1,574.82	26.66	3,600	1,574.82	24.04	43,200	1574.82	23.15	86,400
20c3sU3	1,704.46	29.21	3,600	1,704.46	23.29	43,200	1704.46	22.26	86,400
20c3sU4	no sol	N/A	3,600	**1,482.00**	31.80	43,200	**1,482.00**	28.93	86,400
20c3sU5	1,689.35	20.19	3,600	1,689.35	22.34	43,200	1689.35	17.00	86,400
20c3sU6	1,618.64	21.9	3,600	1,618.64	20.69	43,200	1618.64	18.51	86,400
20c3sU7	1,713.80	36.32	3,600	**1,713.67**	29.77	43,200	**1713.67**	26.72	86,400
20c3sU8	1,709.43	32.61	3,600	**1,706.51**	24.21	43,200	**1706.51**	20.65	86,400
20c3sU9	1,708.84	35.99	3,600	1,708.84	30.67	43,200	1708.84	28.06	86,400
20c3sU10	1,181.31	10.75	3,600	1,181.31	0.00	26,813	1,181.31	0.00	29,140
AVG	**1,633.13**	**26.97**	**3,600**	**1,617.71**	**23.08**	**41,561**	**1,617.71**	**20.61**	**80,674**

Table 6:

INSTANCE	SERVICE	DETOUR	% DETOUR	s_0	s_1	s_2	s_3	#AFVs
20c3sU1	1,691.96	105.55	5.87	0	1	2	0	4
20c3sU2	1,573.85	0.97	0.06	0	0	0	1	6
20c3sU3	1,657.70	46.76	2.74	0	0	1	0	6
20c3sU4	1,449.78	32.22	2.17	0	1	1	0	5
20c3sU5	1,667.18	22.17	1.31	0	1	1	0	6
20c3sU6	1,618.61	0.03	0.00	0	1	0	0	6
20c3sU7	1,709.12	4.54	0.27	0	1	0	0	6
20c3sU8	1,678.06	28.45	1.67	0	1	1	1	6
20c3sU9	1,681.69	27.15	1.59	0	0	1	0	6
20c3sU10	1,172.70	8.61	0.73	0	0	0	1	6
AVG		**27.65**	**1.64%**					**6**

Table 7:

INSTANCE	SERVICE	DETOUR	% DETOUR	s_0	s_1	s_2	s_3	#AFVs
20c3sC1	1,167.69	5.89	0.50	0	0	1	1	4
20c3sC2	1,420.60	119.35	7.75	0	1	0	1	5
20c3sC3	879.90	0.31	0.04	0	0	0	1	3
20c3sC4	1,059.31	0.04	0.00	0	0	1	0	4
20c3sC5	2,156.01	139.50	6.47	0	0	1	3	7
20c3sC6	2,555.33	202.81	8.98	0	2	2	2	8
20c3sC7	1,332.00	61.99	4.45	0	2	0	2	4
20c3sC8	2,896.37	243.33	7.75	0	1	3	3	9
20c3sC9	1,666.91	133.04	7.39	0	0	1	0	6
20c3sC10	2,422.04	161.38	6.25	0	1	0	2	6
AVG		**106.76**	**4.76**					**6**

Table 8:

INSTANCE	SERVICE	DETOUR	% DETOUR	s_0	s_1	s_2	s_3	s_4	s_5	s_6	#AFVs
S1_2i6s	1,573.85	4.30	0.27	0	0	0	0	0	0	1	4
S1_4i6s	1,528.01	32.47	2.17	0	0	1	0	0	0	0	4
S1_6i6s	1,394.09	3.19	0.23	0	1	0	0	0	0	1	5
S1_8i6s	1,674.41	17.93	1.06	0	0	1	1	1	0	0	6
S1_10i6s	1,172.70	0.78	0.07	0	0	1	0	0	0	0	4
S2_2i6s	1,628.09	5.00	0.31	0	0	0	0	3	0	0	6
S2_4i6s	1,462.23	42.83	2.85	0	0	0	0	0	1	0	6
S2_6i6s	2,391.34	40.00	1.65	0	1	0	1	2	1	2	7
S2_8i6s	2,104.25	54.09	2.51	0	1	0	0	0	0	3	7
S2_10i6s	1,576.03	9.40	0.59	0	1	0	0	0	0	3	5
AVG		**21.00**	**1.16**								**6**

Table 9:

INSTANCE	SERVICE	DETOUR	% DETOUR	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	#AFVs
S1_4i2s	1582.21	0.01	0.00	0	1	0	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	6
S1_4i4s	1437.75	22.35	1.53	0	1	0	0	1	N/A	N/A	N/A	N/A	N/A	N/A	5
S1_4i6s	1,394.09	3.19	0.23	0	1	0	0	0	0	1	N/A	N/A	N/A	N/A	5
S1_4i8s	1,394.09	3.19	0.23	0	1	0	0	0	0	1	0	0	N/A	N/A	6
S1_4i10s	1,391.67	4.35	0.31	0	0	0	0	0	0	1	0	0	0	0	5
S2_4i2s	1,059.31	0.05	0.00	0	0	1	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	4
S2_4i4s	1,411.64	34.46	2.38	0	0	0	1	1	N/A	N/A	N/A	N/A	N/A	N/A	5
S2_4i6s	1,462.23	42.83	2.85	0	0	0	0	0	1	0	N/A	N/A	N/A	N/A	6
S2_4i8s	1,434.01	0.13	0.01	0	1	0	0	0	1	0	0	0	N/A	N/A	5
S2_4i10s	1,434.01	0.13	0.01	0	0	0	0	1	0	0	0	0	0	1	5
AVG		**11.07**	**0.76**												**5**

Table 10:

Instance	**UB**			**GAP (%)**			**CPU(s)**		
	EMH	KK	BMP2	EMH	KK	BMP2	EMH	KK	BMP2
Cloud-15-8-1	583.83	583.83	574.44	67.39	0.00	0.00	43,200	335.50	1,030.51
Cloud-15-8-2	574.28	574.28	569.41	65.76	0.00	0.00	43,200	315.60	727.09
Cloud-15-8-3	577.36	576.63	569.55	64.45	0.00	0.00	43,200	3,573.93	2,121.73
Cloud-15-8-4	610.32	610.32	603.33	60.34	0.00	0.00	43,200	207.15	343.83
Cloud-15-8-5	584.84	584.84	571.9	65.62	0.00	0.00	43,200	482.59	688.63
Cloud-15-8-6	604.07	603.67	594.02	59.64	0.00	0.00	43,200	248.47	263.18
Cloud-15-8-7	578.79	579.51	570.46	58.38	0.00	0.00	43,200	307.59	577.26
Cloud-15-8-8	600.25	601.6	589.64	60.98	0.00	0.00	43,200	525.53	1,387.85
Cloud-15-8-9	574.49	579.95	571.26	59.72	0.00	0.00	43,200	512.27	746.13
Cloud-15-8-10	620.78	623.63	615.4	58.11	0.00	0.00	43,200	325.28	368.21
AVG	**590.90**	**591.83**	**582.94**	**62.04**	**0.00**	**0.00**	**43,200**	**683.39**	**825.44**

Table 11:

INSTANCE	CPU (s)		
	NO VI NO DR	DR	VI+DR
Cloud-15-8-1	1030.51	509.96	424.08
Cloud-15-8-2	727.09	562.87	263.55
Cloud-15-8-3	2121.73	1,607.67	676.54
Cloud-15-8-4	343.83	289.11	304.92
Cloud-15-8-5	688.63	231.68	319.29
Cloud-15-8-6	263.18	147.72	190.12
Cloud-15-8-7	577.26	556.80	208.05
Cloud-15-8-8	1387.85	512.37	646.4
Cloud-15-8-9	746.13	327.66	301.38
Cloud-15-8-10	368.21	262.08	157.61
AVG	**825.44**	**500.79**	**349.19**

Table 12:

List of Table captions:

List of Keywords:

- Alternative Fuel Vehicle
- Consecutive Refueling
- Efficient Alternative Fuel Station
- Valid Inequalities