RAIRO-Oper. Res. 52 (2018) 661–690
https://doi.org/10.1051/ro/2017072

RAIRO Operations Research
www.rairo-ro.org

HYBRID DATA MINING HEURISTICS FOR THE HETEROGENEOUS FLEET VEHICLE ROUTING PROBLEM

Marcelo Rodrigues de Holanda Maia[1,2,*], Alexandre Plastino[1] and Puca Huachi Vaz Penna[3]

Abstract. The vehicle routing problem consists of determining a set of routes for a fleet of vehicles to meet the demands of a given set of customers. The development and improvement of techniques for finding better solutions to this optimization problem have attracted considerable interest since such techniques can yield significant savings in transportation costs. The heterogeneous fleet vehicle routing problem is distinguished by the consideration of a heterogeneous fleet of vehicles, which is a very common scenario in real-world applications, rather than a homogeneous one. Hybrid versions of metaheuristics that incorporate data mining techniques have been applied to solve various optimization problems, with promising results. In this paper, we propose hybrid versions of a multi-start heuristic for the heterogeneous fleet vehicle routing problem based on the Iterated Local Search metaheuristic through the incorporation of data mining techniques. The results obtained in computational experiments show that the proposed hybrid heuristics demonstrate superior performance compared with the original heuristic, reaching better average solution costs with shorter run times.

Mathematics Subject Classification. 90B06, 90C27, 90C59.

Received December 19, 2016. Accepted October 5, 2017.

1. Introduction

The vehicle routing problem (VRP) is one of the most widely discussed problems in the fields of combinatorial optimization and operations research. It consists of determining a set of routes for a fleet of vehicles to meet the demands of a given set of customers. The development and improvement of techniques for finding better solutions to this problem have attracted considerable interest since such techniques can yield significant savings in transportation costs [45]. The heterogeneous fleet vehicle routing problem (HFVRP) is distinguished by the

Keywords. Hybrid metaheuristic, data mining, heterogeneous fleet vehicle routing problem.

[1] Instituto de Computação, Universidade Federal Fluminense, Avenida General Milton Tavares de Souza s/n, Niterói, RJ, Brazil, 24210–346.
[*] Corresponding author: mmaia@ic.uff.br
[2] Instituto Brasileiro de Geografia e Estatística, Rua General Canabarro 706, Rio de Janeiro, RJ, Brazil, 20271–205.
[3] Instituto de Ciências Exatas e Biológicas, Universidade Federal de Ouro Preto, Campus Universitário Morro do Cruzeiro, Ouro Preto, MG, Brazil, 35400–000.

Article published by EDP Sciences

consideration of a heterogeneous fleet of vehicles, which is a very common scenario in real-world applications [20], rather than a homogeneous one. Since this problem is NP-hard [24], the use of exact methods is infeasible for non-trivial instances.

In recent decades, various metaheuristics have been proposed for and successfully applied to optimization problems in various areas, allowing satisfactory solutions to be found within an acceptable computational time frame. Techniques from other areas have been incorporated into traditional metaheuristics to obtain even better results, giving rise to hybrid metaheuristics. One successful example of such hybridization is the Greedy Randomized Adaptive Search Procedure (GRASP) metaheuristic [11], into which data mining techniques have been incorporated.

The first proposal for the hybridization of GRASP with data mining techniques resulted in a hybrid heuristic called Data Mining GRASP (DM-GRASP) [37,38]. The basic idea of this hybrid heuristic is that patterns found in good solutions can be used to guide the search, leading to more effective exploitation of the solution space. In this hybrid version, after the execution of half of the GRASP iterations, a data mining procedure is applied to extract patterns from an elite set composed of the best solutions found up to that point. These patterns represent features found in the elite set of solutions and can be used to guide the search in the second half of the iterations.

DM-GRASP has been successfully applied to the set packing [37, 38], maximum diversity [41], server replication for reliable multicasting [40], p-median [34, 35], 2-path network design [4] and one-commodity pickup-and-delivery traveling salesman [17, 18] problems. A survey of this approach has been presented by Santos *et al.* [39].

Subsequently, a new version of the DM-GRASP heuristic, called Multi Data Mining GRASP (MDM-GRASP) [33], was proposed. The main idea underlying this version of the heuristic is to run the data mining procedure multiple times in an adaptive mode. The procedure is executed the first time the elite set becomes stable – as characterized by the absence of changes in the elite set over a given number of iterations – and, subsequently, each time the elite set changes and then becomes stable again.

MDM-GRASP has also been applied to the p-median [34, 35], 2-path network design [4], server replication for reliable multicasting [33] and one-commodity pickup-and-delivery traveling salesman [18] problems, and it has achieved better results than those obtained by DM-GRASP, not only in terms of solution quality but also in relation to computational time.

The promising results achieved in the various applications of DM-GRASP have inspired the similar incorporation of data mining techniques into a hybrid multi-start heuristic that combines elements from several traditional metaheuristics for application to the p-median problem [30], also with good results.

In this paper, we incorporate the data mining techniques used in DM-GRASP and MDM-GRASP into a multi-start ILS (MS-ILS) heuristic proposed by Penna *et al.* [31] for application to the HFVRP. Our goal is to study the impact of these data mining techniques on a state-of-the-art algorithm for solving the HFVRP.

The main contribution of this work is the proposal of highly competitive hybrid heuristics for solving the HFVRP, based on the incorporation of data mining techniques. The effectiveness of the proposed heuristics is demonstrated by the results of computational experiments conducted on a large number of HFVRP instances from widely used collections. The hybrid heuristics exhibit superior performance compared with the original version, achieving better average solution costs with shorter run times. Furthermore, new best solutions for six instances are found.

The remainder of this paper is organized as follows. Section 2 outlines the HFVRP and presents a review of major studies that have addressed this problem. Section 3 describes the process of incorporating data mining into the original heuristic and presents the resulting hybrid heuristics (DM-MS-ILS and MDM-MS-ILS). In Section 4, the computational results obtained in experiments using the proposed hybrid heuristics, DM-MS-ILS and MDM-MS-ILS, are analyzed and compared with those obtained using the original heuristic. Finally, Section 5 presents conclusions and prospects for future work.

2. The heterogeneous fleet vehicle routing problem

The VRP – which belongs to the NP-hard class of problems – consists of determining a set of routes for a fleet of vehicles to meet the demands of a given set of customers. Its formulation originates from real applications in the fields of logistics and transportation.

Efficient transportation has become increasingly important to society. Transportation costs represent a significant part of the total cost of a product. In addition, competition among transportation service providers and between entities of the industry and trade sectors leads to greater demands for rapid delivery to customers and cost savings. Recently, concerns about climate change and other environmental issues have become an additional factor motivating the search for more efficient transportation [20].

The heterogeneous fleet vehicle routing problem (HFVRP) is a generalization of the VRP in which customers are served by a heterogeneous fleet of vehicles, with different capacities and costs, instead of a homogeneous fleet. This situation is closer to those found in real-world applications, and thus, the HFVRP is a better model of such applications.

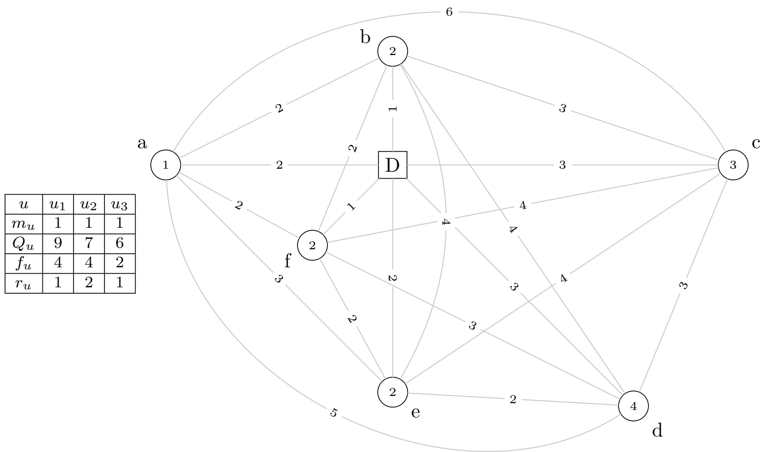
The HFVRP is defined as follows. Let $G = (V, A)$ be a directed graph, where $V = \{0, 1, \ldots, n\}$ is a set composed of $n + 1$ vertices and $A = \{(i, j) : i, j \in V, i \neq j\}$ is the set of arcs. Vertex 0 is the depot, where the vehicle fleet is located, whereas the set $V' = V \backslash \{0\}$ contains the remaining vertices representing the n customers. Each customer $i \in V'$ is associated with a non-negative demand q_i. The fleet consists of m different vehicle types, which compose a set $M = \{1, \ldots, m\}$. For each vehicle type $u \in M$, there are m_u vehicles available, each with a capacity Q_u and a fixed cost f_u. Finally, for each combination of an arc $(i, j) \in A$ and a vehicle type $u \in M$, there is an associated cost $c^u_{ij} = d_{ij} r_u$, where d_{ij} is the distance between vertices i and j and r_u is the dependent (variable) cost per unit distance associated with vehicle type u.

A route is defined by a pair (R, u), where $R = (i_1, i_2, \ldots, i_{|R|})$, $i_1 = i_{|R|} = 0$, and $\{i_2, \ldots, i_{|R|-1}\} \subseteq V'$; that is, each route is a circuit in G that begins and ends at the depot and is assigned to a vehicle of type $u \in M$. A route (R, u) is feasible if the sum of the demands of all customers on R does not exceed the capacity Q_u of the vehicle assigned to it. The cost of a route is the sum of the fixed cost of the assigned vehicle and the dependent costs associated with each of the traversed arcs in combination with the vehicle type. Thus, the goal of the HFVRP is to find feasible routes such that each customer is visited exactly once, the total number of routes assigned to vehicles of each type $u \in M$ does not exceed m_u, and the sum of all route costs is minimized.

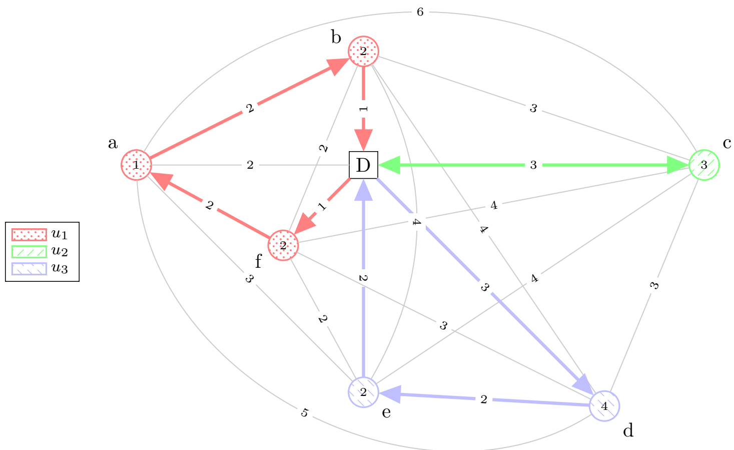
Figure 1a shows an example of an HFVRP instance, and Figure 1b presents a feasible solution for this instance. The instance is represented by a table describing the properties of the available fleet of vehicles and by a graph $G = (V, A)$. The fleet is composed of one vehicle of each of three types, u_1, u_2 and u_3. In the graph, the depot is represented by a square vertex, labeled D, and the customers are represented by circular vertices ($V' = \{$a, b, c, d, e, f$\}$). The demand of each customer is presented inside the corresponding vertex. In this example, the distances are symmetric – that is, $d_{ij} = d_{ji}, \forall i, j \in V$ – and therefore, each pair of symmetric arcs incident on a pair of vertices i and j is represented by one edge, which is labeled with a value corresponding to the distance d_{ij}. The solution presented consists of three routes: (R_1, u_1), (R_2, u_2) and (R_3, u_3), where $R_1 =$ (D, f, a, b, D), $R_2 =$ (D, c, D) and $R_3 =$ (D, d, e, D).

HFVRP instances are usually classified with respect to certain criteria. Two major classes are related to the limitations on the fleet. The problem that characterizes the first class, known as the Fleet Size and Mix (FSM) problem [15], can be regarded as a special case of the above definition in which $m_u = \infty, \forall u \in M$. Therefore, this problem consists of determining the best composition of the fleet as well as its best routing scheme. For the class of instances in which the fleet is limited, the corresponding problem is called the Heterogeneous Fixed Fleet VRP (HFFVRP) [43] and consists of optimizing the routing for an available fixed fleet.

The FSM and HFFVRP classes may be further subdivided with respect to the types of vehicle costs considered. The possibilities are as follows: both fixed and dependent costs (the general case), fixed costs only (a special case in which $r_u = 1, \forall u \in M$), or dependent costs only (a special case in which $f_u = 0, \forall u \in M$). Five subclasses of FSM and HFFVRP instances with respect to this criterion are differentiated in the literature: (1) the FSM problem with both fixed and dependent costs, denoted by FSM-FD [12]; (2) the FSM problem



(a) An HFVRP instance



(b) A feasible solution for the instance above

FIGURE 1. An example of an HFVRP instance and a feasible solution.

with fixed costs only, denoted by FSM-F [15]; (3) the FSM problem with dependent costs only, denoted by FSM-D [43]; (4) the HFFVRP with both fixed and dependent costs, denoted by HFFVRP-FD [25]; and (5) the HFFVRP with dependent costs only, denoted by HFFVRP-D [43]. To the best of our knowledge, the HFFVRP with fixed costs only has never been addressed.

Because of the practical importance of the HFVRP, it has been intensively studied in recent decades, and several solution strategies have been proposed. Some strategies address both the FSM problem and the HFFVRP, whereas others address only one of those classes. The main algorithms found in the literature are listed below.

For the FSM problem: a column generation algorithm based on a formulation of the HFVRP as a set covering problem, proposed by Choi and Tcha [7]; a hybrid genetic algorithm in which a local search is performed as a mutation operator, presented by Liu *et al.* [27]; a deterministic tabu search algorithm that uses different procedures to generate initial solutions, described by Brandão [5]; a Variable Neighborhood Search (VNS) heuristic that uses a procedure based on the sweep algorithm and Dijkstra's algorithm to obtain an initial solution and several neighboring structures in the local search phase, introduced by Imran *et al.* [21]; a memetic algorithm, presented by Prins [36]; an algorithm based on the Iterated Local Search (ILS) [28] metaheuristic that uses an RVND procedure in the local search phase, proposed by Penna *et al.* [31]; a hybrid version of the algorithm of Penna *et al.* [31] that includes a procedure based on a formulation of the set partitioning problem, proposed by Subramanian *et al.* [42]; and a genetic algorithm for several variants of the VRP based on unified genetic operators, diversification methods and local search, reported by Vidal *et al.* [46].

For the HFFVRP: a column generation algorithm based on an adaptive memory procedure that uses an embedded tabu search approach, presented by Taillard [43]; an algorithm based on the threshold accepting algorithm, a deterministic variant of the simulated annealing metaheuristic, described by Tarantilis *et al.* [44]; an algorithm based on the principle of first clustering customers and then determining routes, which also considers the possibility of renting vehicles when the fleet size is insufficient, introduced by Gencer *et al.* [13]; an algorithm based on another deterministic variant of simulated annealing called record-to-record travel, reported by Li *et al.* [25]; a memetic algorithm, described by Prins [36]; a population heuristic based on a genetic algorithm, presented by Liu [26]; an algorithm based on the ILS metaheuristic that uses an RVND procedure in the local search phase, proposed by Penna *et al.* [31]; and a hybrid version of the algorithm of Penna *et al.* [31] that includes a procedure based on a formulation of the set partitioning problem, proposed by Subramanian *et al.* [42].

An overview of the approaches presented in the literature for solving the HFVRP can be found in a survey by Baldacci *et al.* [2]. A literature survey of this problem that focuses on industrial aspects of the FSM problem has been presented by Hoff *et al.* [20]. A more recent literature survey on the HFVRP has been presented by Koç *et al.* [22]. The latter presents a computational comparison among state-of-the-art algorithms. It shows that the best performances for the FSM problem are achieved by the algorithms proposed by Choi and Tcha [7], Penna *et al.* [31] and Vidal *et al.* [46], whereas the best performances for the HFFVRP are achieved by the algorithms proposed by Li *et al.* [25], Liu [26], Penna *et al.* [31] and Subramanian *et al.* [42].

3. Incorporating data mining techniques into an MS-ILS Heuristic for the HFVRP

In this paper, we propose hybrid heuristics for solving the HFVRP by incorporating data mining techniques into the algorithm proposed by Penna *et al.* [31]. This algorithm is based on the ILS metaheuristic and uses a Random Variable Neighborhood Descent (RVND) procedure in the local search phase.

The algorithm by Penna *et al.* [31] was chosen as the basis for the proposed hybrid heuristics because it produces state-of-the-art results for both the FSM problem and the HFFVRP [22]. In addition, since it is a multi-start heuristic, it is well suited to the approaches previously applied for incorporating data mining into GRASP – which is also a multi-start method – to yield its hybrid versions DM-GRASP [39] and MDM-GRASP [33].

In a multi-start strategy, the search is restarted multiple times from new initial solutions. Multi-start heuristics [29] are iterative methods in which each iteration has two phases: generation and local search. In the generation phase, an initial solution is generated, whereas in the local search phase, the solution is typically (but not necessarily) improved. Each iteration produces a solution (usually a local optimum), and the best overall solution is output by the algorithm.

In the strategies that incorporate data mining techniques into GRASP – which can be generalized to the class of multi-start heuristics – an elite set is built that is composed of the best solutions found in all previous iterations. Once an interruption criterion (a number of iterations in DM-GRASP or the stabilization of the elite set in MDM-GRASP) has been satisfied, a data mining procedure is executed. It identifies patterns in the solutions of the elite set, which are then used in the generation phases of subsequent iterations to guide the local searches to start from more promising solutions, with the intent of obtaining even better solutions within a shorter convergence time.

The data mining procedure used in this approach is based on the formulation of the frequent itemset mining problem, which is one step of the association rule mining process [19]. This problem can be defined as follows.

Let $C = \{c_1, c_2, \ldots, c_\eta\}$ be the set of all items in the application domain. A transaction t is a subset of C, and a database D is a set of transactions. A frequent itemset F of D with support sup is a subset of C that occurs in at least $sup\%$ of the transactions in D. The frequent itemset mining problem consists of extracting from D all itemsets with support greater than or equal to some minimum support, which is specified as a parameter. In this context, the elite set is the database and each solution is a transaction.

More specifically, the adopted strategy is based on the mining of maximal frequent itemsets. A frequent itemset is called maximal if none of its immediate supersets is frequent. The FPmax* algorithm[4] [16] is used for mining maximal frequent itemsets in this approach.

The MS-ILS heuristic for the HFVRP proposed by Penna *et al.* [31] is described in Section 3.1. The hybrid heuristics that are proposed in this paper, DM-MS-ILS and MDM-MS-ILS, are presented in Sections 3.2 and 3.3, respectively.

3.1. MS-ILS Heuristic for the HFVRP

The steps of the MS-ILS[5] (Multi-Start Iterated Local Search) heuristic proposed by Penna *et al.* [31] for solving the HFVRP are presented in Algorithm 1.

Algorithm 1. MS-ILS($MaxIter, \beta$)

1: Initialize fleet
2: $n \leftarrow$ total number of customers
3: $v \leftarrow$ total number of vehicles
4: $f(s^*) \leftarrow \infty$
5: **for** $i \leftarrow 1$ **to** $MaxIter$ **do**
6:     $s \leftarrow$ Generate Initial Solution(v)
7:     $MaxIterILS \leftarrow n + \beta|s|$
8:     $s' \leftarrow$ ILS($s, MaxIterILS$)
9:     **if** $f(s') < f(s^*)$ **then**
10:         $s^* \leftarrow s'$
11:     **end if**
12: **end for**
13: **return** s^*

For the HFFVRP, the fleet is initialized with the available vehicles defined according to the problem instance being processed, whereas for the FSM problem, the fleet is initialized with one vehicle of each type (line 1). The number of customers is assigned as the value of n (line 2), and the number of vehicles is assigned to v (line 3). The multi-start heuristic is run for $MaxIter$ iterations (lines 5–12). In each iteration, an initial solution is generated via a constructive procedure (line 6). $MaxIterILS$ is the maximum allowed number of consecutive perturbations without improvements in the ILS heuristic. This value is computed (line 7) based on the number of customers (n), the number of routes ($|s|$) and a parameter β that defines the level of influence of the number of routes in this calculation ($MaxIterILS = n + \beta|s|$). In the local search phase, the ILS heuristic is used to improve the generated initial solution (line 8). This heuristic performs a local search from the initial solution and applies this search repeatedly to a set of solutions obtained by perturbing the locally optimal solutions found. If the solution s' returned by the ILS heuristic represents an improvement in cost, as given by the function f, then the best overall solution s^* is updated (lines 9-11). After the execution of the $MaxIter$ multi-start iterations, the best solution found is returned (line 13).

The pseudocode of the constructive procedure for generating the initial solutions is presented in Algorithm 2. Initially, the candidate list (CL) is filled with all customers (line 1), and the solution (s) is initialized with one

[4]Implementation available at http://fimi.cs.helsinki.fi
[5]The authors call this heuristic ILS-RVND in [31].

empty route associated with each vehicle in the fleet (line 2). Each route is filled with a customer k that is selected at random from CL (lines 3–6). An insertion criterion is randomly selected (line 7) from between the two available, namely, the Modified Cheapest Feasible Insertion Criterion (MCFIC) and the Nearest Feasible Insertion Criterion (NFIC). The first is a modification of the well-known cheapest insertion criterion in which only feasible insertions are allowed and an incentive is provided for serving customers located far from the depot. The second is an adaptation of the classical nearest insertion criterion in which only feasible insertions are allowed. Then, the parallel insertion strategy is used to complete the construction of the initial solution s (line 8). In this strategy, as long as CL is not empty, a customer from CL is selected to be inserted into a route using the insertion criterion chosen in the previous step, and all routes are considered to find the best insertion. If the solution s that is generated in this way is not feasible, then the construction procedure is restarted (lines 9–11). If the fleet is unlimited (FSM problem), then an empty route for each type of vehicle is added to solution s (lines 12–14). These empty routes are necessary to allow for a possible resizing of the fleet during the local search phase. Finally, the generated solution is returned (line 15).

Algorithm 2. Generate Initial Solution(v)

1: Initialize CL
2: Let $s = \{s^1, \ldots, s^v\}$ be a set composed of v empty routes
3: **for** $v' \leftarrow 1$ **to** v **do**
4:     $s^{v'} \leftarrow k \in CL$ selected at random
5:     $CL \leftarrow CL \backslash \{k\}$
6: **end for**
7: *Insertion Criterion* $\leftarrow c \in \{MCFIC, NFIC\}$ selected at random
8: $s \leftarrow$ Parallel Insertion(v, CL, *Insertion Criterion*)
9: **if** $\neg$feasible(s) **then**
10:     Go to line 1
11: **end if**
12: **if** unlimited fleet **then**
13:     Add to s an empty route for each type of vehicle
14: **end if**
15: **return** s

3.2. DM-MS-ILS Hybrid Heuristic

The first of the hybrid heuristics proposed in this paper, called Data Mining MS-ILS (DM-MS-ILS), has a structure similar to that used in the DM-GRASP [39] proposal. As explained previously in this section, the strategy used in this approach requires solutions to be represented as sets of items to allow the data mining procedure to identify patterns (maximal frequent itemsets) common to the best solutions. Several optimization problems have solutions that can be naturally represented as sets of items, and the approach used in this work has been applied mostly to such problems. However, for routing problems – such as the VRP and its variants – solutions are composed of sequences of items in which the order is important.

An application of DM-GRASP to the one-commodity pickup-and-delivery traveling salesman problem was proposed by Guerine *et al.* [17, 18]. Since this problem is a generalization of the traveling salesman problem, a solution consists of a route that serves all customers, respecting customer demand and vehicle capacity constraints. Naturally, the order is important to the route quality and even to its feasibility. A solution to this problem is typically represented as a list of customers, ordered according to the visitation sequence defined for the route.

In that work, the authors proposed an alternative representation to allow the use of the techniques employed in DM-GRASP. Each pair of consecutive customers i_r and i_{r+1}, $r = 1, 2, \ldots, |R| - 1$, on a route R is represented by an arc (i_r, i_{r+1}). Thus, the solution may be represented as a set of arcs while still preserving the order of the customers on the route. Consequently, each pattern found by the data mining procedure consists of a set of frequent arcs in the elite set of solutions. A pattern may contain pairs of adjacent arcs (i_r, i_{r+1}) and (i_{r+1}, i_{r+2}), which are connected, forming longer route segments. Therefore, a pattern can be seen as a (possibly disconnected) subgraph of the graph G formed of the depot and customers, where the connected components of this subgraph represent route segments in the pattern. Thus, each pattern is a nonempty set of route segments.

A solution to the HFVRP consists of a set of routes. However, using such a set representation of a solution as a basis for a frequent itemset mining problem formulation would not be appropriate, as it would result in each item being an entire route. Therefore, a decomposition of the solution with a finer granularity is needed.

In this work, a representation similar to that used by Guerine *et al.* [17, 18] is employed. As described in Section 2, each route in the HFVRP is represented by a pair (R, u), where $R = (i_1, i_2, \ldots, i_{|R|})$ is a list of vertices, ordered according to the defined visitation sequence, and u is the type of vehicle assigned to the route. In the adopted alternative representation, for each route (R, u), the list R is decomposed into a set of arcs $\{(i_1, i_2), (i_2, i_3), \ldots, (i_{|R|-1}, i_{|R|})\}$, as in the work carried by Guerine *et al.* [17, 18]. Then, the vehicle type u is assigned to each arc in the set, resulting in a set in which each element is a pair composed of an arc (i_r, i_{r+1}), $r = 1, 2 \ldots, |R| - 1$, in R and the vehicle type u, with the form $\{((i_1, i_2), u), ((i_2, i_3), u), \ldots, ((i_{|R|-1}, i_{|R|}), u)\}$. To belong to a pattern, an arc must be present and associated with the same vehicle type in a minimum number of solutions in the elite set. Consequently, in this case, the patterns mined are also formed of route segments, but each route segment has a vehicle type assigned to it.

Algorithm 3 presents the pseudocode of the hybrid heuristic DM-MS-ILS. The main difference between it and Algorithm 1 is that the multi-start iterations are divided into two blocks, each representing half of the total number of iterations. The first block (lines 6–14) corresponds to the elite set generation phase. The iterations in this phase are identical to those of the original heuristic, except for the updating of the elite set E, which stores the d best distinct solutions found (line 10), where d is a parameter of the algorithm. After the first block of multi-start iterations, data mining is performed on the elite set, returning a set P composed of the $MaxP$ largest patterns found with the minimal support $MinSup$ (line 15), where the size of a pattern is defined as the number of arcs it contains. The patterns in set P are arranged in decreasing order by size, forming a circular list. Then, the second phase, also called the hybrid phase (lines 16–24), begins. In each iteration in this phase, a pattern $p \in P$ is selected following the sequence of the circular list (line 17) and is used to generate an initial solution by means of an adapted constructive procedure (line 18). After generating the initial solution, the algorithm continues in the same way as the original heuristic.

The pseudocode of the constructive procedure used in the hybrid phase, which uses one of the mined patterns to generate an initial solution, is presented in Algorithm 4.

Again, the candidate list (CL) is filled with all customers (line 1), and the solution (s) is initialized with an empty route associated with each vehicle in the fleet (line 2). In fact, the only difference between this constructive procedure and the original one (Algorithm 2) is that it initializes the generated solution s by inserting route segments from pattern p, which is done in lines 3–16. In general terms, a complete route corresponding to each route segment $r' \in P$ is added to s. There are three possible forms of route segments: (1) a complete route, *i.e.*, a circuit in G that starts and ends at the depot; (2) a path in G that starts at the depot and ends at a customer vertex; and (3) a path in G that starts at a customer vertex and ends at the depot. A complete route corresponding to a route segment r' is obtained by calling the auxiliary function Complete(r'), which simply returns a clone of r', if r' is itself a complete route, or the route resulting from the insertion into r' of the single arc necessary to make it a complete route. The pattern insertion process is performed in slightly different ways for the FSM problem and the HFFVRP, as explained below.

For the FSM problem, since the fleet is unlimited, we can add any number of routes for each vehicle type to the solution without making it infeasible. In such a case, all customers in pattern p are removed from CL (line 4) and a complete route corresponding to each route segment $r' \in p$ is inserted into s as a new route (lines 5–7).

For the HFFVRP, since the fleet is fixed, if we add to the solution more routes associated with a particular vehicle type than the number of vehicles of that type that are available in the fleet, then the solution becomes

Algorithm 3. DM-MS-ILS($MaxIter, \beta, d, MaxP, MinSup$)

```
Initialize fleet
n ← total number of customers
v ← total number of vehicles
f(s*) ← ∞
E ← Ø
for i ← 1 to MaxIter/2 do
    s ← Generate Initial Solution (v)
    MaxIterILS ← n + β|s|
    s' ← ILS(s, MaxIterILS)
    Update EliteSet (E, s', d)
    if f(s') < f(s*) then
        s* ← s'
    end if
end for
P ← Mine (E, MaxP, MinSup)
for i ← MaxIter/2 + 1 to MaxIter do
    p ← Select Pattern(P)
    s ← Hybrid Generation(v, p)
    MaxIterILS ← n + β|s|
    s' ← ILS(s, MaxIterILS)
    if f(s') < f(s*) then
        s* ← s'
    end if
end for
return s*
```

infeasible. However, the number of route segments associated with a particular vehicle type in a pattern may exceed the specified limit on vehicles of that type. Therefore, pattern insertion is performed in the following way. For each route segment $r' \in P$, the algorithm searches for an empty route s associated with the same vehicle type as r' (line 10). The call to the auxiliary function Vehicle(r') returns the vehicle type associated with r', whereas the call to the auxiliary function Find Empty Route (s, Vehicle(r')) returns the index of an empty route in s associated with the same vehicle type as r', if any exists, or zero otherwise. If there is such a route, then it is replaced with a complete route corresponding to r' (line 12), and all customers in r' are removed from CL (line 13). Otherwise, r' is not inserted into the solution, but its customers remain in CL so that they will be inserted into the solution later (through the original strategy).

Figure 2 illustrates an example of the use of a mined pattern to generate an initial solution for the HFVRP instance presented in Figure 1(a) as described in Algorithm 4. Figure 2(a) presents the mined pattern used, which is composed of two route segments: (R'_1, u_1) and (R'_2, u_2), where $R'_1 =$ (f, a, b, D) and $R'_2 =$ (D, c).

Initially, a complete route corresponding to each route segment in the pattern is inserted into the solution. To complete the route segments, we must add the necessary arcs to close the circuits such that they start and end at the depot. In this example, the arc (D, f) is added to the path associated with vehicle type u_1 and the arc (c, D) is added to the path associated with vehicle type u_2. The state of the initial solution after the introduction of the pattern is shown in Figure 2b.

After this step, the candidate list contains the customers that have not yet been added to the solution. Those customers are added to the solution following the strategy of the original heuristic. A customer (e) is randomly selected to be added to the route associated with vehicle type u_3, which was previously empty, bringing the solution into the state shown in Figure 2c. Finally, the only remaining customer in the candidate list (d) is added to the route associated with u_2, completing the construction of the initial solution. Figure 2d presents the generated initial solution.

Algorithm 4. Hybrid Generation(v, p)

1: Initialize CL
2: Let $s = \{s^1, \ldots, s^v\}$ be a set composed of v empty routes
3: **if** unlimited fleet **then**
4:     Remove all customers in p from CL
5:     **for all** $r' \in p$ **do**
6:         $s \leftarrow s \cup \{\text{Complete}(r')\}$
7:     **end for**
8: **else**
9:     **for all** $r' \in p$ **do**
10:         $v' \leftarrow$ Find Empty Route(s, Vehicle(r'))
11:         **if** $v' \neq 0$ **then**
12:             $s^{v'} \leftarrow \text{Complete}(r')$
13:             Remove all customers in r' from CL
14:         **end if**
15:     **end for**
16: **end if**
17: **for** $v' \leftarrow 1$ **to** v **do**
18:     $s^{v'} \leftarrow k \in CL$ selected at random
19:     $CL \leftarrow CL \backslash \{k\}$
20: **end for**
21: *Insertion Criterion* $\leftarrow c \in \{MCFIC, NFIC\}$ selected at random
22: $s \leftarrow$ Parallel Insertion(v, CL, *Insertion Criterion*)
23: **if** $\neg$feasible(s) **then**
24:     Go to line 1
25: **end if**
26: **if** unlimited fleet **then**
27:     Add to s an empty route for each type of vehicle
28: **end if**
29: **return** s

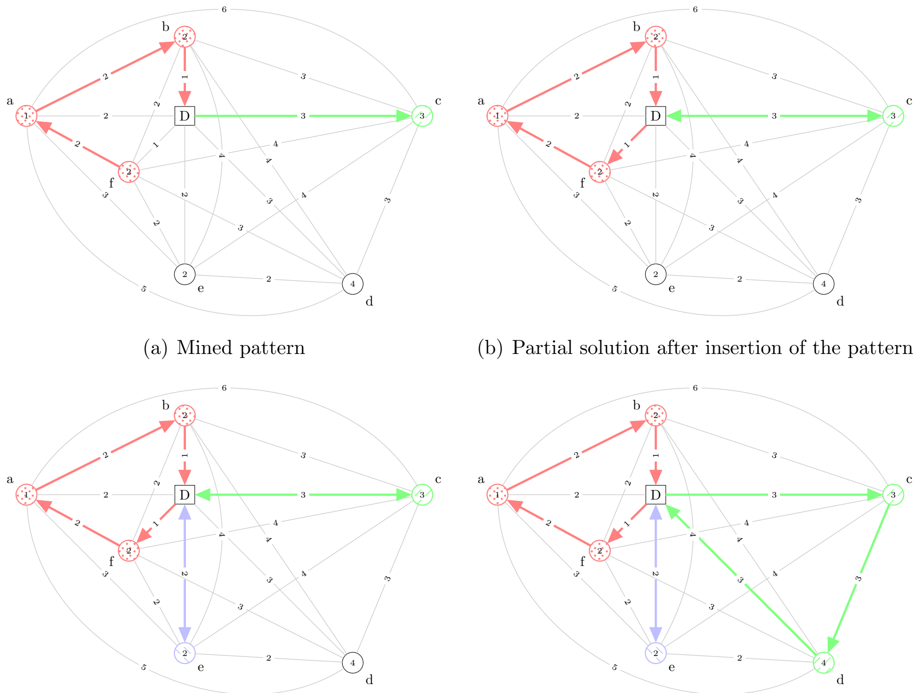
3.3. MDM-MS-ILS hybrid heuristic

The second hybrid heuristic proposed in this paper is called Multi Data Mining MS-ILS (MDM-MS-ILS). Its pseudocode is presented in Algorithm 5. This heuristic has a structure similar to that used in the MDM-GRASP [33] proposal.

In this case, whenever the elite set E becomes stable – that is, when it remains unmodified over ϕ iterations – data mining is performed, which updates the pattern set P (lines 8-10). In the early iterations, when data mining has not yet been performed and the pattern set P is consequently still empty, the initial solutions are generated by Algorithm 2 (line 12). Once data mining has been performed, the initial solutions start being generated by Algorithm 4 using a pattern p selected from the current pattern set P (lines 14–15). After the local search phase (line 18), the elite set is updated (line 19), as is the best solution in the case of improvement (lines 20–22).

4. Computational experiments

In this section, we describe the computational experiments performed in this work. Since our goal is to study the impact of data mining techniques on a state-of-the-art algorithm for solving the HFVRP, we evaluate the hybrid heuristics DM-MS-ILS and MDM-MS-ILS (presented in Sect. 3) and compare them with the MS-ILS heuristic proposed by Penna *et al.* [31], on which they are based. Additionally, our results are compared with the best known solutions reported in the literature.



(a) Mined pattern (b) Partial solution after insertion of the pattern

(c) Partial solution at an intermediate stage (d) Initial solution generated

FIGURE 2. An example of initial solution generation using a mined pattern.

The hybrid heuristics proposed in this work were implemented based on the original source code for the MS-ILS heuristic, which was provided by the authors. The code was written in C++ and compiled with the GCC C++ compiler (g++) version 4.8.2 in the Cygwin environment. The experiments were run on a computer equipped with an Intel® Core$^{\text{TM}}$ i5 3.20 GHz CPU and 1.56 GB of RAM running 32–bit Windows 7 Pro SP1.

Experiments were performed for both classes of HFVRP: the FSM problem and the HFFVRP. Each heuristic was run ten times, with ten different random number seeds, for each instance. In each case, the following values obtained over the ten runs are reported: best solution cost, average solution cost, standard deviation of the costs, average computational time (in seconds), and standard deviation of the computational times.

The parameters were defined as follows. The number of iterations $MaxIter$ of the multi-start heuristic was set to 100. For the parameter β, which controls the level of influence of the number of vehicles v in the calculation of $MaxIterILS$ ($MaxIterILS = n + \beta v$), we adopted a value of 1 (the default value in the MS-ILS implementation). For the parameters associated with the data mining procedure, we adopted values used in other applications of this approach [17, 18, 30, 34, 37, 39, 40]: the maximum size of the elite set, the minimum support, and the number of patterns used were set to 10, 20% and 10, respectively. Finally, the number of iterations without changes to the elite set required to consider it stable was set to 5% of $MaxIter$.

The results for the FSM problem are presented in Section 4.1, whereas Section 4.2 presents the results for the HFFVRP. Two tables are presented for each set of instances. The first one shows a cost comparison and includes, for each instance, the best known solution (BKS) cost reported in the literature and the results obtained using the compared heuristics. The other one shows a computational time comparison among the heuristics. Each table contains two additional rows at the bottom: one presenting the global average – *i.e.*, the mean of the average costs or times reported in the table – for each method and another showing the average percentage

Algorithm 5. MDM-MS-ILS($MaxIter, \beta, d, MaxP, MinSup, \phi$)

Initialize fleet
$n \leftarrow$ total number of customers
$v \leftarrow$ total number of vehicles
$f(s^*) \leftarrow \infty$
$E \leftarrow \emptyset$
$P \leftarrow \emptyset$
for $i \leftarrow 1$ **to** $MaxIter$ **do**
  if Stable(E, ϕ) **then**
    $P \leftarrow$ Mine($E, MaxP, MinSup$)
  end if
  if $P = \emptyset$ **then**
    $s \leftarrow$ Generate Initial Solution(v)
  else
    $p \leftarrow$ Select Pattern(P)
    $s \leftarrow$ Hybrid Generation(v, p)
  end if
  $MaxIterILS \leftarrow n + \beta|s|$
  $s' \leftarrow$ ILS($s, MaxIterILS$)
  Update EliteSet(E, s', d)
  if $f(s') < f(s^*)$ **then**
    $s^* \leftarrow s'$
  end if
end for
return s^*

difference achieved by the new heuristics – *i.e.*, the mean of the percentage differences in average cost or time for each of the new heuristics with respect to the MS-ILS heuristic.

In the comparisons, the best values among all three heuristics are shown in boldface. Furthermore, when a new best solution is found (*i.e.*, a solution with a cost lower than the previous best known solution cost), it is underlined. At the end of each of the following sections, we present an analysis assessing the statistical significance of the differences in the average costs achieved by the compared heuristics. For this analysis, we have used a one-tailed paired Student's t test per instance for each pair of heuristics, with a significance level of 5%.

Finally, we report the results of experiments performed to further investigate the behavior of the compared heuristics in Section 4.3.

4.1. FSM problem

In the experiments addressing the FSM problem, we used the same instances considered in the experiments performed by Penna *et al.* [31], which were described by Taillard [43], and an additional set of instances described by Brandão [6]. Taillard's set contains 12 instances in which the number of customers ranges from 20 to 100, whereas Brandão's contains five instances, each with between 100 and 199 customers. In these experiments, only dependent costs were considered; *i.e.*, all instances were regarded as FSM-D instances.

Tables 1 and 2 present the comparisons of the results for Taillard's set of instances [43]. We observe that the MDM-MS-ILS heuristic outperformed the others, achieving the best average solution costs for nine of the 12 instances, whereas DM-MS-ILS achieved the best average costs for four instances, and MS-ILS, for three. MDM-MS-ILS and DM-MS-ILS found optimal solutions for ten instances, and MS-ILS, for nine[6]. MDM-MS-ILS had the best average times for seven instances, DM-MS-ILS had the best average times for five instances (with

[6]All best known solutions for these instances have been proven to be optimal by Baldacci and Mingozzi [3].

TABLE 1. Cost comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – Taillard's instances [43] (FSM-D).

Inst.	BKS	MS-ILS			DM-MS-ILS			MDM-MS-ILS		
		Best Cost	Avg. Cost	Std. Dev.	Best Cost	Avg. Cost	Std. Dev.	Best Cost	Avg. Cost	Std. Dev.
3	623.22[a]	**623.22**	624.04	1.73	**623.22**	624.09	1.83	**623.22**	**623.66**	1.39
4	387.18[a]	**387.18**	**387.80**	0.81	**387.18**	388.12	0.81	**387.18**	388.12	0.81
5	742.87[a]	**742.87**	743.32	1.42	**742.87**	743.32	1.42	**742.87**	**742.87**	0.00
6	415.03[a]	**415.03**	**415.03**	0.00	**415.03**	**415.03**	0.00	**415.03**	**415.03**	0.00
13	1491.86[b]	1495.43	1500.94	4.79	**1491.86**	1500.28	5.05	**1491.86**	**1496.25**	2.64
14	603.21[c]	**603.21**	**603.21**	0.00	**603.21**	**603.21**	0.00	**603.21**	**603.21**	0.00
15	999.82[b]	**999.82**	1003.25	2.43	**999.82**	1001.95	2.82	**999.82**	**1001.42**	2.66
16	1131.00[d]	**1131.00**	1135.91	2.33	**1131.00**	1134.10	2.98	**1131.00**	**1132.66**	2.91
17	1038.60[a]	**1038.60**	1043.18	3.78	**1038.60**	1042.96	4.83	**1038.60**	**1040.81**	3.31
18	1800.80[e]	1803.32	1814.31	7.71	**1801.40**	**1812.01**	7.98	1803.32	1816.55	8.38
19	1105.44[b]	**1105.44**	1112.25	4.74	**1105.44**	1110.99	4.16	**1105.44**	**1109.38**	4.88
20	1530.43[a]	1543.09	1549.20	4.31	1542.70	**1547.86**	3.29	**1541.46**	1548.52	4.63
Global Avg.			994.37			993.66			**993.21**	
Avg. P.D.						–0.05%			**–0.10%**	

[a]BKS first reported by Choi and Tcha [7], [b]BKS first reported by Gendreau *et al.* [14], [c]BKS first reported by Taillard [43], [d]BKS first reported by Wassan and Osman [47], [e]BKS first reported by Baldacci and Mingozzi [3].

TABLE 2. Time comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – Taillard's instances [43] (FSM-D).

Inst.	MS-ILS		DM-MS-ILS		MDM-MS-ILS	
	Avg. Time (s)	Std. Dev.	Avg. Time (s)	Std. Dev.	Avg. Time (s)	Std. Dev.
3	2.50	0.09	2.43	0.13	**2.40**	0.10
4	2.50	0.09	**2.44**	0.10	2.47	0.13
5	3.06	0.14	**2.97**	0.09	**2.97**	0.08
6	2.91	0.15	**2.86**	0.12	2.89	0.09
13	20.81	1.13	**19.59**	0.82	19.71	1.23
14	22.89	0.76	20.74	0.89	**19.45**	1.14
15	19.67	1.21	17.08	0.56	**16.87**	0.70
16	17.66	0.96	17.19	0.68	**16.98**	0.93
17	73.49	2.83	**69.07**	2.19	74.24	5.01
18	**80.78**	3.90	80.90	4.77	83.46	6.01
19	165.65	7.04	161.26	10.70	**154.97**	5.07
20	151.08	5.57	142.22	6.86	**131.51**	8.67
Global Avg.	46.92		44.89		**43.99**	
Avg. P.D.			–4.79%		**–5.49%**	

a tie between MDM-MS-ILS and DM-MS-ILS for one instance), and MS-ILS had the best average time for only one instance.

Tables 3 and 4 present the comparisons of the results for Brandão's instances [6]. Again, the superior performance of the MDM-MS-ILS heuristic compared with the others is evident. It achieved the best average costs for all instances in this set. Furthermore, MDM-MS-ILS obtained the best average times for three instances, whereas DM-MS-ILS had the best average times for the remaining two instances.

Below, we present the results of the statistical significance analysis for the FSM experiments.

Table 3. Cost comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – Brandão's instances [6] (FSM-D).

Inst.	BKS	**MS-ILS** Best Cost	Avg. Cost	Std. Dev.	**DM-MS-ILS** Best Cost	Avg. Cost	Std. Dev.	**MDM-MS-ILS** Best Cost	Avg. Cost	Std. Dev.
N1	2212.77[a]	**2236.86**	2253.57	7.93	2243.76	2252.87	5.27	2245.38	**2252.25**	6.87
N2	2823.75[a]	2850.07	2871.72	12.03	**2838.35**	2853.09	8.93	2839.96	**2852.90**	9.05
N3	2234.57[b]	2269.96	2316.61	30.13	**2236.09**	2282.18	32.99	2238.25	**2272.86**	29.68
N4	1822.78[b]	**1822.78**	1834.27	6.14	1823.04	1831.13	6.09	1825.82	**1828.05**	2.84
N5	2016.79[b]	**2030.03**	2043.79	10.53	**2030.03**	2041.84	10.71	2031.41	**2040.23**	6.79
Global Avg.			2263.99			2252.22			**2249.26**	
Avg. P.D.						-0.49%			**–0.63%**	

[a]BKS first reported by Subramanian *et al.* [42], [b]BKS first reported by Brandão [6].

Table 4. Time comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – Brandão's instances [6] (FSM-D).

Inst.	**MS-ILS** Avg. Time (s)	Std. Dev.	**DM-MS-ILS** Avg. Time (s)	Std. Dev.	**MDM-MS-ILS** Avg. Time (s)	Std. Dev.
N1	465.23	21.16	464.37	20.24	**462.37**	32.71
N2	1016.68	41.90	**1008.30**	41.13	1029.91	49.60
N3	366.85	14.76	**362.39**	23.95	365.76	17.43
N4	146.25	8.01	142.57	7.46	**141.76**	8.79
N5	626.54	27.51	577.19	28.23	**560.57**	27.14
Global Avg.	524.31		**510.96**		512.07	
Avg. P.D.			–2.60%		**–2.77%**	

Between MS-ILS and DM-MS-ILS, the differences in the average costs are statistically significant for three of Brandão's instances (N2, N3 and N4), all with 100 customers or more. DM-MS-ILS has the advantage for all three of these instances.

Between MS-ILS and MDM-MS-ILS, the differences are statistically significant for the same three of Brandão's instances (N2, N3 and N4) and for two of Taillard's instances (13 and 16), both with fewer than 100 customers. MDM-MS-ILS has the advantage for all five of these instances.

Finally, between DM-MS-ILS and MDM-MS-ILS, there is a statistically significant difference only for instance 13 from Taillard's set, in favor of MDM-MS-ILS.

4.2. HFFVRP

In the experiments addressing the HFFVRP, we used the instances described by Brandão [6] (the same used in the FSM experiments, but with limited fleets), the instances described by Li *et al.* [25] and the instances described by Duhamel *et al.* [9]. The set from Li *et al.* contains five instances with 200 to 360 customers. The 96 instances described by Duhamel *et al.* are divided into four sets: Set 1, which contains 15 instances, each with fewer than 100 customers; Set 2, which contains 38 instances, each with between 100 and 150 customers; Set 3, which contains 31 instances, each with between 151 and 200 customers; and Set 4, which contains 12 instances, each with more than 200 customers. In the experiments using the instances from Brandão and Li *et al.*, only dependent costs were considered; *i.e.*, these instances were regarded as HFFVRP-D instances. In the experiments using the instances from Duhamel *et al.*, both fixed and dependent costs were considered; *i.e.*, these instances were regarded as HFFVRP-FD instances.

TABLE 5. Cost comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – Brandão's instances [6] (HFFVRP-D).

Inst.	BKS	MS-ILS			DM-MS-ILS			MDM-MS-ILS		
		Best Cost	Avg. Cost	Std. Dev.	Best Cost	Avg. Cost	Std. Dev.	Best Cost	Avg. Cost	Std. Dev.
N1	2235.87[a]	**2246.39**	2261.41	8.01	**2246.39**	**2258.65**	7.64	**2246.39**	**2258.65**	7.01
N2	2864.83[a]	2898.17	2911.52	6.81	2897.41	2910.72	7.61	**2891.17**	**2902.85**	9.32
N3	2378.99[a]	**2378.99**	2383.85	3.97	**2378.99**	2384.26	3.84	**2378.99**	**2383.10**	4.10
N4	1839.22[b]	**1839.22**	1839.23	0.03	**1839.22**	**1839.22**	0.00	**1839.22**	1839.42	0.63
N5	2047.81[a]	**2047.81**	**2047.81**	0.00	**2047.81**	**2047.81**	0.00	**2047.81**	**2047.81**	0.00
Global Avg.			2288.77			2288.13			**2286.36**	
Avg. P.D.						–0.03%			**–0.09%**	

[a]BKS first reported by Subramanian *et al.* [42], [b]BKS first reported by Brandão [6]

TABLE 6. Time comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – Brandão's instances [6] (HFFVRP-D).

Inst.	MS-ILS		DM-MS-ILS		MDM-MS-ILS	
	Avg. Time (s)	Std. Dev.	Avg. Time (s)	Std. Dev.	Avg. Time (s)	Std. Dev.
N1	633.57	31.87	620.09	17.87	**605.16**	26.05
N2	1382.33	80.83	1370.50	75.25	**1327.38**	79.02
N3	439.14	21.75	**405.40**	19.76	405.42	26.87
N4	197.03	8.76	176.89	8.87	**168.55**	9.60
N5	744.32	52.15	647.88	24.58	**621.34**	39.13
Global Avg.	679.28		644.15		**625.57**	
Avg. P.D.			–7.12%		**–10.04%**	

Tables 5 and 6 present the comparisons of the results for Brandão's instances [6]. The MDM-MS-ILS heuristic outperformed the others, achieving the best average costs for four of the five instances, whereas DM-MS-ILS achieved the best average costs for three instances, and MS-ILS, for only one instance. Furthermore, MDM-MS-ILS obtained the best average times for four instances, whereas DM-MS-ILS had the best average time for the remaining instance. All heuristics found the best known solutions for three instances (N3, N4 and N5).

Tables 7 and 8 present the comparisons of the results for the instances from Li *et al.* [25]. Again, the MDM-MS-ILS heuristic outperformed the others, achieving the best average costs for four of the five instances, whereas MS-ILS achieved the best average cost for the remaining instance. MDM-MS-ILS had the best average times for two instances, and DM-MS-ILS had the best average times for the other three instances.

Tables 9 and 10 present the comparisons of the results for the instances from Set 1 of Duhamel *et al.* [9]. MDM-MS-ILS achieved the best average costs for eight instances; DM-MS-ILS, for four instances; and MS-ILS, for eight instances. DM-ILS-MS and MS-ILS found the best known solutions for six instances (10, 55, 75, 92, 93 and 94), and MDM-MS-ILS, for nine (08, 10, 43, 55, 75, 82, 92, 93 and 94). Regarding the average times, MDM-MS-ILS obtained the best values for all instances.

Tables 11 and 12 present the comparisons of the results for the instances from Set 2 of Duhamel *et al.* [9]. The MDM-MS-ILS heuristic outperformed the others, achieving the best average costs for 27 of the 38 instances, whereas DM-MS-ILS achieved the best average costs for 10 instances, and MS-ILS, for only one instance. Two new best solutions were found by MDM-MS-ILS (for instances 41 and 48), and another one was found by DM-MS-ILS (for instance 89). Furthermore, MDM-MS-ILS found the best known solutions for five other instances (12, 16, 2A, 53 and 87), whereas DM-MS-ILS found the best known solutions for four instances (12, 16, 2A

TABLE 7. Cost comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – instances from Li *et al.* [25] (HFFVRP-D).

Inst.	BKS	MS-ILS			DM-MS-ILS			MDM-MS-ILS		
		Best Cost	Avg. Cost	Std. Dev.	Best Cost	Avg. Cost	Std. Dev.	Best Cost	Avg. Cost	Std. Dev.
H1	12 050.08[a]	**12 050.39**	12 073.99	18.68	**12 050.39**	12 062.22	7.19	**12 050.39**	**12 058.78**	5.83
H2	10 208.32[a]	10 364.71	10 410.56	31.94	**10 342.83**	10 384.47	27.30	10 351.15	**10 378.21**	20.25
H3	16 223.39[a]	16 270.47	16 358.84	67.02	16 270.47	16 305.38	25.67	**16 239.48**	**16 280.42**	32.87
H4	17 458.65[a]	**17 681.84**	**17 882.54**	93.29	17 720.66	17 901.57	76.72	17 828.15	17 921.21	77.25
H5	23 166.56[a]	23 770.01	24 010.71	155.92	23 651.90	23 960.35	165.91	**23 640.07**	**23 897.62**	137.02
Global Avg.			16 147.33			16 122.80			**16 107.25**	
Avg. P.D.						–0.16%			**–0.23%**	

[a]BKS first reported by Brandão [6].

TABLE 8. Time comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – instances from Li *et al.* [25] (HFFVRP-D).

Inst.	MS-ILS		DM-MS-ILS		MDM-MS-ILS	
	Avg. Time (s)	Std. Dev.	Avg. Time (s)	Std. Dev.	Avg. Time (s)	Std. Dev.
H1	2330.32	137.24	2034.71	89.02	**1953.79**	95.15
H2	4074.28	256.40	**4015.73**	224.74	4141.99	191.95
H3	7659.96	234.09	7046.92	303.14	**6815.66**	435.75
H4	13 343.36	546.35	**13 058.05**	576.18	13 063.40	569.61
H5	24 568.82	1113.21	**24 269.38**	1333.37	24 656.98	1803.24
Global Avg.	10395.35		**10 084.96**		10 126.37	
Avg. P.D.			–5.34%		**–5.87%**	

and 87), as did MS-ILS (12, 16, 53 and 87). MDM-MS-ILS had the best average times for 33 instances, and DM-MS-ILS had the best average times for the remaining five instances.

Tables 13 and 14 present the comparisons of the results for the instances from Set 3 of Duhamel *et al.* [9]. The MDM-MS-ILS heuristic outperformed the others, achieving the best average costs for 18 of the 31 instances, whereas DM-MS-ILS achieved the best average costs for 10 instances, and MS-ILS, for the three remaining instances. A new best solution for instance 67 was found by all heuristics, a new best solution for instance 04 was found by MDM-MS-ILS, and a new best solution for instance 02 was found by DM-MS-ILS. MDM-MS-ILS had the best average times for 25 instances, and DM-MS-ILS had the best average times for the remaining six instances.

Tables 15 and 16 present the comparisons of the results for the instances from Set 4 of Duhamel *et al.* [9]. The MDM-MS-ILS heuristic outperformed the others, achieving the best average costs for six of the 12 instances, whereas DM-MS-ILS obtained the best average costs for five instances, and MS-ILS, for one instance. MDM-MS-ILS has the best average times for five instances, and DM-MS-ILS had the best average times for the remaining seven instances.

Below, we present the results of the statistical significance analysis for the HFFVRP experiments.

Between MS-ILS and DM-MS-ILS, the differences in the average costs are statistically significant for three of the instances from Li *et al.* (H1, H2 and H3) and for 26 of the instances from Duhamel *et al.* – four from Set 1 (08, 10, 11 and 94), 14 from Set 2 (05, 07, 26, 30, 31, 40, 48, 73, 74, 79, 81, 84, 88 and 89), six from Set 3

TABLE 9. Cost comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – Set 1 of Duhamel *et al.* [9] (HFFVRP-FD).

Inst.	BKS	MS-ILS Best Cost	MS-ILS Avg. Cost	MS-ILS Std. Dev.	DM-MS-ILS Best Cost	DM-MS-ILS Avg. Cost	DM-MS-ILS Std. Dev.	MDM-MS-ILS Best Cost	MDM-MS-ILS Avg. Cost	MDM-MS-ILS Std. Dev.
01	9180.45[a]	**9210.14**	**9213.14**	2.84	**9210.14**	9213.88	2.62	**9210.14**	9213.26	3.50
08	4591.75[b]	4594.07	4598.79	2.82	4596.52	4596.98	0.97	**4591.75**	**4595.76**	2.20
10	2107.55[c]	**2107.55**	2107.93	0.68	**2107.55**	**2107.61**	0.17	**2107.55**	**2107.61**	0.17
11	3367.41[b]	3372.16	3376.80	3.33	**3368.50**	**3374.19**	3.20	3370.52	3375.04	2.75
36	5684.61[b]	5707.98	5721.72	9.04	**5685.17**	5723.33	20.81	5688.63	**5720.36**	17.63
39	2921.36[d]	**2932.57**	**2934.47**	0.81	2933.88	2934.57	0.37	2934.55	2935.11	1.01
43	8737.02[b]	8747.16	8767.78	16.57	8747.16	8761.64	11.21	**8737.02**	**8751.83**	13.19
52	4027.27[b]	**4029.42**	**4029.42**	0.00	**4029.42**	4030.04	1.95	**4029.42**	4030.04	1.95
55	10 244.34[c]	**10 244.34**	**10 253.64**	8.97	**10 244.34**	10255.36	9.00	**10244.34**	10254.56	8.80
70	6684.56[d]	6692.91	6705.39	8.56	**6685.24**	6701.09	11.91	**6685.24**	**6695.96**	11.61
75	452.85[c]	**452.85**	**452.85**	0.00	**452.85**	**452.85**	0.00	**452.85**	**452.85**	0.00
82	4766.74[b]	4768.21	4773.28	1.97	4772.94	4773.82	0.87	**4766.74**	**4771.77**	2.74
92	564.39[c]	**564.39**	**564.39**	0.00	**564.39**	**564.39**	0.00	**564.39**	**564.39**	0.00
93	1036.99[e]	**1036.99**	**1037.92**	1.39	**1036.99**	1038.04	1.35	**1036.99**	1038.34	1.42
94	1378.25[c]	**1378.25**	**1378.37**	0.26	**1378.25**	1378.52	0.30	**1378.25**	1378.40	0.25
Global Avg.			4394.39			4393.75			**4392.35**	
Avg. P.D.						–0.01%			**–0.03%**	

[a]BKS first reported by Kochetov and Khmelev [23], [b]BKS first reported by Duhamel *et al.* [8],
[c]BKS first reported by Duhamel *et al.* [9], [d]BKS first reported by Penna *et al.* [32],
[e]BKS first reported by Duhamel *et al.* [10].

TABLE 10. Time comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – Set 1 of Duhamel *et al.* [9] (HFFVRP-FD).

Inst.	MS-ILS Avg. Time (s)	MS-ILS Std. Dev.	DM-MS-ILS Avg. Time (s)	DM-MS-ILS Std. Dev.	MDM-MS-ILS Avg. Time (s)	MDM-MS-ILS Std. Dev.
01	290.77	14.43	256.70	19.58	**238.45**	21.29
08	92.92	2.52	86.33	4.01	**81.85**	3.30
10	170.43	6.21	151.82	8.80	**145.13**	3.44
11	258.90	11.83	237.65	10.34	**236.76**	6.58
36	322.18	11.99	322.54	13.26	**312.95**	12.81
39	258.12	7.03	248.03	9.48	**241.77**	9.49
43	213.85	8.50	198.99	8.33	**190.33**	7.82
52	92.61	6.27	74.53	2.94	**69.21**	4.04
55	43.95	1.65	37.41	1.05	**37.28**	1.47
70	137.00	3.41	126.53	10.03	**119.16**	3.48
75	4.30	0.17	3.76	0.15	**3.49**	0.09
82	111.60	3.83	103.77	3.14	**100.93**	4.65
92	33.31	1.27	30.65	0.96	**30.23**	1.25
93	25.53	0.81	22.05	1.26	**19.66**	1.03
94	56.21	2.26	52.52	1.44	**51.47**	2.30
Global Avg.	140.78		130.22		**125.24**	
Avg. P.D.			–9.80%		**–14.22%**	

TABLE 11. Cost comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – Set 2 of Duhamel *et al.* [9] (HFFVRP-FD).

Inst.	BKS	MS-ILS			DM-MS-ILS			MDM-MS-ILS		
		Best Cost	Avg. Cost	Std. Dev.	Best Cost	Avg. Cost	Std. Dev.	Best Cost	Avg. Cost	Std. Dev.
03	10 727.36[a]	10 791.91	10 826.46	20.30	107 67.72	10 811.24	23.35	**10 754.86**	**10 807.45**	35.45
05	10 896.33[b]	10 931.20	11 003.17	37.71	10 912.38	**10 953.91**	33.58	**10 903.63**	10 965.80	51.61
06	11 692.85[c]	11 791.83	11 852.25	39.20	11 784.11	11 841.67	36.86	**11 753.01**	**11 816.43**	36.32
07	8074.64[a]	81 42.96	8199.10	28.83	8136.26	8177.43	35.18	**8121.77**	**8161.76**	25.72
12	3543.99[d]	**3543.99**	3547.78	3.73	**3543.99**	3547.61	4.11	**3543.99**	**3545.32**	1.41
13	6696.43[c]	6716.37	6725.96	6.99	6714.52	6723.22	12.01	**6710.07**	**6717.21**	5.66
16	4156.97[c]	**4156.97**	4166.56	7.37	**4156.97**	4163.25	4.02	**4156.97**	**4162.80**	4.52
17	5362.83[c]	5368.67	5397.62	15.71	5368.67	5395.98	16.11	**5368.38**	**5390.25**	20.21
2A	7793.16[c]	7797.32	7829.59	17.98	**7793.16**	**7817.67**	18.83	**7793.16**	7820.47	16.09
2B	8464.69[c]	**8481.56**	8531.41	31.36	8487.25	8522.33	27.19	8498.61	**8518.70**	16.57
21	5139.84[b]	**5144.08**	5168.12	20.41	5145.44	5161.73	16.22	5145.44	**5159.28**	13.95
25	7206.64[c]	7250.72	7270.95	17.45	**7209.29**	7255.38	20.95	**7209.29**	**7248.31**	18.75
26	6393.47[a]	**6455.28**	**6462.94**	5.83	6456.29	6464.81	5.52	6456.29	6463.28	5.54
28	5531.06[c]	5537.23	5546.90	5.27	5535.08	5545.51	4.58	**5533.47**	**5543.17**	6.85
30	6313.39[c]	**6321.69**	6350.53	12.89	**6321.69**	**6344.74**	11.42	**6321.69**	6344.92	11.91
31	4091.52[c]	4113.70	4132.58	13.12	**4101.03**	**4124.52**	16.28	4121.04	4132.83	8.71
34	5758.09[c]	5789.29	5831.22	32.62	5801.82	5823.31	15.00	**5780.07**	**5816.02**	19.37
40	11 122.32[b]	11 172.52	11 192.74	13.75	**11 127.02**	11 165.79	24.95	11 132.57	**11 157.41**	17.17
41	7572.07[c]	7682.04	7746.73	41.01	7667.53	7728.94	46.17	**7571.44**	**7699.18**	62.17
47	16 175.22[b]	16 238.39	16 303.32	36.89	16 249.94	16 302.89	30.68	**16 224.78**	**16296.63**	42.16
48	21 316.55[a]	21 384.37	21 473.01	75.02	21 314.81	21 411.13	51.37	**21 287.90**	**21 401.80**	71.36
51	7721.47[c]	**7780.04**	7798.71	18.02	**7780.04**	7804.28	18.42	7783.38	**7790.61**	10.54
53	6434.83[c]	**6434.83**	6454.04	18.41	6435.24	6453.01	12.35	**6434.83**	**6448.14**	14.26
60	17 037.23[b]	17 065.77	17 100.66	29.85	17 075.68	17 105.74	18.53	**17 060.89**	**17 092.11**	19.49
61	7292.03[b]	**7294.77**	7304.77	7.57	7296.72	**7304.63**	7.06	7296.72	7306.39	8.54
66	12 790.56[a]	**12829.08**	12 895.66	45.42	**12 829.08**	**128 82.07**	37.55	**12 829.08**	12 890.90	34.19
68	8935.89[b]	8984.15	9089.52	67.97	8984.15	9082.97	64.22	**8982.72**	**9063.66**	73.32
73	10 195.13[a]	**10 208.72**	10 221.27	7.57	**10 208.72**	**10 213.99**	2.70	**10 208.72**	10 218.65	5.48
74	11 586.87[c]	11 602.43	11 628.75	16.21	**11 600.15**	**11 618.14**	10.53	11 608.06	11 621.81	12.69
79	7259.54[c]	7277.79	7321.96	21.82	7277.79	7307.41	19.61	**7265.58**	**7303.76**	21.80
81	10 675.92[a]	10 697.08	10 713.49	9.11	10 694.37	**10 705.97**	8.08	**10 691.45**	10 707.37	14.75
83	10 019.15[c]	10 053.78	10 065.58	11.85	10 049.81	10 059.47	7.22	**10 039.66**	**10 052.49**	6.02
84	7227.88[c]	7240.82	7271.50	12.90	**7237.41**	7260.49	13.14	7239.22	**7258.67**	16.00
85	8779.76[c]	8825.78	8873.34	26.45	**8820.46**	**8859.99**	28.51	8845.37	8879.04	15.50
87	3753.87[d]	**3753.87**	3758.01	7.62	**3753.87**	3761.85	12.84	**3753.87**	**3754.31**	1.39
88	12 402.85[c]	12 501.24	12 555.96	63.88	**12 417.00**	12 507.39	48.14	12 454.45	**12 492.58**	19.32
89	7099.68[a]	7110.97	7146.11	15.85	**7098.18**	7129.06	21.31	7108.28	**7124.87**	13.94
90	2346.13[c]	2349.24	2359.46	4.10	2354.95	2358.71	1.80	**2347.16**	**2357.12**	4.44
Global Avg.			8634.68			8623.64			**8619.25**	
Avg. P.D.						–0.11%			**–0.17%**	

[a]BKS first reported by Kochetov and Khmelev [23], [b]BKS first reported by Penna *et al.* [32], [c]BKS first reported by Duhamel *et al.* [8], [d]BKS first reported by Duhamel *et al.* [9].

TABLE 12. Time comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – Set 2 of Duhamel *et al.* [9] (HFFVRP-FD).

Inst.	**MS-ILS** Avg. Time (s)	**MS-ILS** Std. Dev.	**DM-MS-ILS** Avg. Time (s)	**DM-MS-ILS** Std. Dev.	**MDM-MS-ILS** Avg. Time (s)	**MDM-MS-ILS** Std. Dev.
03	636.38	25.25	612.57	33.11	**595.05**	22.97
05	226.37	6.42	220.07	7.64	**207.62**	11.22
06	341.35	12.67	328.95	7.71	**324.08**	9.25
07	234.19	10.47	226.05	7.97	**219.76**	10.03
12	520.95	25.60	497.63	17.20	**475.18**	14.38
13	550.66	16.92	534.87	21.92	**528.91**	22.44
16	799.13	18.53	758.76	21.81	**756.53**	23.49
17	237.64	10.33	230.54	5.63	**224.24**	11.19
2A	441.91	10.41	426.46	17.03	**418.39**	11.72
2B	572.26	21.51	560.60	19.45	**548.65**	18.39
21	676.47	27.37	658.29	32.16	**638.00**	15.83
25	1253.55	48.65	1221.47	60.45	**1187.99**	33.52
26	1292.52	46.24	1294.54	63.74	**1242.66**	56.65
28	644.69	27.32	629.75	23.29	**628.04**	19.25
30	719.16	27.93	684.94	20.39	**670.42**	31.57
31	1114.05	54.31	**1071.48**	50.36	1090.02	42.92
34	753.37	24.06	**734.24**	24.57	741.77	30.31
40	645.40	26.79	**613.73**	23.28	618.21	50.92
41	782.60	48.07	**767.90**	44.21	769.72	31.41
47	398.20	27.08	376.94	25.27	**373.04**	22.96
48	509.18	17.77	476.10	22.74	**452.95**	32.38
51	828.77	29.38	791.39	40.46	**786.62**	25.52
53	527.24	13.59	508.86	28.03	**490.91**	14.54
60	611.45	25.57	577.73	31.03	**565.97**	27.01
61	779.92	33.27	752.36	39.15	**717.30**	58.97
66	1465.33	51.75	**1417.25**	63.31	1429.20	55.40
68	671.87	33.08	649.68	24.60	**638.34**	29.88
73	464.41	18.00	428.60	15.22	**413.73**	20.10
74	440.63	18.93	419.77	19.55	**411.45**	16.81
79	1450.56	52.76	1423.95	83.46	**1386.96**	69.78
81	395.20	6.77	380.45	10.08	**378.09**	14.36
83	514.12	14.96	484.04	23.87	**470.33**	25.19
84	285.05	16.13	269.45	15.56	**261.06**	17.98
85	655.82	31.67	640.44	44.54	**616.50**	30.89
87	405.91	11.49	372.38	12.56	**366.48**	21.64
88	315.21	7.18	300.19	13.37	**291.20**	10.86
89	600.53	16.14	585.23	19.79	**584.20**	22.22
90	333.68	10.10	309.86	12.88	**301.13**	19.03
Global Avg.	634.10		611.51		**600.54**	
Avg. P.D.			–4.03%		**–6.11%**	

(14, 15, 35, 45, 54 and 76) and two from Set 4 (32 and 65). DM-MS-ILS has the advantage for 28 of these instances, whereas MS-ILS is favored for only one.

Between MS-ILS and MDM-MS-ILS, the differences are statistically significant for two of Brandão's instances (N1 and N2), four of the instances from Li *et al.* (H1, H2, H3 and H5) and 38 of the instances from Duhamel *et al.* – four from Set 1 (08, 11, 43 and 70), 15 from Set 2 (05, 06, 07, 12, 13, 25, 30, 40, 41, 48, 79, 83, 84, 88

TABLE 13. Cost comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – Set 3 of Duhamel *et al.* [9] (HFFVRP-FD).

Inst.	BKS	MS-ILS			DM-MS-ILS			MDM-MS-ILS		
		Best Cost	Avg. Cost	Std. Dev.	Best Cost	Avg. Cost	Std. Dev.	Best Cost	Avg. Cost	Std. Dev.
02	11 718.86[a]	11 733.14	11 775.17	39.05	**11 703.66**	**11 752.48**	32.62	11 713.88	11 753.96	37.76
04	10 787.03[a]	10 828.33	10 847.27	14.33	10 797.99	10 855.80	28.67	**10 784.61**	**10 823.68**	17.73
09	7619.19[b]	**7636.84**	7675.45	17.23	**7636.84**	**7669.14**	24.98	**7636.84**	7672.33	20.06
14	5644.92[b]	5701.91	5718.68	12.19	**5668.07**	5701.21	19.37	5671.37	**5699.90**	18.08
15	8236.40[b]	8293.26	8318.48	15.78	8273.45	8305.74	16.64	**8264.03**	**8302.24**	20.86
24	9101.47[b]	**9173.59**	9242.72	33.40	**9173.59**	**9240.60**	33.88	**9173.59**	9247.17	35.63
29	9142.86[a]	**9144.78**	**9167.39**	13.55	**9144.78**	9168.07	10.24	**9144.78**	9169.33	12.17
33	9419.00[c]	9493.56	9521.65	17.26	9473.57	9512.85	31.54	**9441.81**	**9502.70**	28.07
35	9574.71[b]	9633.65	9665.27	23.85	9615.85	9649.87	20.86	**9604.04**	**9636.32**	20.98
37	6858.23[b]	6886.29	6903.65	10.72	6886.29	6900.98	9.92	**6878.59**	**6895.29**	10.18
42	10 855.73[a]	11 008.86	11 094.78	40.20	**10 955.01**	**11 058.33**	57.37	10 981.64	11 059.00	67.17
44	12 197.46[b]	12 274.61	12 346.08	55.02	**12 241.70**	**12 327.10**	53.88	12 261.45	12 358.11	52.22
45	10 484.23[b]	10 561.05	10 660.83	57.14	**10 520.84**	**10 606.11**	53.87	10 550.31	10 608.36	39.97
50	12 374.04[b]	12 409.45	12 476.66	39.84	12 409.55	12 480.50	33.06	**12 395.11**	**12 472.87**	45.36
54	10 370.09[a]	10 401.18	10 480.14	41.02	10 401.18	10 465.53	33.72	**10 387.47**	**10 453.46**	42.87
56	31 090.53[a]	31 263.88	31 337.08	59.93	31 195.39	31 311.72	59.02	**31 129.72**	**31 256.17**	72.49
57	44 818.18[b]	45 012.74	45 087.62	56.56	44 963.29	45 059.01	71.75	**44 898.59**	**45 052.81**	82.22
59	14 282.59[b]	14 371.32	14 413.20	31.13	14 352.60	14 393.36	40.52	**14 328.25**	**14 376.27**	29.78
63	19 951.76[b]	**19 974.11**	20 333.24	152.36	20 173.19	20 287.24	80.12	20 173.19	**20 261.42**	57.05
64	17 135.16[a]	17 155.00	17 175.62	16.73	17 154.96	17 175.20	15.50	**17 136.39**	**17 164.60**	14.91
67	10 915.60[a]	**10 850.16**	10 979.48	54.17	**10 850.16**	**10 965.14**	43.78	**10 850.16**	10 980.20	58.93
69	9162.78[b]	9212.99	9264.78	28.33	**9195.71**	9261.74	35.97	9206.22	**9236.71**	20.17
71	9870.22[b]	9947.38	9984.87	34.78	9918.92	9978.84	44.78	**9912.60**	**9963.08**	26.44
72	5883.33[a]	5957.32	5981.85	12.81	5957.32	5975.75	12.63	**5939.91**	**5967.48**	12.33
76	12 007.57[c]	12 073.74	12 124.36	31.71	**12 040.13**	**12 093.53**	34.56	12 055.19	12 101.97	22.16
77	6929.67[a]	**6957.61**	7013.94	31.86	6984.57	7011.54	22.04	6962.09	**6997.63**	24.99
78	7035.01[b]	**7135.30**	**7149.32**	9.98	7138.00	7150.64	8.63	7137.86	7149.40	9.78
80	6816.89[b]	6834.69	6849.51	8.31	**6826.81**	**6849.03**	10.49	6837.84	6849.47	8.35
86	9030.68[b]	**9058.78**	9075.67	10.58	9059.59	**9073.55**	11.37	9061.45	9074.48	10.70
91	6374.01[c]	**6380.87**	6420.53	16.05	6402.86	6422.71	12.76	6396.21	**6418.84**	11.26
95	6175.62[a]	**6236.95**	**6243.86**	5.02	6237.53	6243.88	4.92	6237.31	6244.21	5.09
Global Avg.			11 784.81			11 772.49			**11 766.11**	
Avg. P.D.						–0.10%			**–0.15%**	

[a]BKS first reported by Penna *et al.* [32], [b]BKS first reported by Duhamel *et al.* [8],
[c]BKS first reported by Kochetov and Khmelev [23].

TABLE 14. Time comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – Set 3 of Duhamel *et al.* [9] (HFFVRP-FD).

Inst.	**MS-ILS**		**DM-MS-ILS**		**MDM-MS-ILS**	
	Avg. Time (s)	Std. Dev.	Avg. Time (s)	Std. Dev.	Avg. Time (s)	Std. Dev.
02	1440.85	54.36	1363.59	36.85	**1344.83**	87.21
04	1385.93	81.61	**1351.59**	78.52	1358.64	101.29
09	843.19	47.66	818.16	39.82	**799.24**	46.46
14	2122.12	118.51	**2033.79**	149.85	2034.79	125.47
15	1416.58	46.22	1398.99	46.05	**1347.05**	70.71
24	1616.92	69.40	1568.10	69.82	**1553.03**	66.34
29	1909.24	57.90	1864.62	51.57	**1830.00**	78.31
33	1939.18	84.98	1875.66	73.17	**1868.95**	78.25
35	836.18	24.47	804.86	23.77	**786.79**	27.86
37	1243.23	70.83	1249.63	82.92	**1230.06**	50.15
42	3741.81	145.83	3710.33	145.16	**3552.28**	137.10
44	3321.39	135.24	3277.85	203.43	**3220.46**	120.86
45	2162.92	86.33	2075.02	108.65	**2006.37**	93.85
50	5417.19	321.70	5305.08	269.49	**5291.74**	197.25
54	2789.87	131.03	2696.98	131.08	**2655.06**	107.30
56	1092.90	70.12	1090.92	58.49	**1075.61**	50.12
57	1288.42	71.03	1238.64	65.05	**1224.30**	52.49
59	3943.32	118.75	3813.79	117.36	**3641.57**	172.77
63	1599.57	43.32	1604.42	79.62	**1557.71**	44.82
64	2063.66	94.10	1919.59	87.86	**1822.52**	74.64
67	2179.39	93.05	2121.16	114.25	**2093.24**	167.30
69	873.47	22.74	**845.72**	27.39	858.37	32.71
71	1246.78	42.62	**1224.96**	70.41	1230.48	49.59
72	2194.97	70.30	**2187.25**	59.92	2215.37	77.44
76	863.27	14.98	822.05	26.32	**792.30**	26.12
77	3947.87	154.86	**3878.65**	170.08	3946.85	104.34
78	3387.32	67.47	3562.57	75.90	**3379.75**	57.10
80	1929.63	77.73	1950.01	75.99	**1876.03**	54.27
86	1022.39	27.25	976.17	34.55	**959.10**	40.76
91	1875.25	77.28	1804.52	60.16	**1801.47**	110.03
95	1396.00	164.16	1353.51	74.08	**1328.73**	51.85
Global Avg.	2035.19		1993.17		**1957.50**	
Avg. P.D.			–2.42%		**–4.15%**	

and 89), 15 from Set 3 (04, 14, 15, 33, 35, 37, 45, 54, 56, 59, 64, 69, 72, 76 and 77) and four from Set 4 (32, 38, 58 and 65). MDM-MS-ILS has the advantage for all of these 44 instances.

Finally, between DM-MS-ILS and MDM-MS-ILS, there are statistically significant differences for instance N2 from Brandão's set, for instance H3 from Li *et al.* and for 12 of the instances from Duhamel *et al.* – three from Set 1 (39, 43 and 82), seven from Set 2 (06, 12, 51, 73, 83, 85 and 87) and two from Set 3 (04 and 56). MDM-MS-ILS has the advantage for 11 of these instances, whereas DM-MS-ILS is favored for the other three.

4.3. Behavior analysis

To further investigate the behavior of the heuristics, additional experiments were performed using FSM-D instance N4 from Brandão's set and HFFVRP-FD instance 02 from Set 3 of Duhamel *et al.*

In the first of these experiments, the heuristics were run with 1000 iterations. Figures 3 and 4 present an analysis of the solution costs obtained during the experiment for instance N4 from Brandão and instance 02

TABLE 15. Cost comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – Set 4 of Duhamel *et al.* [9] (HFFVRP-FD).

Inst.	BKS	MS-ILS Best Cost	MS-ILS Avg. Cost	MS-ILS Std. Dev.	DM-MS-ILS Best Cost	DM-MS-ILS Avg. Cost	DM-MS-ILS Std. Dev.	MDM-MS-ILS Best Cost	MDM-MS-ILS Avg. Cost	MDM-MS-ILS Std. Dev.
18	9668.17[a]	9669.61	9713.14	20.32	**9669.27**	**9705.47**	17.29	9677.03	9706.76	22.65
19	11 702.77[b]	11 783.19	11 808.25	18.71	**11 774.79**	**11 797.94**	17.03	11 782.19	11 807.30	15.04
22	13 068.03[b]	**13 128.84**	**13 164.07**	27.20	13 137.59	13 182.04	29.31	13 132.48	13 171.46	28.99
23	7750.27[b]	7810.31	7832.72	13.66	7805.56	7826.80	16.96	**7789.30**	**7824.32**	22.40
27	8417.62[c]	8454.54	8471.00	10.33	**8433.37**	**8465.91**	16.53	8454.54	8472.42	10.18
32	9378.30[c]	9497.46	9527.22	22.56	**9413.12**	**9500.37**	38.91	9465.39	9508.99	22.60
38	11 217.53[a]	11 282.72	11 311.42	23.47	11 257.82	11 297.80	19.74	**11 241.61**	**11 283.28**	24.85
46	24 428.54[a]	24 612.01	24 756.86	80.17	24 612.01	24 722.43	74.30	**24 593.25**	**24 713.63**	76.04
49	16 219.41[a]	16 340.16	16 423.62	57.93	16 296.28	**16 386.15**	51.77	**16 260.83**	16 401.84	59.06
58	23 397.76[b]	23 687.00	23 794.68	99.94	**23 596.00**	23 756.18	92.20	23 661.47	**23 702.72**	25.24
62	22 952.06[a]	**23 037.04**	23 243.62	130.86	23 120.23	23 216.92	82.49	23 042.89	**23 177.49**	87.28
65	13 013.89[a]	13 064.86	13 085.03	17.23	13 057.42	13 076.46	13.87	**13 026.62**	**13 063.28**	16.73
Global Avg.			14 427.64			14 411.20			**14 402.79**	
Avg. P.D.						–0.11%			**–0.14%**	

[a]BKS first reported by Penna *et al.* [32], [b]BKS first reported by Duhamel *et al.* [8],
[c]BKS first reported by Kochetov and Khmelev [23].

TABLE 16. Time comparison of MS-ILS, DM-MS-ILS and MDM-MS-ILS – Set 4 of Duhamel *et al.* [9] (HFFVRP-FD).

Inst.	MS-ILS Avg. Time (s)	MS-ILS Std. Dev.	DM-MS-ILS Avg. Time (s)	DM-MS-ILS Std. Dev.	MDM-MS-ILS Avg. Time (s)	MDM-MS-ILS Std. Dev.
18	4823.40	198.59	4749.34	318.76	**4670.61**	116.16
19	3127.09	124.53	**3070.05**	121.88	3091.52	120.75
22	3046.88	138.35	**2922.02**	125.46	2940.41	132.36
23	2235.00	112.76	**2163.28**	98.49	2182.00	61.51
27	3162.39	238.48	3048.85	222.60	**3024.37**	134.47
32	4215.96	204.05	4069.08	219.29	**4009.94**	192.26
38	2491.13	129.90	**2389.68**	59.35	2427.41	142.54
46	7493.94	406.03	**7321.39**	299.55	7363.81	295.16
49	8106.02	474.86	**7882.29**	402.17	7894.89	475.40
58	3029.16	50.02	**2940.62**	94.09	2946.24	167.81
62	4019.55	186.59	3968.45	180.80	**3895.87**	159.51
65	5434.10	216.84	5251.93	160.65	**5234.71**	252.90
Global Avg.	4265.38		4148.08		**4140.15**	
Avg. P.D.			–2.91%		**–3.03%**	

from Duhamel *et al.*, respectively. In each figure, the three charts in the first row show, for each heuristic, the solution costs obtained per iteration in the generation and local search phases, whereas the second row presents enlarged views focusing on the local search phase costs. In the charts for the hybrid heuristics, the label **DM** is used to denote the iterations preceding a data mining procedure run.

There is a consistent pattern in the behavior exhibited by the heuristics for both instances. The charts show that all heuristics exhibit the same behavior until the first run of the data mining procedure. In DM-MS-ILS,

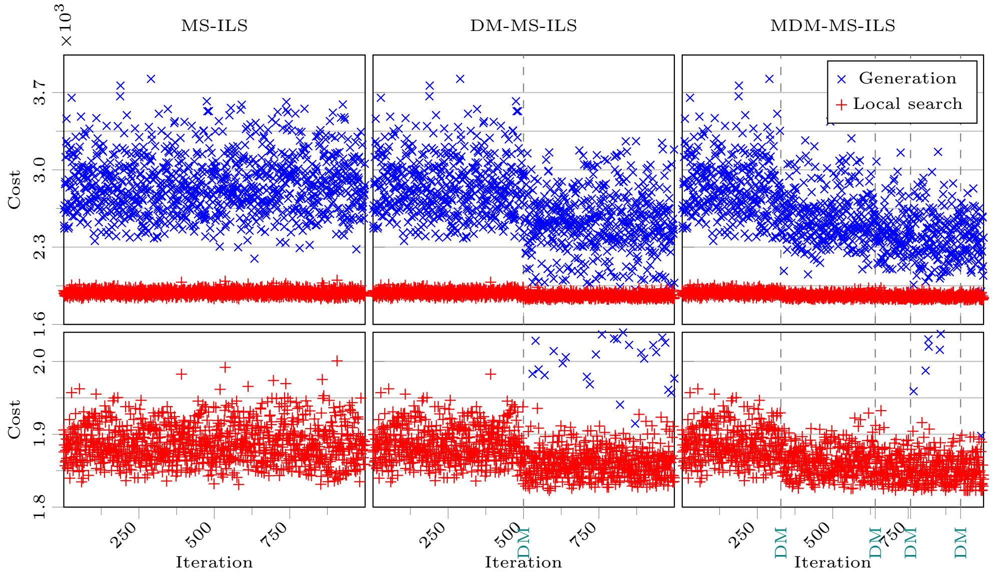


FIGURE 3. Cost vs. iteration charts illustrating the behavior of MS-ILS, DM-MS-ILS and MDM-MS-ILS for FSM-D instance N4 from Brandão's set over 1000 iterations.

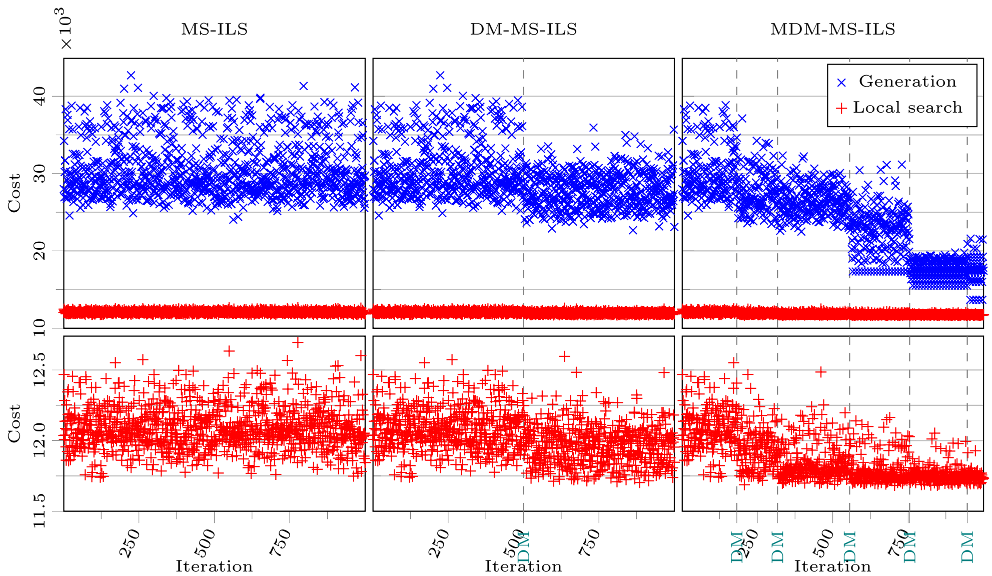


FIGURE 4. Cost vs. iteration charts illustrating the behavior of MS-ILS, DM-MS-ILS and MDM-MS-ILS for HFFVRP-FD instance 02 from Set 3 of Duhamel *et al.* over 1000 iterations.

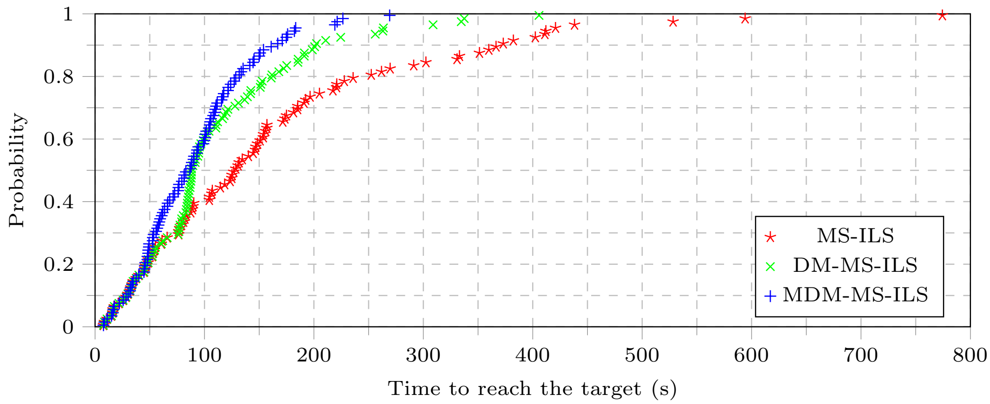


FIGURE 5. TTT plots comparing MS-ILS, DM-MS-ILS and MDM-MS-ILS for FSM-D instance N4 from Brandão's set [6].

data mining occurs only once, after iteration 500, whereas in MDM-MS-ILS, it is conducted several times in an adaptive manner (triggered by elite set stabilization). In both Figures 3 and 4, the charts in the first row reveal a noticeable reduction in the costs of the solutions generated by the hybrid heuristics after data mining, and the zoomed views in the second row show that there is also a related reduction in the costs of the solutions found in the local search phase. In MS-ILS, the generated solutions and those found in the local search phase remain at the same cost levels throughout the execution of the algorithm.

The charts show that this approach may lead to initial solutions very close to those found in the local search phase. In Figure 3 in particular, some of the initial solutions generated after data mining even appear in the zoomed views.

These results illustrate the behavior of the hybrid heuristics. After data mining, the patterns found are used in the generation phase, resulting in better-quality initial solutions (closer to local optima). Consequently, the local search benefits from better starting points to reach better solutions. The effects of data mining are observed with greater intensity in Figure 4, which is explained by the fact that instance 02 of Duhamel *et al.* is much larger (it includes 181 customers, whereas instance N4 from Brandão's set has 100).

In this experiment, a new best solution for instance 02 of Duhamel *et al.*, with a cost of 11683.24, was found by MDM-MS-ILS.

The second experiment focused on the generation and analysis of time-to-target (TTT) plots [1]. A TTT plot displays, on the ordinate axis, the probability that an algorithm will find a solution at least as good as a given target value within a given running time, which is shown on the abscissa axis. Such plots are used as a way to characterize the computational times of stochastic algorithms for optimization problems. In this experiment, each heuristic was run 100 times, with 100 different random seeds, targeting solutions with costs lower than or equal to the average costs obtained by the MS-ILS heuristic in the experiments presented in Sections 4.1 and 4.2 (1834 for instance N4 from Brandão's set and 11775 for instance 02 of Duhamel *et al.*). The TTT plots obtained are shown in Figures 5 and 6.

Again, the heuristics exhibit similar behaviors for both instances. The TTT plots show that the probabilities are equal for the lower time values. This is because the initial iterations of the hybrid heuristics are identical to those of the original heuristic. Therefore, if the target is achieved within these iterations by one of the heuristics, it is achieved by all of them. However, as the time values increase, the hybrid heuristics begin to outperform the original heuristic. For example, the probability that the target for instance N4 from Brandão's set (Fig. 5) will be reached within 225 seconds by MDM-MS-ILS is almost 100%, and the corresponding probability for DM-MS-ILS is approximately 92%, whereas that for MS-ILS is approximately 77%. Similarly, the probability that the target for instance 02 of Duhamel *et al.* (Fig. 6) will be reached within 2000 seconds by MDM-MS-ILS

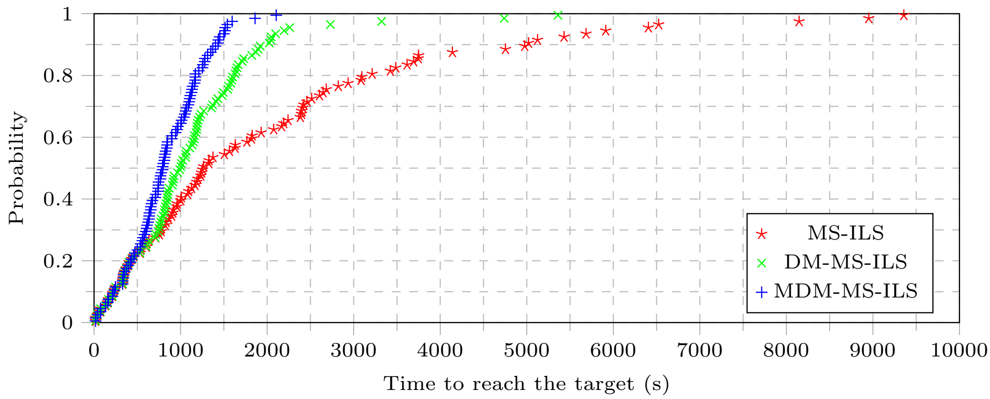


FIGURE 6. TTT plots comparing MS-ILS, DM-MS-ILS and MDM-MS-ILS for HFFVRP-FD instance 02 from Set 3 of Duhamel *et al.* [9].

is almost 100%, and the corresponding probability for DM-MS-ILS is approximately 90%, whereas that for MS-ILS is approximately 62%.

5. Conclusions and directions for future research

In this study, two hybrid heuristics were proposed and implemented based on the incorporation of data mining techniques into the state-of-the-art MS-ILS heuristic for solving the heterogeneous fleet vehicle routing problem proposed by Penna *et al.* [31]. These hybrid heuristics, named DM-MS-ILS and MDM-MS-ILS, were designed using the approaches applied in the DM-GRASP [39] and MDM-GRASP [33] hybrid metaheuristics, respectively.

We conducted computational experiments for the two major HFVRP classes. For the FSM problem, we considered 17 instances with the number of customers ranging from 20 to 199. For the HFFVRP, we used 106 instances with the number of customers ranging from 19 to 360. The results obtained confirm the effectiveness of the hybridization of heuristics with data mining, as the proposed hybrid heuristics achieved better solution costs and better computational times for most instances. Furthermore, new best solutions were found for six instances.

Both hybrid heuristics obtained better results than the original heuristic. The MDM-MS-ILS heuristic, the best of them, achieved better average costs for 78% of the instances (40% with statistical significance) compared with the original heuristic. When only instances with more than 100 customers are considered, the MDM-MS-ILS heuristic achieved better average costs for 86% (45% with statistical significance) compared with the original heuristic.

As demonstrated by the results obtained in this work and in the applications of DM-GRASP and MDM-GRASP reported in the literature, the data-mining-hybridized versions of the explored heuristics are able to generate initial solutions of higher quality, and consequently, the quality of the solutions found during the local search phase is also improved. In addition, the convergence time of the local search phase is systematically reduced. Considering that the improvement of the initial solution generation is the source of the advantages of the data mining hybridization approach, we intend to explore it more deeply in future research.

Appendix A. new best solutions

In this appendix, we present the new best solutions found in the experiments performed. For each solution, we identify the problem instance and indicate which heuristic found it, the number of routes used and its total cost. The solution itself is represented as a set of routes. For each route, we indicate the corresponding vehicle type and present the list of customers in visitation order.

Instance 41 (HFFVRP-FD, Duhamel *et al.* [9]): found by MDM-MS-ILS, 16 routes, cost 7571.44
(vehicle type): list of vertices
(A): 0 16 79 0
(B): 0 101 33 0
(C): 0 129 95 131 3 56 34 108 65 13 0
(C): 0 69 15 115 92 0
(C): 0 12 44 60 121 42 58 53 83 118 49 68 0
(E): 0 51 133 94 70 96 57 40 67 27 23 0
(E): 0 38 17 122 110 120 86 78 134 48 19 77 29 0
(E): 0 104 39 22 98 0
(E): 0 82 11 99 7 24 87 0
(E): 0 113 64 91 0
(E): 0 10 18 127 72 8 0
(E): 0 123 119 75 54 109 45 116 61 102 130 50 90 0
(E): 0 88 25 36 35 0
(F): 0 55 28 63 117 5 20 107 124 0
(F): 0 111 14 43 0
(G): 0 80 81 93 1 89 105 71 52 125 132 76 128 106 97 85 47 30 4 59
114 112 26 66 32 100 37 73 6 84 41 2 62 74 103 46 31 9 126 21 0

Instance 48 (HFFVRP-FD, Duhamel *et al.* [9]): found by MDM-MS-ILS, 12 routes, cost 21287.90
(vehicle type): list of vertices
(C): 0 59 1 15 98 75 0
(D): 0 77 61 41 31 107 0
(D): 0 63 104 43 56 35 64 108 54 103 65 89 87 94 95 49 99 13 10 79 0
(D): 0 93 51 50 82 30 80 42 0
(D): 0 84 48 91 4 11 97 67 0
(D): 0 28 40 100 92 69 12 106 0
(D): 0 85 78 37 6 57 0
(E): 0 96 101 17 46 7 68 44 81 22 21 5 27 0
(E): 0 105 2 9 25 3 110 71 73 19 38 60 72 33 0
(E): 0 47 55 109 32 16 0
(E): 0 70 8 45 86 58 24 14 29 76 36 83 20 0
(E): 0 102 62 66 34 26 18 39 88 52 90 53 23 74 0

Instance 89 (HFFVRP-FD, Duhamel *et al.* [9]): found by DM-MS-ILS, 21 routes, cost 7098.18
(vehicle type): list of vertices
(A): 0 98 8 93 0
(A): 0 103 117 105 116 0
(A): 0 87 23 63 0
(A): 0 14 58 59 0
(A): 0 80 75 10 128 20 0
(A): 0 100 2 74 53 0
(B): 0 41 104 27 18 133 34 36 48 0
(B): 0 95 79 57 64 91 47 0
(B): 0 26 6 107 66 0
(B): 0 1 92 62 89 0

(B): 0 15 115 61 113 44 99 0
(D): 0 32 67 33 35 81 90 0
(D): 0 70 114 84 111 72 38 97 22 0
(D): 0 25 17 71 0
(D): 0 123 56 37 0
(D): 0 50 69 121 126 0
(E): 0 49 85 5 4 21 132 46 119 31 45 39 82 0
(E): 0 40 13 3 120 16 118 78 51 122 54 108 28 0
(E): 0 55 124 68 106 112 96 110 65 130 0
(E): 0 101 42 52 83 102 77 88 127 125 129 94 131 30 76 86 0
(E): 0 109 19 43 24 60 11 12 73 29 9 7 0

Instance 02 (HFFVRP-FD, Duhamel *et al.* [9]): found by MDM-MS-ILS, 45 routes, cost 11683.24
(vehicle type): list of vertices
(A): 0 156 68 88 50 0
(A): 0 90 85 28 47 129 0
(A): 0 148 58 81 0
(A): 0 104 0
(A): 0 46 87 0
(A): 0 158 8 6 121 45 0
(A): 0 117 92 83 0
(A): 0 54 53 0
(A): 0 71 40 0
(A): 0 9 136 0
(A): 0 99 74 0
(A): 0 153 66 150 0
(A): 0 57 14 119 174 62 37 120 152 0
(A): 0 10 20 162 0
(A): 0 39 56 133 0
(A): 0 113 98 48 176 0
(A): 0 5 1 106 130 0
(A): 0 51 110 13 16 0
(A): 0 32 127 146 94 109 73 0
(A): 0 89 180 165 0
(A): 0 84 33 0
(A): 0 160 0
(A): 0 179 23 25 36 0
(A): 0 103 111 27 137 12 118 171 0
(A): 0 95 15 0
(A): 0 105 167 64 0
(A): 0 21 0
(A): 0 61 145 0
(A): 0 96 76 128 0
(A): 0 166 163 0
(A): 0 43 11 0
(A): 0 140 91 177 132 3 0
(C): 0 34 59 0
(C): 0 7 142 29 134 0
(C): 0 102 60 22 67 0

```
(C): 0 125 77 65 24 172 143 0
(C): 0 159 49 4 0
(C): 0 144 30 100 0
(C): 0 178 18 155 0
(C): 0 151 80 141 0
(C): 0 161 139 35 0
(C): 0 2 154 0
(D): 0 107 97 122 147 86 93 101 126 116 164 169 0
(D): 0 26 42 108 44 70 17 115 55 79 63 123 41 131 124 149 114 82 72
112 173 38 0
(D): 0 138 31 170 19 175 157 52 168 135 78 75 69 0
```

Instance 04 (HFFVRP-FD, Duhamel *et al.* [9]): found by MDM-MS-ILS, 31 routes, cost 10784.61
(vehicle type): list of vertices

```
(A): 0 131 166 178 36 3 0
(A): 0 90 142 179 124 103 26 111 133 0
(A): 0 48 5 79 93 167 0
(A): 0 44 64 119 0
(A): 0 154 95 57 0
(A): 0 116 17 75 134 12 0
(A): 0 34 39 176 60 0
(A): 0 152 69 38 0
(A): 0 51 50 122 138 99 149 0
(A): 0 31 22 25 153 132 19 0
(A): 0 16 2 0
(A): 0 4 42 161 136 0
(A): 0 23 67 129 20 0
(A): 0 168 14 33 58 0
(A): 0 165 76 68 0
(A): 0 110 73 59 9 123 83 29 0
(A): 0 61 108 180 37 150 182 0
(A): 0 128 173 112 27 0
(A): 0 146 45 62 72 143 0
(A): 0 181 162 47 0
(B): 0 74 81 15 43 0
(B): 0 155 13 32 46 0
(B): 0 87 120 77 49 135 21 78 0
(B): 0 97 89 0
(C): 0 106 177 169 70 144 114 63 24 0
(C): 0 66 71 163 35 0
(C): 0 101 30 88 104 157 53 127 117 86 55 115 0
(D): 0 28 7 113 160 98 164 94 65 158 175 174 151 0
(D): 0 18 172 109 147 40 126 105 159 171 148 6 125 85 54 0
(D): 0 11 8 100 80 84 170 10 52 145 156 92 0
(D): 0 96 137 121 107 130 1 140 56 91 118 102 141 41 139 82 0
```

Instance 67 (HFFVRP-FD, Duhamel *et al.* [9]): found by MS-ILS, DM-MS-ILS and MDM-MS-ILS, 16 routes, cost 10850.16
(vehicle type): list of vertices

```
(A): 0 25 2 158 71 81 151 169 56 39 0
(A): 0 51 152 170 77 168 123 137 76 91 4 125 164 0
(A): 0 162 98 155 80 167 135 73 128 129 24 0
(A): 0 15 156 163 130 5 122 103 119 149 0
(B): 0 95 45 44 116 3 36 0
(B): 0 66 7 89 113 20 154 104 97 120 50 86 138 33 157 87 0
(B): 0 26 166 165 65 83 0
(B): 0 96 22 111 54 69 131 27 147 90 0
(B): 0 126 159 105 148 0
(D): 0 171 124 142 53 140 61 146 8 107 11 46 60 99 0
(D): 0 41 12 57 68 63 31 94 93 141 17 133 19 35 10 88 0
(D): 0 82 160 115 132 21 14 108 49 117 139 58 153 110 0
(D): 0 118 43 16 101 161 67 136 145 143 28 55 112 40 70 74 121 9 0
(E): 0 62 34 79 84 42 37 102 38 64 114 78 0
(E): 0 32 134 59 144 92 75 48 23 150 106 0
(E): 0 29 30 72 85 100 109 6 13 127 47 1 18 52 0
```

References

[1] R.M. Aiex, M.G.C. Resende and C.C. Ribeiro, TTT plots: a perl program to create time-to-target plots. *Optimiz. Lett.* **1** (2007) 355–366.

[2] R. Baldacci, M. Battarra and D. Vigo, Routing a heterogeneous fleet of vehicles, in *The Vehicle Routing Problem: Latest Advances and New Challenges*, edited by B. Golden, S. Raghavan and E. Wasil. Springer US, New York, NY, USA (2008) 3–27.

[3] R. Baldacci and A. Mingozzi, A unified exact method for solving different classes of vehicle routing problems. *Math. Program.* **120** (2009) 347–380.

[4] H. Barbalho, I. Rosseti, S.L. Martins and A. Plastino, A hybrid data mining GRASP with path-relinking. *Comput. Oper. Res.* **40** (2013) 3159–3173.

[5] J. Brandão, A deterministic tabu search algorithm for the fleet size and mix vehicle routing problem. *Eur. J. Oper. Res.* **195** (2009) 716–728.

[6] J. Brandão, A tabu search algorithm for the heterogeneous fixed fleet vehicle routing problem. *Comput. Oper. Res.* **38** (2011) 140–151.

[7] E. Choi and D.W. Tcha, A column generation approach to the heterogeneous fleet vehicle routing problem. *Comput. Oper. Res.* **34** (2007) 2080–2095.

[8] C. Duhamel, C. Gouinaud, P. Lacomme and C. Prodhon, A multi-thread GRASPxELS for the heterogeneous capacitated vehicle routing problem, in Hybrid Metaheuristics, edited by E.G. Talbi. Springer Berlin Heidelberg, (2013) 237–269.

[9] C. Duhamel, P. Lacomme and C. Prodhon, A GRASPxELS with depth first search split procedure for the HVRP. Tech. Report LIMOS/RR-10-08. Laboratoire d'Informatique, de Modélisation et d'Optimisation des Systèmes (2010).

[10] C. Duhamel, P. Lacomme and C. Prodhon, Efficient frameworks for greedy split and new depth first search split procedures for routing problems. *Comput. Oper. Res.* **38** (2011) 723–739.

[11] T.A. Feo and M.G.C. Resende, Greedy randomized adaptive search procedures. *J. Global Optim.* **6** (1995) 109–133.

[12] J.A. Ferland and P. Michelon, The vehicle scheduling problem with multiple vehicle types. *J. Oper. Res. Soc.* **39** (1988) 577–583.

[13] C. Gencer, İ. Top and E.K. Aydogan, A new intuitional algorithm for solving heterogeneous fixed fleet routing problems: Passenger pickup algorithm. *Appl. Math. Comput.* **181** (2006) 1552–1567.

[14] M. Gendreau, G. Laporte, C. Musaraganyi and É.D. Taillard, A tabu search heuristic for the heterogeneous fleet vehicle routing problem. *Comput. Oper. Res.* **26** (1999) 1153–1173.

[15] B. Golden, A. Assad, L. Levy and F. Gheysens, The fleet size and mix vehicle routing problem. *Comput. Oper. Res.* **11** (1984) 49–66.

[16] G. Grahne and J. Zhu, Efficiently using prefix-trees in mining frequent itemsets, in *Proc. of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations* (2003).

[17] M. Guerine, I. Rosseti and A. Plastino, Extending the hybridization of metaheuristics with data mining to a broader domain, in *Proc. 16th Int. Conf. Enterprise Infor. Sys.* SCITEPRESS (2014) 395–406.

[18] M. Guerine, I. Rosseti and A. Plastino, Extending the hybridization of metaheuristics with data mining: Dealing with sequences. *Intell. Data Anal.* **20** (2016) 1133–1156.

[19] J. Han, M. Kamber and J. Pei, Data mining: Concepts and techniques. 3rd ed. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA (2011).

[20] A. Hoff, H. Andersson, M. Christiansen, G. Hasle and A. Løkketangen, Industrial aspects and literature survey: Fleet composition and routing. *Comput. Oper. Res.* **37** (2010) 2041–2061.

[21] A. Imran, S. Salhi and N.A. Wassan, A variable neighborhood-based heuristic for the heterogeneous fleet vehicle routing problem. *Eur. J. Oper. Res.* **197** (2009) 509–518.

[22] Ç. Koç, T. Bektaş, O. Jabali and G. Laporte, Thirty years of heterogeneous vehicle routing. *Eur. J. Oper. Res.* **249** (2016) 1–21.

[23] Y.A. Kochetov and A.V. Khmelev, A hybrid algorithm of local search for the heterogeneous fixed fleet vehicle routing problem. *J. Appl. Ind. Math.* **9** (2015) 503–518.

[24] J.K. Lenstra and A.H.G.R. Kan, Complexity of vehicle routing and scheduling problems. *Networks* **11** (1981) 221–227.

[25] F. Li, B. Golden and E. Wasil, A record-to-record travel algorithm for solving the heterogeneous fleet vehicle routing problem. *Comput. Oper. Res.* **34** (2007) 2734–2742.

[26] S. Liu, A hybrid population heuristic for the heterogeneous vehicle routing problems. *Transp. Res. Part E Logist. Transp. Rev.* **54** (2013) 67–78.

[27] S. Liu, W. Huang and H. Ma, An effective genetic algorithm for the fleet size and mix vehicle routing problems. *Transp. Res. Part E Logist. Transp. Rev.* **45** (2009) 434–445.

[28] H.R. Lourenço, O.C. Martin and T. Stützle, Iterated local search, in *Handbook of Metaheuristics*, edited by F. Glover and G.A. Kochenberger. Kluwer Academic Publishers, Dordrecht, Netherlands (2003) 321–353.

[29] R. Martí, Multi-start methods, in *Handbook of Metaheuristics*, edited by F. Glover and G.A. Kochenberger. Kluwer Academic Publishers, Dordrecht, Netherlands (2003) 355–368.

[30] D. Martins, G.M. Vianna, I. Rosseti, S.L. Martins and A. Plastino, Making a state-of-the-art heuristic faster with data mining. *Ann. Oper. Res.* **263** (2018) 141–162.

[31] P.H.V. Penna, A. Subramanian and L.S. Ochi, An iterated local search heuristic for the heterogeneous fleet vehicle routing problem. *J. Heuristics* **19** (2013) 201–232.

[32] P.H.V. Penna, T. Vidal, L.S. Ochi and C. Prins, New compound neighborhoods structures for the heterogeneous fixed fleet vehicle routing problem, in *Proc. of the XLV Brazilian Symp. Oper. Res.* (2013) 3623–3633.

[33] A. Plastino, B. Barbalho, L.F.M. Santos, R. Fuchshuber and S.L. Martins, Adaptive and multi-mining versions of the DM-GRASP hybrid metaheuristic. *J. Heuristics* **20** (2014) 39–74.

[34] A. Plastino, E.R. Fonseca, R. Fuchshuber, S.L. Martins, A.A. Freitas, M. Luis and S. Salhi, A hybrid data mining metaheuristic for the p-median problem, in *Proc. of the 2009 SIAM Int. Conf. Data Mining*, edited by H. Park, K. Wang, C. Apte and M.J. Zaki. SIAM (2009) 305–316.

[35] A. Plastino, R. Fuchshuber, S.L. Martins, A.A. Freitas and A. Salhi, A hybrid data mining metaheuristic for the p-median problem. *Stat. Anal. Data Min.* **4** (2011) 313–335.

[36] C. Prins, Two memetic algorithms for heterogeneous fleet vehicle routing problems. *Eng. Appl. Artif. Intell.* **22** (2009) 916–928.

[37] M.H. Ribeiro, A. Plastino and S.L. Martins, Hybridization of GRASP metaheuristic with data mining techniques. *J. Math. Model. Algorithms* **5** (2006) 23–41.

[38] M.H. Ribeiro, V.A. Trindade, A. Plastino and S.L. Martins, Hybridization of GRASP metaheuristic with data mining techniques, in *Proc. of the First International Workshop on Hybrid Metaheuristics* (2004) 69–78.

[39] L.F. Santos, S.L. Martins and A. Plastino, Applications of the DM-GRASP heuristic: a survey. *Int. Trans. Oper. Res.* **15** (2008) 387–416.

[40] L.F. Santos, R. Milagres, C.V. Albuquerque, S. Martins and A. Plastino, A hybrid GRASP with data mining for efficient server replication for reliable multicast, in *GLOBECOM '06*. IEEE (2006) 1–6.

[41] L.F. Santos, M.H. Ribeiro, A. Plastino and S.L. Martins, A hybrid GRASP with data mining for the maximum diversity problem, in *Hybrid Metaheuristics*, edited by M.J. Blesa, C. Blum, A. Roli and M. Sampels. Springer Berlin Heidelberg (2005) 116–127.

[42] A. Subramanian, P.H.V. Penna, E. Uchoa and L.S. Ochi, A hybrid algorithm for the heterogeneous fleet vehicle routing problem. *Eur. J. Oper. Res.* **221** (2012) 285–295.

[43] É.D. Taillard, A heuristic column generation method for the heterogeneous fleet VRP. *RAIRO: OR* **33** (1999) 1–14.

[44] C.D. Tarantilis, C.T. Kiranoudis and V.S. Vassiliadis, A threshold accepting metaheuristic for the heterogeneous fixed fleet vehicle routing problem. *Eur. J. Oper. Res.* **152** (2004) 148–158.

[45] P. Toth and D. Vigo, An overview of vehicle routing problems, in *The Vehicle Routing Problem*, edited by P. Toth and D. Vigo. SIAM, Philadelphia, PA, USA (2001) 1–26.

[46] T. Vidal, T.G. Crainic, M. Gendreau and C. Prins, A unified solution framework for multi-attribute vehicle routing problems. *Eur. J. Oper. Res.* **234** (2014) 658–673.

[47] N.A. Wassan and I.H. Osman, Tabu search variants for the mix fleet vehicle routing problem. *J. Oper. Res. Soc.* **53** (2002) 768–782.