# Efficient solutions for the Green Vehicle Routing Problem with Capacitated Alternative Fuel Stations

M. Bruglieri<sup>1</sup>, D. Ferone<sup>2</sup>, P. Festa<sup>3</sup>, O. Pisacane<sup>4</sup>

<sup>1</sup> Dipartimento di Design, Politecnico di Milano, Italy

maurizio.bruglieri@polimi.it

 $^2\,$ Dipartimento di Ingegneria Meccanica, Energetica e Gestionale, Universitá della Calabria, Italy

danieleferone@gmail.com

<sup>3</sup> Dipartimento di Matematica e Applicazioni "R. Caccioppoli", Universitá degli Studi di Napoli

Federico II paola.festa@unina.it

<sup>4</sup> Dipartimento di Ingegneria dell'Informazione, Universitá Politecnica delle Marche, Italy o.pisacane@univpm.it

#### Abstract

In this paper, we propose a metaheuristic approach for efficiently solving the *Green Vehicle Routing Problem with Capacitated Alternative Fuel Stations* (G-VRP-CAFS). The G-VRP-CAFS, a variant of the traditional G-VRP, aims at routing a fleet of *Alternative Fuel Vehicles* (AFVs), based at a common depot, in order to serve a set of customers, minimizing the total travel distance. Due to the limited autonomy of the AFVs, some stops at *Alternative Fuel Stations* (AFSs) may be necessary during each trip. Unlike the G-VRP, in the G-VRP-CAFS, the AFS capacity, in terms of fueling pumps that are simultaneously available, is realistically assumed limited. For such a problem, we design an *Iterated Local Search* algorithm, in order to obtain good quality solutions in reasonable amount of time also on real-life alike case studies. Preliminary results, carried out on a set of benchmark instances taken from the literature, are promising.

**Keywords** : Metaheuristics, Iterated Local Search, Vehicle Routing Problem, Alternative Fuel Stations, Fueling Pump Reservation

# 1 Introduction and statement of the problem

The recent technological developments are making more and more the Alternative Fuel Vehicles (AFVs) competitive with regard to the traditional Internal Combustion Engine Vehicles (ICEVs). In particular, since the AFVs use alternative fuel (e.g., methane, hydrogen, electricity and so on), they are currently representing the right answer to the worries due to the climatic and environmental conditions. Moreover, because the transportation sector is responsible of about the 23% of the global CO<sub>2</sub>, several companies operating in the Logistics field are currently using AFVs for distributing goods and serving customers.

However, the AFVs usually require some stops during the trips for being refueled but currently, the Alternative Fuel Stations (AFSs) are not widespread across the territory. Therefore, the problem of properly routing them arises. This problem, introduced in the literature by the seminal work of [4], is called the Green Vehicle Routing Problem (G-VRP). It consists in routing a fleet of m AFVs, based at a common depot, in order to serve a set of customers, minimizing the total travel distance. Each route starts from the depot, serves a set of customers, with possibly stops at AFSs, and returns to the depot within a maximum duration  $T_{max}$ . The G-VRP is represented on a directed complete graph G = (N, A), where N represents the set of nodes including the set of customers I, the depot (denoted by 0) and the set of AFSs F. Instead, A denotes the set of arcs. For each pair of node  $(i, j) \in A$ , the travel distance  $d_{ij}$  is given and assuming an average speed v, the travel time  $t_{ij} = d_{ij}/v$  is also known. For each customer  $i \in I$ , the service time  $p_i$  is given as well as for each AFS  $s \in F$ , the constant time  $p_s$ to fully refuel a vehicle is known. Instead,  $p^{start}$  represents the time spent at the depot, before the route starts, when it is considered an AFS itself. For each AFV, the maximum fuel capacity Q is given and assuming a fuel consumption r linearly proportional to the travel distance, one can easily derive the maximum distance,  $D_{max}$ , an AFV can travel without stopping at any AFS, i.e.,  $D_{max} = Q/r$ .

In the original G-VRP, no limit on the AFS capacity is considered, i.e., an unlimited number of fueling pumps is implicitly assumed for each AFS. In order to overcome this unrealistic assumption, in [2], the G-VRP with Capacitated AFSs (G-VRP-CAFS) was introduced, assuming that  $\eta_s$  fueling pumps are available, in each AFS s. This means that only  $\eta_s$  AFVs can simultaneously refuel at each AFS s. The G-VRP-CAFS was addressed by the authors in both a public and a private scenario. In the public scenario, the AFSs are not owned by the transportation company and it is assumed that fueling pumps can be reserved in order to avoid queues. As a consequence, the pumps are available only on specific multiple time windows to take into account the reservations already made by the other AFVs. The authors mathematically formulated both the scenarios through an arc-based and a path-based model, the last solved also through cutting-plane based exact methods. In particular, the path-based formulations are inspired by that already proposed for the G-VRP by the same authors in [1]. A *path* is made up by a sequence of customers served either between the depot and a station, or a station and the depot or the depot and itself (i.e., a route without stops at stations). A path is *feasible* when its total travel distance and time do not overcome  $D_{max}$  and  $T_{max}$ . respectively. Due to these limitations, the number of feasible paths is somehow limited and can be exhaustively generated a-priori. Moreover, it was further limited by the authors introducing some *dominance rules*. Then, pairs of *compatible* paths are determined and given in input to the path-based model that selects those that belong to the optimal solution, i.e., that with the minimum total travel distance. However, when the number of customers to be served increases, the path-based approach can become impracticable since the number of pairs of compatible paths given to the model can become huge. Therefore, alternative solution approaches, such as metaheuristics, are necessary in order to address also real-life alike instances. For this reason, in this work, we propose an Iterated Local Search metaheuristic. In Section 2, the Iterated Local Search is presented, whereas in Section 3, some preliminary results are shown and discussed.

# 2 An Iterated Local Search for the G-VRP-CAFS

Iterated Local Search (ILS) can be considered a general stochastic local search method that aims at iteratively perturbing the current solution through a local search [6]. It consists of the following main blocks: an *initial solution generation* procedure that builds a feasible solution used as the starting point of the search; a *local search* procedure that improves locally the current solution and that is based on the definition of a suitable neighborhood on which the space of the solutions depends; a *perturbation* step that generates a new solution starting from the current one; an *acceptance criterion* according to which a new solution is considered the new starting point or not. The algorithm ends when a stopping criterion is met (typically, based on the maximum number of iterations without improvements).

Generation of the initial solution. In order to construct the initial solution, we use a variant of the well-known Clarke and Wright Savings (CWS) heuristic [3]. Generally, CWS begins with a solution made of all round-trip routes from the depot to each customer, and iteratively merges the two routes that result in the higher saving cost. Our CWS has been modified in order to manage the feasibility problems due to both maximum distance  $D_{max}$  and the shortage of free refuel pumps.

Indeed, when two routes  $r_1, r_2$  are selected for a merge, the procedure checks if the resulting

route r is feasible with respect to the maximum distance  $D_{max}$  without stopping at any AFS. In case r is feasible,  $r_1$  and  $r_2$  are merged and the resulting saving is given by  $d_{ij} - d_{i0} - d_{j0}$ , where i is the last customer visited by  $r_1$ , and j is the first customer visited in  $r_2$ .

On the contrary, if r is not feasible with respect to  $D_{max}$ , the procedure checks if there exists a position where the resulting route can be deviated to an available AFS.

For the sake of clarity, an example is reported in Figure 1. The routes in Figure 1(a) cannot be merged due to maximum travel distance, therefore the procedure verifies if it is possible to include the AFS 5 in all the positions (Figures 1(b–f)), checking that there are free pumps in that time window.



FIG. 1: Merging and AFS insertion example.

It is worth noting that, unlike the standard CWS where all merges result in a cost saving, some of these merging operations can lead to an increase of the current cost, but can still be profitable due to a reduction of the number of routes (i.e., used vehicles).

After calculating all the possible merging operations, the procedure sorts them respect to the saving cost, and randomly chooses among them with a skewed (or biased) distribution probability, in order to encourage the selection of the most favorable merges [5].

**Perturbation phase.** The perturbation phase performs three steps. The first step removes a percentage p of nodes from the current routes and for each of them builds a new round-trip route. In the second step, the procedure analyzes the routes from which a node has been removed and checks if there is an AFS visit that can be dropped, since the route is shorter. Finally, the CWS procedure is executed using the current routes as input and tries to merge them.

**Local search phase.** The local search removes a node v from a route r and inserts it into a feasible position of a route r' (r' can also be equal to r). The insertion in r' follows a similar scheme as in Figure 1. Indeed, when an insertion position is evaluated, if the resulting route is not feasible with respect to  $D_{max}$ , the procedure tries to fix the route planning a visit to an available AFS.

For each node, the local search follows a best improvement strategy. Therefore, given a customer v, it evaluates all the insertion positions and selects the one that results in the best cost improvement, if it exists. The local search iterates on all the customers until no further improvements are possible.

Acceptance criterion At the end of the local search, the new solution s' is compared with the base solution s. If s' corresponds to a better cost c(s') than c(s), then s' replaces s as base solution. Otherwise, in order to not being trapped in the same local optimum, a pejorative solution is accepted as base solution with a probability of  $e^{-RPD}$ , where RPD is the relative percentage difference given by  $RPD = \frac{c(s')-c(s)}{c(s)}$ . With this rule, the better is the solution, the higher is the acceptance probability.

# **3** Results and conclusions

In order to evaluate the proposed approach, a preliminary test has been performed on the benchmark of TRIANGLE instances used in [2]. The set is composed by 10 instances with 15 customers, 3 AFSs and 10 vehicles. Given their layout, the instances can be considered medium challenging for an exact solver (consideration confirmed by the experiments in [2]).

The algorithm has been implemented in Java and the tests have been executed on a INTEL i5-6400@2.70 GHz processor with 8GB of RAM. We used as stopping condition the reaching of 2000 iterations without improvement.

	CP-proactive		ILS		
	Total-distance	CPU	Total-distance	CPU	%-gap
Triangle1	1871.61	1.04	1871.61	1.80	0.00
Triangle2	2191.73	4.09	2258.48	1.94	3.05
Triangle3	1872.12	4.56	1872.12	2.12	0.00
Triangle4	1869.07	4.52	1963.83	1.80	5.07
Triangle5	1852.73	11.02	1972.42	2.03	6.46
Triangle6	1865.49	5.25	1876.49	1.87	0.59
Triangle7	1898.00	3.10	1922.16	1.88	1.27
Triangle8	2197.49	13.76	2321.31	2.04	5.63
Triangle9	1862.50	6.30	2005.62	1.90	7.68
Triangle10	1864.73	5.04	1975.26	1.97	5.93
Average	1934.55	5.87	2003.93	1.93	3.59

TAB. 1: Computational results

The results are reported in Table 1. The results of ILS are an average on 5 runs for each instance. The CP-proactive is the proactive cutting planes proposed in [2]. The results show how the ILS reaches near optimal solutions (with a gap of 3.59% on average) in shorter times. Despite these results are preliminary, they are also encouraging. In particular, we plan to improve the local search phase using a Variable Neighborhood Search.

### References

- M Bruglieri, S Mancini, F Pezzella, and O Pisacane. A path-based solution approach for the green vehicle routing problem. Computers & Operations Research, 103:109–122, 2019.
- [2] M Bruglieri, S Mancini, and O Pisacane. The green vehicle routing problem with capacitated alternative fuel stations. Computers & Operations Research, 112:104759, 2019.
- [3] G. Clarke and J. W. Wright. Scheduling of vehicles from a central depot to a number of delivery points. Operations Research, 12(4):568–581, 1964.
- [4] S Erdoğan and E Miller-Hooks. A green vehicle routing problem. Transportation Research Part E: Logistics and Transportation Review, 48(1):100–114, 2012.
- [5] Daniele Ferone, Sara Hatami, Eliana M. González-Neira, Angel A. Juan, and Paola Festa. A biased-randomized iterated local search for the distributed assembly permutation flowshop problem. *International Transactions in Operational Research*, 27(3):1368–1391, may 2020.
- [6] Helena R Lourenço, Olivier C Martin, and Thomas Stützle. Iterated local search. In Handbook of metaheuristics, pages 320–353. Springer, 2003.