To cite this article:

Molina, J. C., Salmeron, J. L., Eguia, I., & Racero, J. (2020). The heterogeneous vehicle routing problem with time windows and a limited number of resources. *Engineering Applications of Artificial Intelligence*, *94*, 103745..

The heterogeneous vehicle routing problem with time windows and a limited number of resources

Abstract This paper introduces the heterogeneous vehicle routing problem with time windows and a limited number of resources (HVRPTW-LR), a practical extension of the classical vehicle routing problem in which routes to be designed share common scarce resources. The HVRPTW-LR arises when a limited number of resources, such as vehicles, drivers, instruments, and so on, are available but are insufficient to serve all customers in a route planning. Therefore, the route design involves the selection of customers to be visited at each route and resources to be used. Applications to this problem are found in real services companies with high seasonal demand which attend to different types of works and have to decide on how to effectively manage their resources. For designing optimal routes, a hierarchical objective function is considered, maximizing the total number of served customers as the primary objective, and minimizing the travel costs as secondary. A mathematical model of linear programming is introduced to describe and understand all constraints clearly. The problem is first heuristically solved by a semi-parallel insertion heuristic. Then, solutions are improved by a hybrid variable neighborhood descent metaheuristic based on a Tabu Search algorithm for the exploration of the neighborhood and a holding list. Experiments are conducted on numerous sets of benchmark instances from the literature to evaluate the performance of the proposed algorithm. Results show that the algorithm proposed in this paper has a good performance and can be easily applied for solving numerous vehicle routing problem variants from the literature. A new set of benchmark cases for the HVRPTW-LR are also presented and solved.

Keywords Vehicle Routing Problem, VRP with time windows, Fixed Heterogeneous Fleet, Variable Neighborhood Descent, Analytical and Numerical Modelling, Performance Analysis.

1 Introduction

The world of logistics is changing at breakneck speed. Excess of inventory or overstock is a supply chain problem costing companies large amounts of money every year. Often such inventory is reduced by requiring smaller and more frequent order quantities. Moreover, some logistics sectors are also exposed to high demand seasonality during the year. Thus, in specific months companies have a higher demand than the rest of the year requiring an increase of their logistical resources to complete all customers' orders.

Most distribution companies need resources for the provision of their services. Resources represent sources of supply, support, or aid which are needed to complete the work. In this work, the resources are classified into two types: renewable resources (RRs) and non-renewable or consumable resources (CRs). RRs are needed during the execution of the service action but are not consumed. They are available on a period by period basis, i.e. the available amount is renewed from period to period. Typical examples are the instruments needed to perform the service, vehicles, skilled drivers and others.

On the other hand, CRs are consumed during the service action and they are also available on a period by period basis but rather have limited consumption availability for the route planning. Examples are raw materials, wires, electric counters, etc. Obviously, both types of resources are limited and the decision on how to efficiently use them to distribute products or to perform services in a route planning is difficult to take.

Fleet managers play an important role in logistics and transportation companies. They are responsible for routing drivers and vehicles in order to keep distributions on schedule and within established budgets. The growing demand for efficiency in all areas of companies induces fleet managers to make their fleets as effective and efficient as possible, using the least amount of resources. Based on this observation, if on certain occasions the total customer demand in a specific time period exceeds the capabilities of the company, some of the received orders from customers may be postponed to the next planning horizon as the cost of increasing the logistical resources is too high. On the other hand, the fleet of vehicles in a company is usually heterogeneous as the company incorporates vehicles of different characteristics over time (Hoff et al., 2010) and also it provides a better adaptation to the customer demand (Yepes and Medina, 2002).

The heterogeneous vehicle routing problem with time windows (TW) and a limited number of resources (HVRPTW-LR) is an extension of the classical heterogeneous vehicle routing problem with time windows (HVRPTW) where some customers of a specific planning may not be served due to an excess in the required resources (RRs and/or CRs). In this situation, the HVRPTW-LR requires maximizing the use of those resources, postponing as few orders as possible to the next planning horizon.

This paper makes two main scientific contributions. Firstly, the HVRPTW-LR is formally defined by a mixed-integer linear programming model that considers the resource limitations in the route planning. To the best of our knowledge, the HVRPTW-LR has not yet been introduced in the literature and better defines real-world vehicle routing problems (VRPs). The second contribution is the development of a methodology for solving the HVRPTW-LR. For this purpose, a hybrid variable neighborhood descent (VND) metaheuristic based on a Tabu Search (TS) algorithm for the exploration of the neighborhood is proposed in this paper. The algorithm introduces new neighborhood structure definitions and a holding list (HL) that contains the list of unserved customers. The HL also prevents the algorithm from being trapped at local optimal and achieves to explore a larger search space. To the best of our knowledge, hybrids of VND and TS with HL (VNDTS-HL) are not proposed in the literature for solving VRPs.

The rest of the paper is structured as follows: Section 2 provides a review of the literature. The problem description and the mathematical formulation are defined in Section 3. Section 4 presents a detailed description of the main components of the solution approach. Computational experiments are presented in Section 5 and finally, the conclusions are given in Section 6.

2 Literature survey

The Heterogeneous Fleet VRP (HF-VRP) is a variant of the classical vehicle routing problem (VRP) that appears when a fleet of vehicles (limited or unlimited), characterized by different capacities and costs is available for the distribution activities (Koç et al., 2016). There are two main divisions of such problems depending on the available fleet. The HF-VRP with unlimited fleet, known as the Fleet Size and Mix VRP (FSMVRP), consists of determining the best fleet composition and routing when there is no limitation on the number of available vehicles of each type. On the other hand, the variant with a limited number of vehicles, called the Heterogeneous VRP (HVRP) consists of optimizing the routes with the available fixed fleet. Two different objective functions have been considered to compute route costs to be minimized. Both consist of minimizing the sum of fixed vehicle costs and dependent routing costs. The latter can be related to the total trip duration without service times or based on the total distance traveled. The HF-VRP with TW (HF-VRPTW) appears when additional restrictions are introduced to force customers to be served by a vehicle in a predefined time interval [EW_i , LW_i], where EW_i and LW_i are the earliest and latest times to start the service respectively.

The HF-VRPTW has been widely studied and a large number of papers have been published in the literature, as can be observed in the surveys of Baldacci et al. (2008), Irnich et al. (2014) and Koç et al. (2016). The majority of these works address the FSMVRP with TW (FSMVRPTW), maybe because it is easier to find feasible solutions than in HVRP with TW (HVRPTW).

In the area of HVRPTW, Paraskevopoulos et al. (2008) proposed a two-phase solution framework based on a semi-parallel construction heuristic and a hybridized TS integrated within a new reactive variable neighborhood search (VNS) metaheuristic algorithm. The algorithm solved both HVRPTW and FSMVRPTW. Later, Koç et al. (2015) proposed a hybrid evolutionary algorithm to also solve both problems. The algorithm combines advanced procedures and several metaheuristics principles from Vidal et al. (2012, 2014), such as adaptive large-scale neighborhood search and population search. They obtained some new solutions on benchmark instances outperforming all previous algorithms.

Both the FSMVRPTW and HVRPTW have given rise to a multitude of variants which have received particular attention in the last recent years. We refer to the survey of Koç et al. (2016) for further information on HF-VRPTW.

All these variants assume that a feasible solution occurs when every customer of a route plan is served by a vehicle on a route satisfying all problem constraints, whereas a solution of the HVRPTW-LR may contain unserved customers. Following this consideration, Lau et al. (2003) proposed the VRP with TW and a limited number of vehicles (m-VRPTW). This variant considers a homogeneous limited fleet that is available at the depot. In contrast to the HVRPTW, a solution to m-VRPTW may contain some unserved customers due to the limited amount of vehicles. The objective in this problem is a hierarchical function where maximizing the total number of served customers is considered as the primary objective and minimizing the total traveled distance as secondary. Our research not only considers the limitation in the number of vehicles but also incorporates limitations in RRs and/or CRs, which are used and/or consumed at customer locations. Thus, the different routes designed in a specific day have to share these scarce resources.

In order to solve the aforementioned m-VRPTW, Lau et al. (2003) proposed a TS approach characterized by a HL and a mechanism to force dense packing within a route. The algorithm was tested on a set of nine benchmark instances derived from the clustered set of Solomon (1987) for the VRPTW; they also showed good results for the standard VRPTW. Extending this research, many authors have presented different approaches for the m-VRPTW (Lim and Wang 2004; Lim and Zhang 2007; Wang et al 2008).

Jiang et al. (2014) defined the VRP with heterogeneous fleet and time windows to generalize the three variants existing in the literature (m-VRPTW, FSMVRPTW and HVRPTW). They considered a hierarchical objective function where the total number of served customers is considered as the primary objective and minimizing the total number of vehicles and total traveled distance as secondary and tertiary respectively. The authors developed a two-phase TS algorithm using the TS developed by Lau et al. (2003).

Relating to the use of limited resources, Molina et al. (2018) present the problem of designing routes in service companies that are responsible for the metrological control of measuring equipments. They solve a rich vehicle routing problem that includes fixed heterogeneous fleet of vehicles, time windows for customers and depot, resource synchronization between tours, driver-customer and vehicle-customer constraints, customer priorities and unserved customers. Hempsch and Irnich (2008) introduced the VRPs with inter-tour resource constraints in which the feasibility of a solution depends on properties of several tours and cannot be decided by considering the individual tours separately. Examples of inter-tour resource constraints are the presence of a limited number of docking stations at depot, a limited number of long tours, the number of stops, the arrival time at the depot, etc. For further information, an extended survey on VRPs with multiple synchronization constraints is due to Drexl (2012). Our problem differs from the previous one in the types of resources and interdependencies considered. In the HVRPTW-LR, the utilization or consumption of a specified resource is always performed at customer sites and there exists a global limitation for all vehicles.

In order to design a methodology in accordance with the characteristics of the tackled problem, a hybrid VND metaheuristic based on a TS algorithm for the exploration of the neighborhood is proposed in this paper. Although the algorithm is specifically designed to consider the specific assumptions, objectives and constraints of the HVRPTW-LR, it could also be very useful in other research areas, such as in Shuttle-Based Storage and Retrieval Systems (SBS/RS) (see e.g. Carlo et al., 2012; Borovinšek et al., 2017; Lerher et al., 2017, 2015; Dukic et al., 2015; Lerher, 2016; Rosi et al., 2016).

3. Problem Description and Mathematical Model

The HVRPTW-LR is defined on a graph G= {N, A} with N ={0,1,…,N} as a set of nodes for a route planning, where node 0 represents the depot, and A is a set of arcs defined between each pair of nodes. A set of K heterogeneous vehicles is represented by K={1,…,K} and is available from the depot. With the purpose of containing those nodes that cannot be served in the specific routes planning, a HL is available. Each customer i ∈ {1,…,N} has a fixed demand and a time window for starting the service The sets of RRs, except vehicles, and CRs are represented by R={1,…,R} and C={1,…,C} respectively. Moreover, there exists a latest returning time to the depot for each vehicle k ∈ {1,…,K} and a limited number of each resource.

The aim of the HVRPTW-LR is to design a set of vehicle routes making the following assumptions:

- Each vehicle serves a single route during the route planning.
- All vehicle routes start and end at the depot.
- The vehicle fleet is heterogeneous, i.e. vehicles have different capacities, variable costs and latest returning times to the depot.
- Customers of a specific planning may not be served.
- Each customer has a positive demand which has to be fully satisfied once by exactly one vehicle, if the customer is served in the route planning.
- Customers may require RR and/or CR to be served.
- The total demand of a route does not exceed the vehicle capacity.
- The service of every visited customer starts within its time windows; if the vehicle arrives before the earliest time, it must wait.
- A vehicle cannot exceed its latest returning time to the depot.

- There is a limitation in the use of RRs and CRs in the route planning.

The notation adopted is presented in Table 1:

Table 1. Data notation

$[EW_i,LW_i]$	*Earliest and latest time to begin the service at node i* $\in \{1,\ldots,N\}$.
ST_i	*Service time in node i* $\in \{1,\ldots,N\}$.
TD_{ij}	*Distance from node i to node j* $(i \neq j)$.
TT_{ij}	*Travel time from node i to node j*
D_i	*Load demanded by node i* $\in \{1,\ldots,N\}$.
Q_k	*Capacity of vehicle k* $\in \{1,\ldots,K\}$.
β_k	*Variable cost of vehicle k* $\in \{1,\ldots,K\}$.
$TMax_k$	*Latest returning time to the depot of vehicle k* $\in \{1,\ldots,K\}$.
SR_{ir}	*Equal to 1 if node i* $\in \{1,\ldots,N\}$ *needs the renewable resource r* $\in \{1,\ldots,R\}$ *to be served, 0 otherwise.*
NR_r	*Number of renewable resource r* $\in \{1,\ldots,R\}$ *available in the route planning*
SC_{ic}	*Quantity of consumable resource c* $\in \{1,\ldots,C\}$ *demanded by node i* $\in \{1,\ldots,N\}$.
NC_c	*Quantity of consumable resource c* $\in \{1,\ldots,C\}$ *available in the route planning.*
ρ	*Small positive scalar*

The problem uses the following decision variables:

- X_{ijk} : binary variable, equal to 1 if vehicle k ∈ {1,…,K} travels from nodes i to j (i ≠ j)
- Z_i : binary variable, equal to 1 if customer i ∈ {1,…,N} is not served by any vehicle k ∈ {1,…,K}.
- Y_{ik} : starting service time at node i ∈ {0,1,…,N} performed by a vehicle k ∈ {1,…,K}; y_{0k} is the arrival time at the depot.
- W_{rk} : binary variable, equal to 1 if renewable resource r ∈ {1,…,R} is assigned to vehicle k ∈ {1,…,K}.
- W'_{ck} : quantity of consumable resource c ∈ {1,…,C} assigned to vehicle k ∈ {1,…,K}.

According to the established assumptions, the constraints of the mixed-integer linear programming model are as follows:

$$\sum_{j=1}^{N} X_{0jk} \le 1 \quad \forall k \tag{1a}$$

$$\sum_{\substack{j=0\\ j\neq i}}^{N} X_{jik} - \sum_{\substack{j=0\\ j\neq i}}^{N} X_{ijk} = 0 \quad \forall i, \forall k \tag{1b}$$

$$\sum_{k=1}^{K} \sum_{\substack{j=0\\ j\neq i}}^{N} X_{ijk} + Z_i = 1 \quad \forall i \neq 0 \tag{1c}$$

$$Y_{ik} + ST_i + TT_{ij} \le Y_{jk} + TMax_k \cdot (1 - X_{ijk}) \quad \forall i \neq 0, \forall j; i \neq j, \forall k \tag{1d}$$

$$TT_{0j} \le Y_{jk} + TMax_k \cdot (1 - X_{0jk}) \quad \forall j \neq 0, \forall k \tag{1e}$$

$$EW_i \cdot \sum_{j=0}^{N} X_{ijk} \le Y_{ik} \le LW_i \cdot \sum_{j=0}^{N} X_{ijk} \quad \forall i \neq 0, \forall k \tag{1f}$$

$$Y_{0k} \le TMax_k \quad \forall k \tag{1g}$$

$$\sum_{i=1}^{N} D_i \sum_{\substack{j=0\\ j\neq i}}^{N} X_{ijk} \le Q_k \quad \forall k \tag{1h}$$

$$\sum_{\substack{j=0\\ j\neq i}}^{N} X_{jik} + SR_{ir} \le 1 + W_{rk} \quad \forall i \neq 0, \forall k, \forall r \tag{1i}$$

$$\sum_{k=1}^{K} W_{rk} \le NR_r \quad \forall r \tag{1j}$$

$$W'_{ck} = \sum_{i=1}^{N} \sum_{\substack{j=0\\ j\neq i}}^{N} X_{ijk} \cdot SC_{ic} \quad \forall c \ \forall k \tag{1k}$$

$$\sum_{k=1}^{K} W'_{ck} \le NC_c \quad \forall c \tag{1l}$$

$$\sum_{c=1}^{C} W'_{ck} + \sum_{r=1}^{R} W_{rk} \le (R + C) \cdot \sum_{j=1}^{N} X_{0jk} \quad \forall k \tag{1m}$$

Constraints (Eq. 1a) mean that each vehicle departs from the depot once or it does not, i.e., no more than K vehicles (fleet size) depart from the depot. Constraints (Eq. 1b) are the flow conservation on each node. Constraints (Eq. 1c) guarantee that each customer is visited exactly once by a vehicle or transferred to the HL ($Z_i = 1$). Starting service times for visited customers are calculated in constraints (Eq. 1d) and (Eq. 1e), where Y_{0k} is the ending time of the tour for vehicle k. These constraints also avoid subtours. Time windows are imposed by constraints (Eq. 1f). Constraints (Eq. 1g) avoid exceeding the latest returning time to the depot. Constraints (Eq. 1h) ensure that no vehicle can be overloaded. Constraints (Eq. 1i) ensure the use of RRs for every served customer in a route, and constraints (1j) limit the available number of each one in the route planning. The limitation of CRs is given by constraints (1k) and (1l). Finally, constraints (1m) avoid assigning RRs and CRs to a vehicle if it does not depart from the depot in the route planning.

The objective of the HVRPTW-LR is a hierarchical objective function, where maximizing the total number of served customers is considered as the primary objective, and minimizing the total travel cost as secondary. The travel cost is determined by the product of the vehicle unit travel cost and the travel distance of this vehicle. Obviously, maximizing the total number of served customers is equivalent to minimizing the total number of unserved customers in the HL. If multiple solutions serving the same number of customers exist, the model must choose the one with lower travel costs. For this purpose, ρ is introduced in the objective function (Eq. 2). This value represents a small positive number to be a necessary and sufficient condition for Pareto optimality (Steuer, 1986).

$$MIN \quad \sum_{i=1}^{N} Z_i \; + \rho \cdot \sum_{k=1}^{K} \beta_k \sum_{j=0}^{N} \sum_{\substack{i=0 \\ i \neq j}}^{N} TD_{ij} X_{ijk} \tag{2}$$

Contrary to the formulation of HVRPTW where unfeasible solutions appear when there are unserved customers, the HVRPTW-LR always generates feasible solutions. Recall that the HL is a component of the final solution and contains the customers that are not served by any vehicle of the available fleet composition ($Z_i=1$). Based on these observations, it is reasonable to assume that the HVRPTW-LR is reduced to a standard HVRPTW when RRs and CRs are not considered and all customers can be served by the available fleet.

4. The Variable Neighborhood Descent Tabu Search Algorithm with Holding List

In this section, the authors introduce the Variable Neighborhood Descent Tabu Search algorithm with Holding List (VNDTS-HL)–to efficiently solve the HVRPTW-LR. The basic VND algorithm is upgraded by the use of TS for the exploration of the neighborhood and by the inclusion of several new neighborhood structures' definitions and a holding list that achieves flexibility. In addition, a deterministic diversification procedure is introduced to readjust the use of resources in the routes escaping from local optima.

In the proposed algorithm, a semi-parallel construction heuristic is first used to generate several initial solutions with different sequences of parameter values. From these solutions, only those of higher quality are selected and stored in a set F for an optimization process in a second phase. The optimization process includes the VNDTS-HL and a Resources Readjustment Mechanism (RRM) which is applied to modify the solution in order to escape from the local optima. Therefore, the algorithm starts with an initial solution and the optimization process is repeated for all the initial solutions of the set F, re-starting from a new initial solution once a region has been extensively explored. The proposed solution methodology is terminated when either all selected initials solutions have been examined or an upper bound limit with respect to the computational time consumption has been reached. It is observed that the algorithm adopts a multi-start strategy achieving diversification to obtain high quality solutions.

The remainder of this section introduces the proposed solution methodology. The semi-parallel construction heuristic for the HVRPTW-LR is briefly described in Section 4.1 while Section 4.2 presents the components of the optimization process.

4.1 Semi-parallel construction heuristic

The heuristic proposed in this paper is based on the semi-parallel construction heuristic proposed by Paraskevopoulos et al. (2008) with a special mechanism to tackling infeasible instances (HL) and considering the global limitation in the number of RRs and CRs required in the route planning. It is important to note that when the problem specification includes an unlimited type of resource or does not consider resources, no restriction will be assumed on the design of the routes.

Following the insertion scheme of Paraskevopoulos et al. (2008), routes are initialized by a "seed" criterion based on the customer with minimum slack (LT_i - max(TD_{0i}, TD_{i0}). Then, as limitations on RRs and CRs need to be considered, the new greedy function that measures the cost of inserting a customer u between i and j served by vehicle k is denoted in (3) where the α weights define the relative contribution of each individual metric to the overall selection

$$\phi_{ij}^{uk} = \left(\alpha_1 \cdot \left(C_{ij,u}^0 + C_{ij,u}^1\right) + \alpha_2 \cdot C_{ij,u}^2 + \alpha_3 \cdot C_{ij,u}^3 + \alpha_4 \cdot \xi \cdot C_{ij,u}^4\right) * PC_R * PC_C \tag{3}$$

The first metric (Eq. 4a) is a measure of the coverage of the time window for the selected customer u. (Ioannou et al., 2001) and represents the closeness between the vehicle arrival time at customer u (Y_u) and its earliest service time (EW_u). Furthermore, the compatibility of the customer time window in the specific insertion position is introduced in the second metric (Eq. 4b). It represents the time gap between the latest service time (LW_u) and the time of the vehicle arrival at customer u (Ioannou et al., 2001). When a customer u is inserted between two consecutive customers (i,j) in a route, a driving time increase is produced and is given by the third metric (Eq. 4c). Metric (Eq. 4d) considers the time difference between the vehicle arrival time at customer j, before and after the insertion of customer u into the current route. This metric represents the time gap that has to be pushed forward in customer j to insert u (Ioannou et al., 2001).

Metric (Eq. 4e) gives priority to the insertion of customers with large demands on the route and maximizes the utilization of the vehicle capacity. The metric was used by Paraskevopoulos et al. (2008) and Kritikos et al. (2013) in their respective works but they did not consider that the result is a load measurement resulting in a difference in the dimension units with respect to the other cost components of the greedy function, which are expressed in terms of time. To solve this difference, a factor ξ is introduced multiplying this metric and is the result of dividing the average travel time between all nodes in the problem by the average demand. This factor is expressed in equation (Eq. 4f).

$$C_{ij,u}^0 = Y_u - EW_u \tag{4a}$$

$$C_{ij,u}^1 = LW_u - \left(Y_i + ST_i + TT_{iu}\right) \tag{4b}$$

$$C_{ij,u}^2 = TT_{iu} + TT_{uj} - TT_{ij} \tag{4c}$$

$$C^3_{ij,u} = \left[LW_j - \left(Y_i + ST_i + TT_{ij}\right)\right] - \left[LW_j - \left(Y_u + ST_u + TT_{uj}\right)\right] \tag{4d}$$

$$C^4_{ij,u} = Q_k - \left(\sum_{i \in N \cap u} D_i \sum_{j \in N} X_{ijk}\right) - D_u \tag{4e}$$

$$\xi = \frac{\sum_{i,j \in N} TT_{ij} \Big/ (N+1) \cdot N}{\sum_{i \in N} D_i \Big/ N} = \frac{\sum_{i,j \in N} TT_{ij}}{(N+1)\sum_{i \in N} D_i} \tag{4f}$$

A point of primary importance, for the HVRPTW-LR, is the effective utilization and/or consumption of the available resources. For this reason, two different penalty costs are introduced in the “greedy” function. First, the renewable penalty cost (PCR) gives priority to insert customers which do not modify the RR utilization on the route. Thus, the value of the PCR will be equal to one if the candidate customer for insertion requires the same types of RRs as customers belonging to the partially constructed route. On the other hand, the consumable penalty cost (PCC) only penalizes those customers requiring a great number of CRs. In the PCC, the evaluation criterion for penalizing a customer insertion is based on the third quartile.

Although the formulae for percentiles are uniquely defined for continuous random variables, in the discrete case, quartiles divide the customers, previously sorted from lowest to highest consumptions, into four quarters having the same number of customers in each quarter. The third quartile represents the value for which the resource consumption of the 75 per cent of the customers is less than that value. Therefore, those customers with a greater value in the consumable resource (CR) consumption than the quartile will be penalized. As observed, the main idea of this method is to design routes that require the least number of RRs and CRs. Recall that penalty costs only will be considered according to the problem specification.

Finally, as the primary objective of the HVRPTW-LR consists of maximizing the total number of served customers, a new metric (M_k) is introduced in Eq. 5 for the routes selection to prioritize the objective of the tackled problem. The main idea is to measure the number of customers that is served in a route using a vehicle of type k. In the case of having multiple routes serving the same number of customers, the metric will choose the route with smaller travel costs. At the end of the procedure, if there are any unassigned customers left, a HL is generated to locate them.

$$M_k = \frac{\sum_{(i,j)\epsilon route\ k} X_{ijk}}{1+\rho\cdot\sum_{(i,j)\epsilon route\ k} TD_{ij}\cdot X_{ijk}} \quad (5)$$

4.2 The optimization process

The metaheuristic method we propose, the VNDTS-HL, is originally inspired in VNS. VNS is a metaheuristic proposed by Mladenovic and Hansen (1997) that is composed of three phases: *shaking*, *local search* and *move*. Thus, VNS explores a set of neighborhoods of a current solution, makes a local search and moves to another solution only if there has been an improvement. Since a local optimum for a given type of move (neighborhood structure) is not necessarily so for another, VNS changes the neighborhood structure during the search in order to escape from local optima. The search continues until a local minimum with respect to all neighborhood structures is reached (Hansen et al., 2010).

The *shaking* procedure is a diversification mechanism that consists in perturbing a solution by applying a random move, to provide a new starting point for the local search. Since the shaking mechanism is used to allow a more efficient and effective intensification local search, random moves are not implemented in our approach because they may lead to different route solutions for different executions of the algorithm without changing the parameters, which is not desirable for real distribution companies. For this reason, the deterministic version of VNS, called VND, is considered. Other variants of the VNS are presented in Hansen and Mladenovic (2003).

To avoid termination at a local minimum and in order to modulate the intensification and diversification of the search, a TS mechanism, that allows non-improving moves, is incorporated to the local search phase. Moreover, a new mechanism, called RRM, is applied to HVRPTW-LR to escape from local minimum. Finally, a HL is also introduced to tackle unserved customers in the final solution.

Unlike evolutionary algorithms, VNDTS-HL is not population-based, and successively moves from one solution to another. This is an advantage to efficiently update the objective function and to check the feasibility in the resource limitations restrictions, without recalculation of the overall solution. Moreover, the choice of VNDTS-HL is motivated by the high complexity of the HVRPTW-LR, which requires algorithms with substantial diversification possibilities as the VND scheme. Furthermore, the utilization of TS results in an intensification of the search, which is of vital importance to find promising solutions. On the other hand, implementation of TS requires specific definition of a neighborhood, which is given by the VND.

In the literature, VNS scheme has proved to be very adaptable to VRP variants and it has been successfully applied combined with TS for solving the HVRPTW. Some authors (Molina et al., 2019; Paraskevopoulos et al., 2008) performed the local search of VNS by TS while VNS controls the neighborhood changes. In contrast, Brandão (2006, 2009, 2011) presented a hybrid algorithm where different neighborhood structures are considered in a TS mechanism. Recent research on applications of VNS, or of hybrids of VNS combined with other metaheuristics is diverse and numerous (Bräysy, 2001, 2003; Chen et al., 2010; De Armas et al., 2015; Hansen and Perez., 2010; Paraskevopoulos et al., 2008). For further information, Hansen and Perez (2010) present an extensive review on VNS applications.

More precisely, the VNDTS-HL starts by defining a set of neighborhood structures N_λ ($\lambda = 1,\ldots, \lambda_{\max}$). The iterative process starts from an initial solution *s*. Then, a local search based on TS is performed to determine a new solution *s´* in N_λ. If *f(s´)* is better than the best solution *f(s)*, then *s* is replaced by *s´*, and the search returns to N_1, otherwise the search explores the next neighborhood $N_{\lambda+1}$. This is repeated until all neighborhood structures are examined ($\lambda = \lambda_{max}$). At this point, the RRM phase is executed once to escape from a local optimum and the VNDTS-HL restarts from the modified solution until a new local optimum solution *s´* is reached. If the obtained solution *f(s´)* improves *f(s)*, *s* is replaced by *s´* and the algorithm starts from a new initial solution from the set *F*. The pseudocode of the proposed algorithm is presented in Algorithm 1.

Variable Neighborhood Descent Tabu Search algorithm

```
F ← Semi-parallel_Insertion_Heuristic;
Define a set of neighborhood structures Nλ, λ=1, 2,…,λmax;
For all solutions s of set F do:
   While (CPU time consumed ≤ γ) do:
      Set k ← 1 ;  RRM ← False;
      While (k ≤ kmax) do:
     s' ← TabuSearch (s, Nλ);
     If f (s') improves f (s) then
        s ← s'; k ← 1;
     Else
           If  k< kmax then
         k ← k+1;
           Else
              If   RRM=False  then
                 ŝ ← s;  s ← RRMPhase( s) ; k ← 1; RRM ← True;
              Else
                 If  f (s) improves f (ŝ) then
                    ŝ ← s
                 EndIf
              EndIf
           EndIf
        EndIf
      EndWhile
      UpdateBestSolution (ŝ);
   EndWhile
EndFor
```

4.2.1 Tabu Search

The TS is a widely used metaheuristic due to Glover (1989) that carries out the exploration of the solution space moving successively, in each iteration from one solution s to the best or first improving solution of its neighborhood $N_\lambda(s)$ even if it may cause a deterioration in the objective function.

The central mechanism in the TS is a short-term memory known as the Tabu List (TL) which stores the solutions explored throughout the search or, more commonly, some relevant attributes of these solutions. In this problem type, due to the computational effort, the proposed TL consists of storing the attributes of the performed movements such as the involve nodes and their initial positions in the routes before moving to other solution.

To prevent the search from returning to recently visited solutions and to drive the search toward regions of the solution space not yet explored, these selected attributes are declared as tabu and remain in the TL for a specific number of iterations (tabu tenure) unless the aspiration criterion is satisfied. The latter is applied when a move declared as tabu builds a solution that overcomes the best solution found so far. In this case, the mechanism dismisses the TL and the move is accepted.

Literature shows several schemes to determine and control the size of the TL during the search (Glover, 1989; Paraskevopoulos et al. 2018). Obviously, the use of a small tabu tenure results in a more effective intensification search since it allows the TS to explore some areas of the solution space allowing the cycling of small periods. In contrast, a large tabu tenure will drive the search toward a part of the solution space that has not been explored yet, usually escaping from a current local optimum solution. This work proposes a scheme for the tabu tenure that provides a balance between diversification and intensification search strategies using a particular mechanism that has been successfully applied in the works of Molina et al. (2019) and Paraskevopoulos et al. (2008). Initially, the tabu tenure t_{size} is set equal to a lower value t_{min}. A diversification mechanism is provided by incrementing at each iteration the tabu tenure in one unit up to an upper bound t_{max} while no improvement is observed. In contrast, the intensification mechanism is performed when an improvement in the objective function is achieved. For this purpose, the tabu tenure is reinitialized to t_{min}. The pseudocode of the proposed TS is presented in Algorithm 2.

Tabu Search algorithm

```
Given a solution s and a neighborhood structure k;
Initialize tabu list TL_z of t_min size;
elite ← s, counter ← 0, t_size ← t_min, AspirationCondition(s);
   While (counter ≤ MaxIters) do:
      Find s' ∈ N_λ(s) | s subject to tabu & aspiration conditions
      AllowedSet(s) ← s';
      s ← ChooseFirstImproving(AllowedSet(s));
         UpdateTabulist();
      If f(s) improves f(elite) then
         elite ← s; counter ← 0, t_size ← t_min, AspirationCondition(elite);
      Else
         counter ← counter + 1;
            If (t_size < t_max) then
               t_size ← t_size + 1;
            EndIf
         EndIf
   EndWhile
         s ← elite
```

4.2.2 Neighborhood structures

The VND scheme implemented in this work oscillates between seven neighborhood structures (λ_{max}=7) designed for transforming solutions with the purpose of finding improved configurations. As opposed to other metaheuristics based on VNS, where the neighborhood structures are defined by a single type of move, in our implementation, most of the neighborhood structures are defined by a set of types of moves.

Thus, the TS explores at each iteration the neighboring solutions of a current solution s. These are obtained applying the different types of moves defined on a neighborhood Nλ. Then, the first improving solution or the best solution of Nλ(s) is selected as the new current solution. As observed, it is obtained at each iteration by performing only one type of move of the neighborhood Nλ. Moreover, the selected move for finding the new solution in the neighborhood Nλ(s) may be different at each iteration of the TS.

The order of the neighborhoods was selected after some experiments considering the impact in the final solution and their cardinality. They are briefly described in the order adopted as follows:

- MIXTURE: This neighborhood structure is defined by Relocate (Savelsbergh, 1992) and Exchange (Kindervater and Savelsbergh, 1998) operators which are only applied on pairs of routes (inter-route). The Relocate operator aims to generate a solution by removing a customer from a route and inserting it into another route while the Exchange operator consists of swapping a pair of customers from two different routes.
- CROSSINGS: This neighborhood structure is also applied only on pairs of routes and is defined by the CROSS-exchange (Taillard et al., 1997) operator. The CROSS-exchange swaps segments of customers between two routes. The different segments may contain an arbitrary number of customers but due to the typically vast number of neighbors that would result, the segment length is limited to three customers. Thus, sets of 1-2, 2-2, 1-3, 2-3 and 3-3 swaps are defined and executed in the listed order.
- λ-OPT: This neighborhood structure is applied only on single routes (intra-route) and it aims to generate feasible solutions by examining all possible moves defined in 2-OPT (Croes, 1958), and 3-OPT (Lin, 1965) operators.

- INTERCHANGES: This neighborhood structure is similar to the Mixture type but is applied only on single routes (intra-route).
- INSERTIONS: This neighborhood structure is applied only on pairs of routes and is composed by double and triple insertion operators (Brandão, 2011). In a double or triple insertion move, the operation is similar to a single insertion except for removing a segment length of two or three consecutive customers respectively in a route.
- GENI INSERTION: This neighborhood structure is applied only on pairs of routes and is only composed of the Generalized Insertion (GENI) operator for the time window variant (Gendreau et al., 1998). It basically consists of removing a customer from a route and inserting it into any two customers of another route. If these customers are not consecutive, different moves in the adjacent customers of the route are performed to make the insertion possible. To reduce the computational effort required for the evaluation of the insertions a p-neighborhood is defined. Thus, the allowable insertions are restricted to between any two customers from the p-neighborhood of the customer to be inserted.
- GENI SWAP: This neighborhood structure is applied only on pairs of routes. It consists of removing a pair of customers belonging to two different routes and swapping them but performing the insertions through a GENI operator.

It is important to note that the proposed solution scheme includes a HL containing the list of the customers that are not served. The HL is similar to a “phantom” route which participates in the regular local search inducing an extended neighborhood search space for every inter-route operator (Lau et al., 2003). Consequently, there exist some additional moves:

- Relocate from holding list: Transferring customers from the holding list to an existing route.
- Relocate to holding list: Transferring customers from an existing route to the holding list.
- Exchange with holding list: Exchanging customers from an existing route with another group of customers in the holding list.

Thus, customers of a selected route will be searched completely for possible transfer to/from or for exchange with customers in the HL. The hierarchical cost structure of this problem, favors the transfer of customers from the HL to the routes, increasing the chances of finding high quality solutions to the problem. In addition, the HL favors the procedure to search for better solutions by going through the infeasible solution space (Lim and Zhang, 2007).

4.2.3 Resources Readjustment Mechanism

The Resources Readjustment Mechanism (RRM) introduced in this section is only applied to VRPs with limitations on RRs and/or CRs. After the VNDTS-HL terminates at a local optimum, the RRM is executed to modify the solution in order to escape from the local optimum and to diversify the search. The purpose of the RRM is to preserve some favorable features of the current solution in a similar way to the Reformation Phase in Paraskevopoulos et al. (2008) or the Adaptive Diversification Mechanisms in Wei et al. (2014).

The RRM is a Ruin and Recreate approach (Schrimpf et al., 2000) which partially destructs a solution and then tries to rebuild it to obtain a new admissible solution. Thus, in the first phase, a part of the solution is destroyed by deleting several chosen customers. It is reasonable to expect that the sequences of nodes of the routes requiring the same types of RRs are relatively good. In the same manner, as the feasibility of the solution is also determined by the CRs consumption, customers with higher demands in CRs are not desired. For these reasons, those customers belonging to the current solution with a greater value in the CR consumption than the third quartile are first penalized and moved to the HL. Then, the utilization of each RR $r \in \{1,\ldots,R\}$ is counted for each vehicle $k \in \{1,\ldots,K\}$ in each route identifying the resource less used on each route (r_min_k). In order to change the configuration in the use of RRs on the routes, all customers of a route k using r_min_k are extracted from the solution and also moved to the HL. We can reasonably hope that it is possible to find again an admissible solution with a new distribution in the use of RRs and CRs.

The second phase consists in the reconstruction of the ruined solution. For this purpose, the semi-parallel construction heuristic, introduced in Section 4.1 is applied but considering all current constructed routes at the same time. More specifically, the list of unserved customers is formed by the customers placed in the HL. Next, all customers in the HL are evaluated in all possible positions between two adjacent customers in all partially constructed routes from the ruined solution using the "greedy" function. The customer with the lowest value in the "greedy" function is placed in the specified position of the selected route and the procedure is repeated until no customer can be inserted in any route. Finally, the VNDTS-HL restarts the optimization procedure from the new solution until a new local optimum solution is reached.

Despite the RRM apparently degrading the quality of the solution, in terms of number of served customers, re-starting the optimization procedure from this state can guide the search toward better solutions from the solution space.

5. Experiments

This section describes the computational experiments carried out to validate the effectiveness of the algorithm presented in Section 4. The algorithm was programmed in C++ and run on a 3.30 GHz Intel® Core(TM) i5-2400 CPU. Section 5.1 describes the parameters used within the algorithm. In section 5.2, the sets of standard benchmark VRPTW instances from the literature are presented and a new set of HVRPTW-LR instances are also proposed. In section 5.3, experiments are conducted to demonstrate the performance of different components of the VNDTS-HL. Moreover, the VNDTS-HL is compared with other metaheuristic approaches in HVRPTW-LR instances. Finally, section 5.4 evaluates the performance of the algorithm, comparing the results obtained by the VNDTS-HL with the best metaheuristics developed for different VRPTW variants.

5.1 Parameter settings

In the first phase of the proposed methodology, the algorithm was tested with different parameter settings to identify the best parameter values. The results indicated that the parameter settings highly depend on the problem attributes and as in Paraskevopoulos et al. (2008) the alpha´s parameters are the most sensitive.

Based on these observations and in order not to present a large computational effort, the same parameter values adopted by these authors remain unchanged for all our experiments regardless of the type of problem solved. Thus, during the construction of initial solutions the values of $\alpha 1$ and $\alpha 3$ ranged between 0.1 - 0.3, $\alpha 2$ was always less than 0.3, while $\alpha 4$ was within 0.2–0.4. All ranges were explored via increments of 0.01. The number of solutions to be processed was established at 20.

In the second step of the algorithm, t_{min} and t_{max} were set to 10 and 30 respectively and the maximum number of iterations was set to 30. The p-neighborhood for GENI operators was set to five customers.

The algorithm is run once for every instance, with a maximum computer time (γ) of 1200 sec. For the new set of test cases for the HVRPTW-LR and based on a set of previous experiments, PCR and PCC were both set to 30. Table 1 summarizes the parameters used by the algorithm.

Table 1. Parameter settings

Heuristic parameters				Metaheuristic parameters		
α_1	α_2	α_3	α_4	*F*	*Iterations*	t_{min}-t_{max}
[0.1-0.3]	[0-0.3]	[0.1-0.3]	[0.2-0.4]	20	30	[15-30]
$\Delta\alpha$	*PCR*	*PCC*		*p-neighb.*	*γ (sec)*	
0.01	30	30		5	1200	

5.2 Test case collection

The computational experiments were performed on a collection of four sets of test cases. Section 5.2.1 describes the sets of benchmark-problem instances proposed in the literature while section 5.2.2 introduces a new set of test cases, which has been generated for solving HVRPTW-LR instances.

5.2.1 Test cases from the literature

In order to evaluate the VNDTS-HL, three different sets of benchmark instances from the literature are used in this work.

The first set comprises the m-VRPTW benchmark instances proposed by Lau et al. (2003) and Lim and Wang (2004), which derived from Solomon (1987) VRPTW instances. In total there are 56 problems grouped into 6 types of data sets. Customers are randomly distributed in instances of type R, clustered in type C and semi-clustered in instances of type RC. Problem sets shown by R1, C1 and RC1 have a short scheduling horizon and small vehicle capacities, contrary to R2, C2 and RC2.

Lau et al. (2003) are the first authors to generate a set of instances based on C1 set of Solomon (1987). They progressively reduced the number of vehicles in the fleet for every test case, from 10 vehicles where all customers are serviced, to 4 vehicles. The objective is to find the largest number of served customers with the available homogeneous fleet. Later, Lim and Wang (2004) extended the experiments to R1 and RC1 test sets of Solomon (1987).

The second set consists of the HVRPTW benchmark data sets proposed by Paraskevopoulos et al. (2008). They used the benchmark data sets proposed by Liu and Shen (1999) and extended a fixed fleet for each problem with the best known solutions of Liu and Shen (1999). In Liu and Shen (1999) test cases, different vehicles types, which differ in capacities and costs, are added to the classical Solomon (1987) VRPTW instances to solve the heterogeneous version of the FSMVRPTW. Finally, there are 24 benchmark instances grouped into the 6 types of data sets mentioned above. The total cost of a route is obtained by the sum of the fixed vehicle cost and of the total en-route time which includes travelling and waiting times.

The third set is composed of the HVRPTW instances proposed by Jiang et al. (2014). These authors generated a new set of HVRPTW instances derived from the Solomon (1987) test cases. They introduced a set of heterogeneous fleets with different fixed costs, variable costs and latest returning times to the depot.

5.2.2 New HVRPTW-LR test cases

As there are not benchmark problems for the HVRPTW-LR in the literature, and in order to evaluate the performance of the algorithm, a new set of test cases has been generated. The new instances are derived from Paraskevopoulos et al. (2008) test cases, in which additional data fields are provided.

In particular, four different test cases have been proposed which differ in the type of resources considered. In test cases (A), the number of available vehicles is reduced, in order to not be able to serve all customers in planning. Test cases (B) and (C) only consider the use of RRs or CRs at customer locations respectively but maintaining the available fleet. Finally, test cases (D) take into account the three types of limited resources at the same time (reduction of the number of available vehicles, RRs and CRs). In test cases (B), three different RRs (r1, r2 and r3) are considered for each problem. They could represent the necessary instruments to perform a service. The consumption of a type of CR is introduced in each customer in test cases (C), assuming that it adopts the same value as the delivery demand of the customer. The need to use RRs or CRs at a specific customer location is the same for all considered instances. The details of the HVRPTW-LR test cases are given in Appendix B.

As mentioned in Section 3, the objective of the HVRPTW-LR is a hierarchical objective function, where minimizing the total number of unserved customers is considered as the primary objective and minimizing the total travel cost as secondary. Particularly, in the HVRPTW-LR test cases, travel costs are obtained by the sum of the fixed vehicle cost and of the total trip duration without service times.

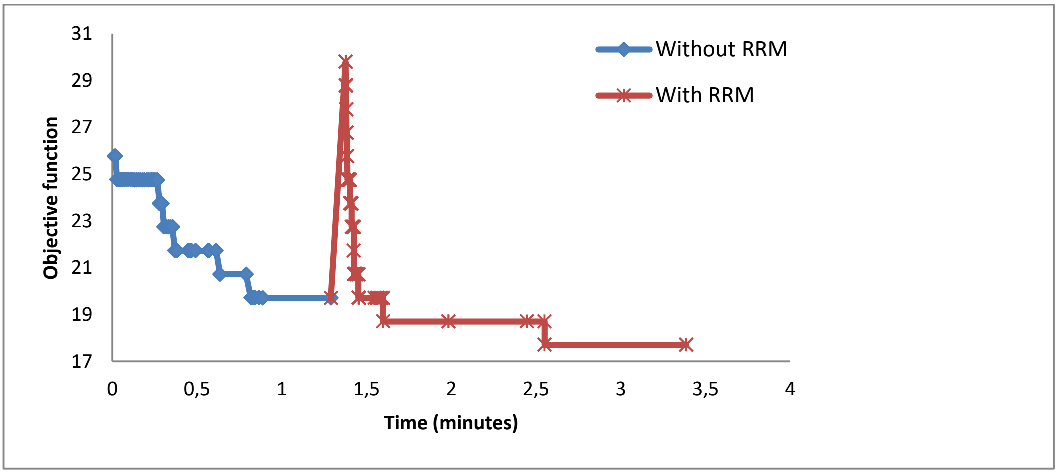
5.3 Computational studies in the HVRPTW-LR

For the evaluation of the proposed methodology in HVRPTW-LR instances, various computational experiments are conducted. In section 5.3.1, the performances of the RRM and the TS algorithm are compared within a computational study. Finally, additional experiments are introduced in section 5.3.2, to evaluate and compare the effectiveness of the VNDTS-HL, VND and TS algorithms.

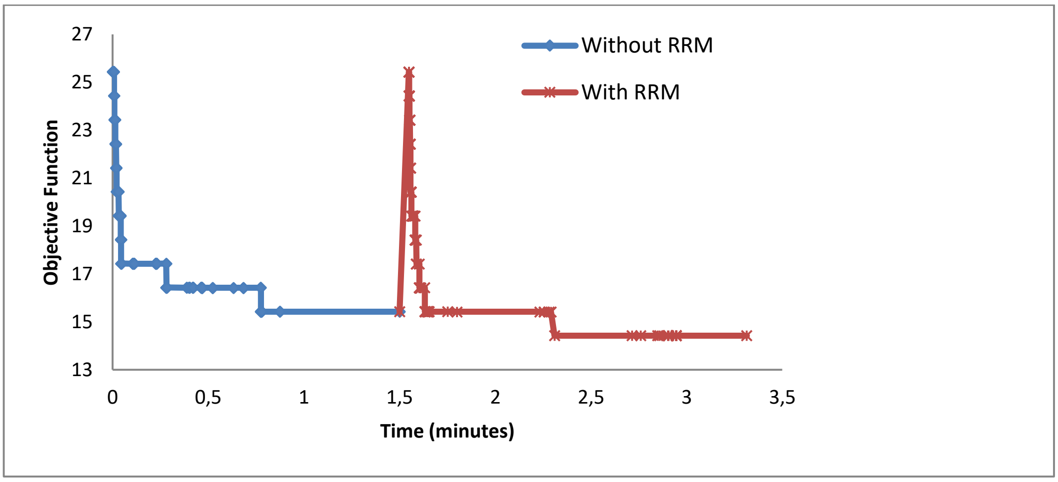
5.3.1 RRM and TS performances

In order to demonstrate the performance of the RRM and the TS scheme introduced in this work, some experiments are carried out in this section. In particular, C101 and RC101 problem instances from tests cases (D) are used for the experiments. The total cost to be minimized is given by the objective function of the HVRPTW-LR, with a value of ρ=0.0001.

Figures 1(a)-1(b) illustrate the total cost (y-axes) obtained during the search versus computational time consumed in minutes (x-axes). Specifically, figures indicate the search progress of the VNDTS-HL, with and without executing the RRM. It is observed how the RRM degrades the quality of the solution, in terms of number of served customers, to re-start the optimization procedure from a different solution. The idea is to destroy a part of the previous solution, maintaining the customers that fit well with others in the routes, and generate a new one with a new distribution in the use of RRs and CRs. Next, the optimization process is performed to find again a feasible solution in the solution space. In terms of efficiency, the RRM provides higher quality solutions in both instances. Particularly, in C101-LR(D) and RC101-LR(D) instances, the RRM achieves to increase the number of served customers by two and one customer respectively. However, the computational time required is longer, since it involves a more thorough exploration of the solution space.



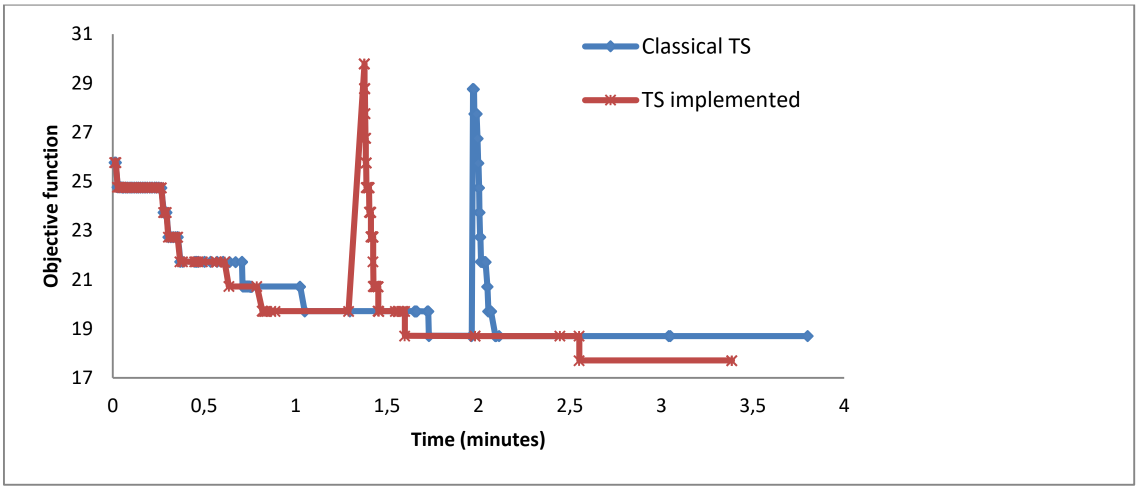
(a) C101-LR(D)



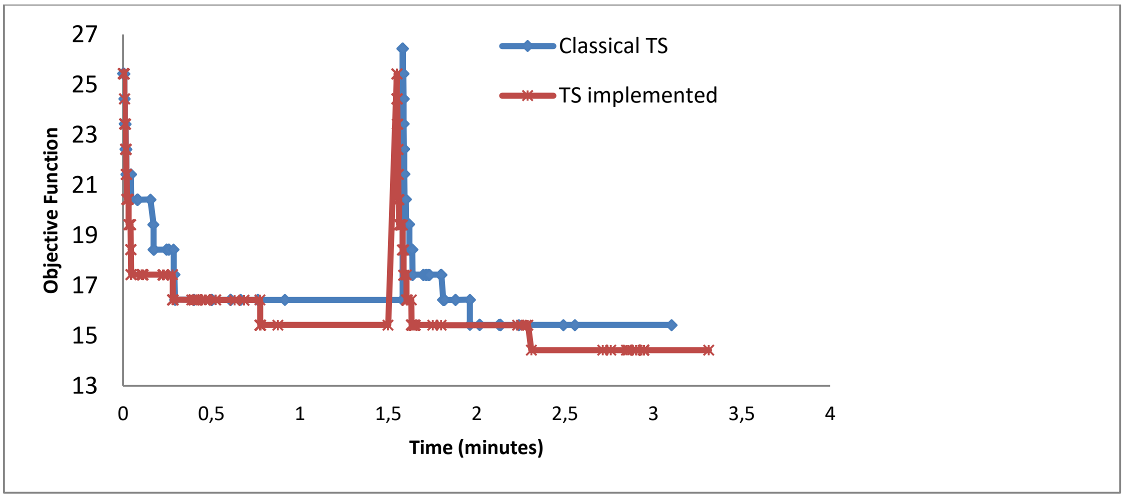
(b) RC101-LR(D)

Figure.-1 VNDTS-HL with and without RRM

Furthermore, in order to demonstrate the performance of the TS scheme implemented in this work (TS_imp), a comparison is made against the classical TS scheme. Figures 2(a)-2(b) indicate the search progress of the VNDTS-HL for both schemes. The observations are similar in the two problem instances examined. The performance of the TS_imp is clearly superior in terms of quality, with similar computational times and achieving to serve one more customer. According to figures 2(a)-2(b) it can be seen that the search progress of the TS_imp tends to generate better quality solutions when compared to the classical TS. A possible explanation to this fact is that the intensification mechanism introduced in the TS achieves a better exploitation of the local search in the neighborhood of a new obtained solution, avoiding getting trapped in local optima.



(a) C101-LR(D)



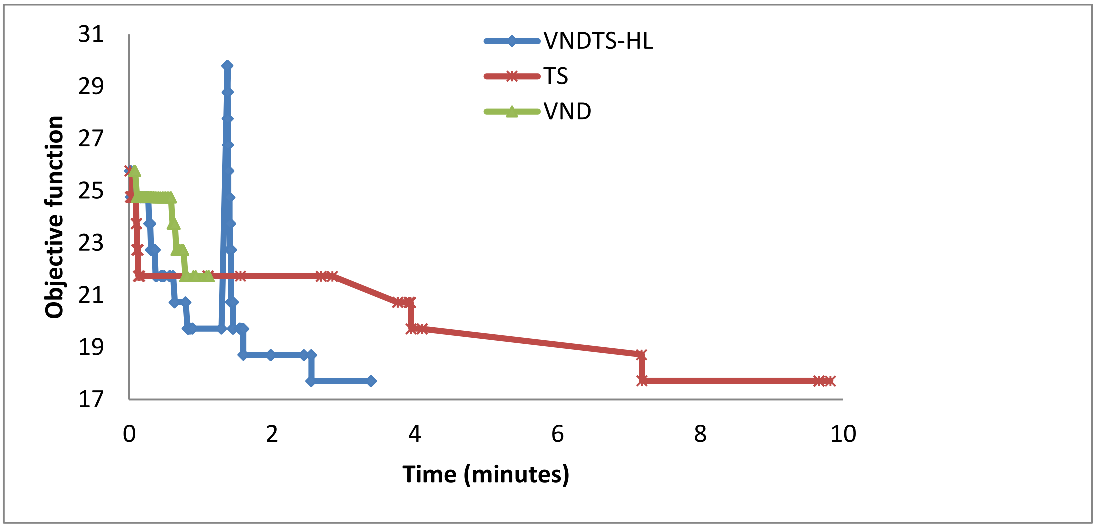
(b) RC101-LR(D)

Figure.-2 Comparison between classical TS and TS implemented.

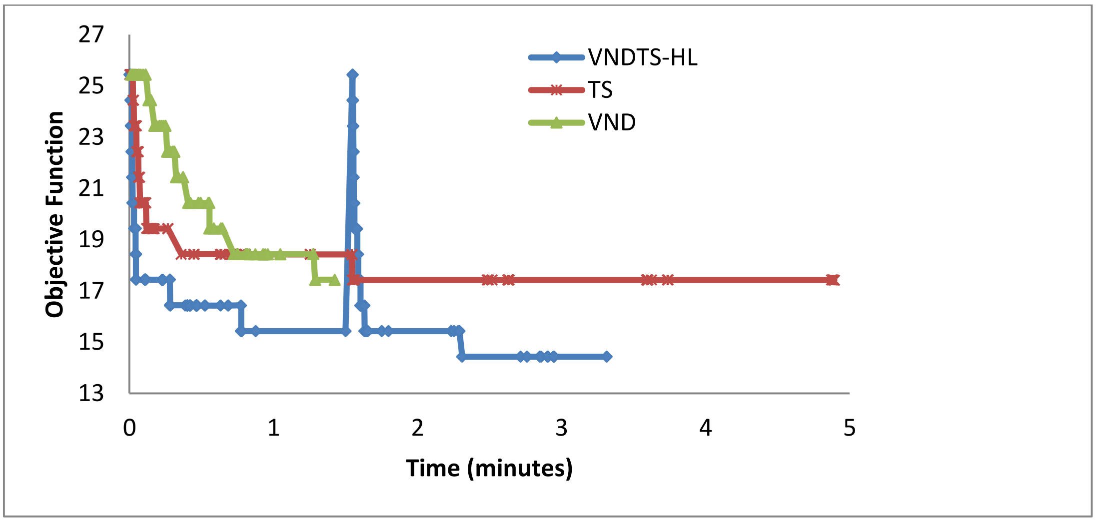
5.3.2 VNDTS-HL performance comparison in HVRPTW-LR instances

This section compares the VNDTS-HL with the VND, proposed by Mladenović and Hansen (1997), and the TS algorithm.

The basic VND considers a steepest descent procedure (known also as best improvement local search) for the local search, while the TS algorithm is implemented following the scheme described in section 4.2.1 with the described neighborhood structures.



(a) C101-LR(D)



(b) RC101-LR(D)

Figure.-3 Comparative performance of VND, TS and VNDTS-HL.

Figures 3(a)-3(b) illustrate the search progress of VND, TS and VNDTS-HL. They show that the VNDTS efficiently improves the performance of the VND and TS algorithm in both instances. Although the VND algorithm is the most effective in terms of computational time consumption, the obtained solutions are of lower quality. Contrary, TS requires larger computational times, since it performs a more thorough exploration of the neighborhood at each iteration. It is observed that the VNDTS-HL obtains solutions of higher quality in similar computational times. Moreover, the RRM allows to escape from the local minimum and to diversify the search in reasonable computing times.

5.4 Comparative analysis

As there are not benchmark instances for the HVRPTW-LR, the performance of the VNDTS-HL is assessed in test cases from two different VRPTW variants, comparing the results obtained with those presented in the literature. Thus, experimental results for m-VRPTW instances are shown in Section 5.4.1. In Section 5.4.2, a comparative analysis with the best results from the literature corresponding to the HVRPTW benchmark instances is performed. Finally, the test cases for the HVRPTW-LR are solved in Section 5.4.3.

5.4.1 Performance on m-VRPTW test cases

Table 2 compares the results obtained by the proposed metaheuristic, denoted as VNDTS-HL with the results obtained by Lim and Zhang (2007) (LZ), Lim and Wang (2004) (LW) and Lau et al. (2003) (LST) in C1 instances. Since LST did not propose results for either R1 or RC1 sets, Tables 3 and 4 only compares the results with those obtained by LZ and LW. These tables illustrate the maximum number of served customers for each instance as it is the primary objective function of the m-VRPTW. The first line of the table indicates the number of available vehicles while the first column of the table shows the instance category. The numbers in the brackets correspond in the following order to LZ, LW and LST published results. As mentioned above, results for R1 and RC1 instances are only provided for the first two works. For each problem a bold face refers to match with current best-known solution (BKS), whereas a bold face with a '*' indicates new BKS.

For C1 instances, the performance of the VNDTS-HL is superior to LW and LST algorithms, with 2 and 7 more customers served respectively. Only LZ approach produces a large number of total served customers than VNDTS-HL. Moreover, the VNDTS-HL achieved 1 new solution with an improvement of 1 customer.

For set R1 and RC1 the performance of VNDTS-HL is similar to C1 instances. The average performance of VNDTS-HL is better than LW approach, serving 16 more customers, except for the algorithm of LZ. Nevertheless, our algorithm found 4 new best solutions in RC instances. Resuming, the VNDTS-HL found 5 new best solutions and matched 122 out of 246 instances.

Problems of large dimension impose several additional difficulties for the VNDTS-HL, especially when the size of the fleet is reduced when it is compared with the total demand of the customers. The adaptation of random mechanisms to the local search phase of the VNDTS-HL, such as those used in the works published by LZ, LW or LST, could also have a rather positive influence in the final solution. They can help the VNDTS-HL to explore the solution space in a more effective way, allowing finding better solutions in m-VRPTW instances. Random mechanisms would allow insertions and/or exchanges of unserved customers from the HL, even if they may cause a decrease in the total number of served customers. However, they would not be as effective when solving other HVRPTW variants. Recall that the VNDTS-HL is executed in a deterministic way, and the main contribution of this work is to provide an effective method to solve the HVRPTW-LR, being flexible to provide reasonably good results for other variants.

5.4.2 Performance on HVRPTW test cases

Table 5 summarizes the average results obtained by VNDTS-HL compared to the current state-of-the-art solution approaches for the HVRPTW in Paraskevopoulos et al. (2008) instances. The first line of the table lists the authors using the following abbreviations: LS for Liu and Shen (1999), ReVNTS for Paraskevopoulos et al. (2008) and HEA for Koç et al. (2015).

The results show that the VNDTS-HL produced reasonability good results generating 9 BKSs (problems R104A, C102A, C103A, C104A, RC101A, RC102A, RC103A, RC104A, RC204A) and obtaining the same BKS on 4 problems (C101A, C201A, C202A, C204A). Although VNDTS-HL was capable of finding 13 BKSs in a reasonable computational time, the average percentage deviation (0.35%) shows that our algorithm is not as effective for the HVRPTW as HEA (0.12%). However, it is observed that the three algorithms are very accurate with an average deviation that is less than 0.50% for the 24 test problems. The VNDTS-HL presents a worst case performance of 1.63%.. Moreover, the VNDTS-HL even slightly outperforms HEA on C1A and RC1A instances (Table 5), where it produces competitive quality solutions. These results can also be extended to C2A instances, where the difference with HEA is less than 0.01% on average.

Furthermore, table 7 shows the results obtained by VNDTS-HL compared to the solutions obtained by Jiang et al. (2014). To the best of our knowledge, we are the first authors to apply the algorithm in their instances. The first line of the table lists the authors using the abbreviation of TSJ for Jiang et al. (2014). The first column of the table shows the instance category. Then, the fixed costs (FC), the variable costs (VC), the total costs (TC) and the percentage deviation (% Dev) of the total costs with respect to VNDTS-HL are showed. The last rows indicate the minimum, average and maximum percentage deviation over all problem instances and the average values for fixed and variable costs for each method.

The VNDTS-HL yields quality solutions with an average deviation of -0.51% and a worst-case performance of 1.59%. Moreover, the VNDTS-HL found 44 new best solutions and outperforms TSJ in almost all RC instances. Looking at the results obtained, on average the VNDTS-HL presents -1.35% of lower vehicle fixed costs than TSJ, which means that our algorithm obtains better fleet compositions to reduce the total costs. On the other hand, this fact adversely affects to the average variable costs, which only yields an increase of 0.40%.

5.4.3 Performance on HVRPTW-LR test cases

The details of the results obtained by VNDTS-HL for all HVRPTW-LR instances can be found in Appendix C.

6. Conclusions

Real-world VRPs present a variety of constraints and attributes that are not considered in traditional problem model formulations. In this study, a new variant of the HVRPTW is introduced to consider the limitation of resources in the design of the routes (HVRPTW-LR). It considers a fixed fleet of heterogeneous vehicles and the need to use

RRs and/or CRs for performing the services at customer sites. Due to the resource limitation (vehicles, RRs and CRs), the HVRPTW-LR includes the possibility of being unable to serve all customers in a route-planning and defines a hierarchical objective function, where maximizing the total number of served customers is considered as the primary objective, while minimizing the total travel cost as secondary.

The complexity of the HVRPTW-LR is tackled by a hybrid variable neighborhood descent metaheuristic based on a TS algorithm for the exploration of the neighborhood and a HL (VNDTS-HL). In the first phase, several solutions are obtained with different sequences of parameter values, using a semi-parallel construction heuristic. In the second phase, a hybrid VNDTS-HL is proposed for processing a subset of initial solutions. A HL which holds temporarily unserved customers is integrated within the VNDTS-HL to introduce flexibility for solving instances in which it is impossible to serve all customers due to a limitation in the available resources. To improve the performance of the algorithm, a scheme that provides a balance between diversification and intensification search strategies, is incorporated to the TS. Moreover, a new mechanism, called RRM, is proposed when considering RRs and/or CRs. The positive impact of both mechanisms has been observed through computational experiments.

Due to a lack of benchmark data sets for the problem under investigation, computational experiments were carried out on benchmark instances with a limited number of vehicles. Experimental results show that the algorithm is very robust, showing a good performance on a wide range of very different problems. To fully evaluate the effectiveness of the algorithm, new test cases for the HVRPTW-LR were generated and solved.

The HVRPTW-LR provides a research gap into some real-world practical applications; i.e. service companies that are responsible for the metrological control of measuring equipment at customer sites. They design routes accordingly to the daily limitation on the number of RRs (measure instruments or vehicles) and CRs in the central laboratory. Thus, the different routes designed in a specific day have to share these scarce resources. Moreover, some customers must be served in a predefined TW, early in the workday, in order to not interfere with the opening hours. The home care providers industry or the industrial maintenance services providers sector are other real-life examples that includes the consideration of the attributes presented in the HVRPTW-LR. Managers from these types of companies can use the proposed algorithm to increase the companies´ capabilities (reducing the number of unserved orders in a route planning) and also to reduce the total costs.

Performance improvement in operation management in environments with limited resources needs much attention and study. The further research will be the development of more effective algorithms to consider skill requirements in customer orders as well to incorporate a weekly route planning.

Table 2. Results for m-VRPTW for C1 instances

Nº Veh.	10	9	8	7	6	5	4
C101	**100 (100-100-100)**	**92 (92-92-92)**	**84 (84-84-84)**	**75 (75-75-75)**	**66 (66-66-66)**	**57 (57-57-57)**	**47 (47-47-47)**
C102	**100 (100-100-100)**	**99 (99-99-99)**	92 (**93**-92-92)	83 (**84**-83-**84**)	**72 (72-72-72)**	**61*** (60-60-60)	**49 (49-49-49)**
C103	**100 (100-100-100)**	**99 (99-99-99)**	93 (**94**-93-93)	**84 (84-84-84)**	72 (**73**-72-72)	**61** (**61**-60-60)	**49** (**49**-48-48)
C104	**100 (100-100-100)**	**99 (99-99-99)**	93 (**94-94-94**)	**84 (84-84-84)**	**73(73-73**-72)	**61** (**61-61**-60)	**49** (**49-49**-48)
C105	**100 (100-100-100)**	92 (**93**-92-92)	**84 (84-84-84)**	**75 (75-75-75)**	**66 (66-66-66)**	**57 (57-57-57)**	**47 (47-47-47)**
C106	**100 (100-100-100)**	93 (**94**-93-93)	**86 (86-86-86)**	**77 (77-77-77)**	**68 (68-68-68)**	**58 (58-58-58)**	**47 (47-47-47)**
C107	**100 (100-100-100)**	93 (**94**-93-93)	85 (**86**-85-85)	76 (**77**-76-76)	**67** (**67-67**-66)	**57 (57-57-57)**	**47 (47-47-47)**
C108	**100 (100-100-100)**	96 (**98**-96-96)	87 (**89**-87-87)	78 (**80**-78-78)	69 (**70**-69-69)	**59 (59-59-59)**	**48 (48-48-48)**
C109	**100 (100-100-100)**	**99 (99-99-99)**	**92 (92-92-92)**	82 (**83**-82-82)	**72 (72-72-72)**	**61** (**61-61**-60)	**49** (**49-49**-48)
TOTAL LZ=4878; LW=4859 ; LST=4854; VNDTS-HL=4861							

Table 3. Results for m-VRPTW for R1 and RC1 instances

Nº Veh.	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4
R101	**100(100-100)**	98(**99**-98)	97 (**98**-97)	95 (**96**-94)	91(**93**-91)	88(**91**-88)	86(**87**-86)	81(**82**-81)	76(**77**-75)	70(**71**-69)	**65(65-65)**	**59(59-59)**	**53(53-53)**	**46(46-46)**	**39(39-39)**	**32(32-32)**
R102			**100(100-100)**	**99(99-99)**	97(**98-98**)	95(**96**-95)	92(**94**-92)	90(**91**-89)	88(**89**-85)	84(**85**-81)	79(**81**-78)	74(**75**-72)	67(**68**-67)	60(**61**-59)	**53(53-53)**	**45(45-45)**
R103						**100(100-100)**	99(**100**-99)	98(**99**-98)	97(**98**-96)	93(**96**-93)	89(**91**-88)	82(**85**-83)	74(**78**-74)	67(**69**-67)	58(**60**-58)	48(**49**-48)
R104										**100(100-100)**	98(**100**-98)	90(**92**-90)	83(**84**-83)	73(**74**-73)	62(**64**-62)	52(**53**-52)
R105						**100(100-100)**	98(**99**-98)	95(**97**-96)	**92(92-92)**	**87(87**-86)	80(**81**-80)	73(**74**-73)	66(**67**-66)	**59(59-59)**	50(**51-51**)	41(**42**-41)
R106							**100(100-100)**	99(**100**-99)	97(**99**-96)	93(**96**-92)	87(**90**-86)	80(**83**-80)	74(**75**-73)	66(**67**-66)	57(**58**-57)	**48(48-48)**
R107									**100(100-100)**	99(**100**-99)	95(**96**-95)	88(**89**-88)	80(**81-81**)	71(**72**-71)	**62(62-62)**	**50(50-50)**
R108											**100(100-100)**	92(**94**-93)	**85(85-85)**	74(**75**-74)	63(**64**-63)	**53(53-53)**
R109									**100(100-100)**	94(**97**-95)	89(**90**-89)	82(**83**-82)	**75(75**-74)	**67**(66-**67**)	**58**(57-**58**)	**47(47-47)**
R110										**100(100-100)**	94(**95**-94)	86(**87**-86)	76(**79**-76)	68(**69-69**)	59(**60**-59)	**49**(48-**49**)
R111									**100(100-100)**	99(**100**-99)	94(**96**-95)	87(**89**-88)	**80(80-80)**	70(**71**-70)	**61(61-61)**	**50(50-50)**
R112										**100(100-100)**	98(**100**-99)	91(**92-92**)	**83(83-83)**	**73(73-73)**	62(**63**-62)	**51(51-51)**
RC101						**100(100-100)**	97(**98**-97)	93(**94**-93)	89(**90**-89)	**84(84-84)**	**78(78-78)**	**71(71-71)**	63(63-**64**)	**56(56-56)**	**49(49-49)**	**40(40-40)**
RC102								**100(100-100)**	95(**97**-95)	89(**92**-89)	85(**86**-85)	**79(79**-77)	71(**72**-71)	**63(63-63)**	**54(54-54)**	**44(44-44)**
RC103									**100(100-100)**	97(**99**-98)	91(**94**-91)	85(**86**-85)	**77(77-77)**	67(**68**-67)	**58(58**-57)	**47(47-47)**
RC104										**100(100-100)**	96(**98**-95)	89(**91**-89)	81(**82**-81)	71(**72-72**)	**61(61-61)**	**50(50-50)**
RC105						**100(100-100)**	99(**100**-99)	97(**98-98**)	94(**96**-94)	89(**91**-89)	83(**85**-82)	**77(77-77)**	**70(70-70)**	**62**(61-**62**)	**53(53-53)**	42(**44**-42)
RC106								**100(100-100)**	98(**100**-99)	94(**95-95**)	87(**89**-87)	**80(80-80)**	71(**72**-71)	62(**63**-62)	**53(53-53)**	**43(43-43)**
RC107									**100(100-100)**	97(**99**-97)	91(**93**-90)	**84(84-84)**	**77***(76-76)	**67***(65-66)	**56(56-56)**	**46(46-46)**
RC108										**100(100-100)**	94(**96**-93)	87(**88**-86)	**78(78-78)**	**70***(68-69)	**59***(58-58)	**47(47-47)**
TOTAL LZ=14380; LW=14220 ; VNDTS-HL=14236																

Table 4. Results for HVRPTW instances

Instance	Fleet	ReVNTS			HEA			VNDTS-HL			BKS
		Mix	TC	Dev(%)	Mix	TC	Dev(%)	Mix	TC	Dev(%)	TC
R101A	$A^{1} B^{11} C^{11} D^{1}$	$B^{10} C^{11} D^{1}$	**4583.99**	0.00	$B^{10} C^{11} D^{1}$	4588.76	0.10	$B^{10} C^{11} D^{1}$	4654.72	1.52	4583.99
R102A	$A^{1} B^{4} C^{14} D^{2}$	$B^{3} C^{14} D^{2}$	4420.68	1.00	$A^{1} B^{4} C^{13}$	**4376.54**	0.00	$B^{3} C^{14} D^{2}$	4449.05	1.63	4376.54
R103A	$B^{7} C^{15}$	$B^{6} C^{15}$	**4195.05**	0.00	$B^{6} C^{15}$	4201.71	0.16	$B^{6} C^{15}$	4198.80	0.09	4195.05
R104A	$B^{9} C^{14}$	$B^{8} C^{14}$	4065.52	1.21	$B^{9} C^{13}$	4027.69	0.29	$B^{7} C^{14}$	**4016.17***	0.00	4016.17
C101A	$A^{1} B^{10}$	B^{10}	**8828.93**	0.00	B^{10}	**8828.93**	0.00	B^{10}	**8828.93**	0.00	8828.93
C102A	A^{19}	A^{19}	7137.79	0.26	A^{19}	7153.13	0.47	A^{19}	**7119.35***	0.00	7119.35
C103A	A^{19}	A^{19}	7143.88	0.54	A^{19}	7122.57	0.24	A^{19}	**7105.39***	0.00	7105.39
C104A	A^{19}	A^{19}	7104.96	0.33	A^{19}	7083.74	0.03	A^{19}	**7081.51***	0.00	7081.51
RC101A	$A^{7} B^{7} C^{7}$	$A^{4} B^{7} C^{7}$	5279.92	0.42	$A^{4} B^{7} C^{7}$	5266.36	0.16	$A^{4} B^{7} C^{7}$	**5257.67***	0.00	5257.67
RC102A	$A^{5} B^{6} C^{8}$	$A^{4} B^{5} C^{8}$	5149.95	1.30	$A^{4} B^{5} C^{8}$	5099.55	0.32	$A^{2} B^{6} C^{8}$	**5083.08***	0.00	5083.08
RC103A	$A^{11} B^{2} C^{8}$	$A^{10} B^{2} C^{8}$	5002.41	0.23	$A^{10} B^{2} C^{8}$	4991.29	0.01	$A^{10} B^{2} C^{8}$	**4990.94***	0.00	4990.94
RC104A	$A^{2} B^{13} C^{3} D^{1}$	$A^{2} B^{13} C^{3}$	5024.25	0.36	$A^{2} B^{13} C^{3}$	5016.97	0.22	$A^{2} B^{13} C^{3}$	**5006.16***	0.00	5006.16
R201A	A^{5}	A^{5}	**3779.12**	0.00	A^{5}	3782.49	0.09	A^{5}	3823.44	1.16	3779.12
R202A	A^{5}	A^{5}	**3578.91**	0.00	A^{5}	3583.92	0.14	A^{5}	3616.66	1.04	3578.91
R203A	$A^{4} B^{1}$	$A^{4} B^{1}$	3582.51	0.80	$A^{4} B^{1}$	**3553.92**	0.00	$A^{4} B^{1}$	3590.10	1.01	3553.92
R204A	A^{5}	A^{5}	3143.68	1.97	A^{5}	**3081.80**	0.00	A^{5}	3092.29	0.34	3081.80
C201A	$A^{4} B^{1}$	$A^{4} B^{1}$	**6140.64**	0.00	$A^{4} B^{1}$	**6140.64**	0.00	$A^{4} B^{1}$	**6140.64**	0.00	6140.64
C202A	$A^{1} C^{3}$	$A^{1} C^{3}$	7752.88	1.66	$A^{1} C^{3}$	**7623.96**	0.00	$A^{1} C^{3}$	**7623.96**	0.00	7623.96
C203A	$C^{2} D^{1}$	$C^{2} D^{1}$	**7303.37**	0.00	$C^{2} D^{1}$	**7303.37**	0.00	$C^{2} D^{1}$	7303.70	0.00	7303.37
C204A	A^{5}	A^{5}	5721.09	0.71	A^{5}	**5680.46**	0.00	A^{5}	**5680.46**	0.00	5680.46
RC201A	$C^{1} E^{3}$	$C^{1} E^{3}$	**5523.15**	0.00	$C^{1} E^{3}$	5534.59	0.21	$C^{1} E^{3}$	5550.88	0.50	5523.15
RC202A	$A^{1} C^{1} D^{1} E^{2}$	$A^{1} C^{1} D^{1} E^{2}$	**5132.08**	0.00	$A^{1} C^{1} D^{1}$	5150.23	0.35	$A^{1} C^{1} D^{1} E^{2}$	5192.38	1.16	5132.08
RC203A	$A^{1} B^{1} C^{5}$	$A^{1} B^{1} C^{5}$	4508.27	0.81	$A^{1} B^{1} C^{5}$	**4471.92**	0.00	$A^{1} B^{1} C^{5}$	4473.13	0.03	4471.92
RC204A	$A^{14} B^{2}$	$A^{14} B^{2}$	4252.87	0.29	$A^{14} B^{2}$	4241.83	0.03	$A^{14} B^{2}$	**4240.35***	0.00	4240.35
Average				0.50			0.12			0.35	
Max				1.97			0.47			1.63	
Nº BKS				9			9			13	

Table 5. Average results for HVRPTW

Instance set	ReVNTS		HEA		VNDTS-HL
	TC	% Dev.	TC	% Dev.	TC
R1A (4)	4316.31	0.31	4298.68	0.72	4329.68
C1A (4)	7553.89	-0.27	7547.09	-0.18	7533.79
RC1A (4)	5114.13	-0.58	5093.54	-0.18	5084.46
R2A (4)	3521.06	0.27	3500.53	0.85	3530.62
C2A (4)	6729.50	-0.63	6687.11	0.00	6687.19
RC2A (4)	4854.09	0.14	4849.64	0.27	4864.18

Table 6. Results for HVRPTW Jiang et al (2014) instances

Instance set	TSJ			Dev.	VNDTS-HL			Instance set	TSJ			Dev.	VNDTS-HL		
	F.C	V.C	T.C.		F.C	V.C	T.C.		F.C	V.C	T.C.		F.C	V.C	T.C.
HC101	670	1215.33	1885.33	0,66%	650	1247.81	1897.81	**HR112**	2540	1712.71	4252.71	-0.15%	2540	1706.44	***4246.44**
HC102	670	1220.66	1890.66	-2,37%	670	1176.96	***1846.96**	**HR201**	255	1510.74	1765.74	-3.97%	320	1378.34	***1698.34**
HC103	650	1258.04	1908.04	-2,75%	670	1186.91	***1856.91**	**HR202**	255	1281.81	1536.81	-0.89%	250	1273.27	***1523.27**
HC104	670	1139.78	1809.78	-1,07%	670	1120.62	***1790.62**	**HR203**	255	1082.39	1337.39	-0.19%	295	1039.82	***1334.82**
HC105	670	1184.73	1854.73	0,38%	670	1191.74	1861.74	**HR204**	275	839.94	1114.94	1.17%	275	853.15	1128.15
HC106	670	1210.64	1880.64	-0,81%	670	1195.57	***1865.57**	**HR205**	270	993.91	1263.91	1.02%	225	1051.96	1276.96
HC107	670	1169.52	1839.52	0,41%	670	1177.16	1847.16	**HR206**	270	910.44	1180.44	0.80%	225	964.99	1189.99
HC108	670	1156.49	1826.49	-0,21%	670	1152.65	***1822.65**	**HR207**	225	877.06	1102.06	-0.90%	225	867.23	***1092.23**
HC109	670	1129.22	1799.22	-0,06%	670	1128.19	***1798.19**	**HR208**	225	782.00	1007.00	-0.38%	225	778.14	***1003.14**
HC201	520	793.28	1313.28	0.45%	520	799.21	1319.21	**HR209**	225	894.04	1119.04	0.43%	225	898.84	1123.84
HC202	540	743.58	1283.58	-1.23%	540	727.98	***1267.98**	**HR210**	275	1032.53	1307.53	-0.02%	250	1057.26	***1307.26**
HC203	540	719.97	1259.97	-0.21%	540	717.36	***1257.36**	**HR211**	225	785.22	1010.22	-0.75%	225	777.71	***1002.71**
HC204	540	716.09	1256.09	1.59%	540	736.36	1276.36	**HRC101**	3330	2373.97	5703.97	-1.63%	3270	2342.57	***5612.57**
HC205	520	805.84	1325.84	-0.24%	540	782.67	***1322.67**	**HRC102**	3420	2136.02	5556.02	-0.12%	3360	2189.20	***5549.20**
HC206	540	723.63	1263.63	-0.10%	540	722.39	***1262.39**	**HRC103**	3510	1928.89	5438.89	-0.26%	3390	2035.00	***5425.00**
HC207	520	787.35	1307.35	-3.36%	540	724.91	***1264.91**	**HRC104**	3450	1881.41	5331.41	-0.44%	3390	1918.05	***5308.05**
HC208	500	690.81	1190.81	-0.35%	500	686.67	***1186.67**	**HRC105**	3570	2135.79	5705.79	-1.17%	3390	2249.81	***5639.81**
HR101	3030	2095.52	5125.52	-0.98%	2870	2205.85	***5075.85**	**HRC106**	3510	2018.42	5528.42	-0.99%	3360	2114.27	***5474.27**
HR102	3040	1942.39	4982.39	-0.54%	3040	1915.72	***4955.72**	**HRC107**	3450	2001.31	5451.31	-1.67%	3330	2031.62	***5361.62**
HR103	2880	1781.22	4661.22	0.04%	2870	1793.10	4663.10	**HRC108**	3510	1812.31	5322.31	-0.66%	3390	1897.23	***5287.23**
HR104	2940	1590.55	4530.55	-2.95%	2740	1660.53	***4400.53**	**HRC201**	2900	1601.65	4501.65	0.31%	2900	1615.62	4515.62
HR105	2700	1870.39	4570.39	-0.43%	2700	1850.89	***4550.89**	**HRC202**	2950	1458.53	4408.53	-0.12%	2950	1453.22	***4403.22**
HR106	2610	1821.31	4431.31	0.03%	2620	1812.72	4432.72	**HRC203**	2900	1421.87	4321.87	-0.03%	2950	1370.45	***4320.45**
HR107	2700	1691.14	4391.14	-0.07%	2680	1707.93	***4387.93**	**HRC204**	2900	1406.65	4306.65	-0.67%	2900	1378.09	***4278.09**
HR108	2580	1700.05	4280.05	-0.39%	2540	1723.42	***4263.42**	**HRC205**	2900	1552.88	4452.88	-0.10%	2900	1548.51	***4448.51**
HR109	2540	1799.86	4339.86	-0.32%	2540	1785.92	***4325.92**	**HRC206**	2900	1519.09	4419.09	-0.69%	2900	1488.86	***4388.86**
HR110	2540	1732.25	4272.25	-0.39%	2540	1715.82	***4255.82**	**HRC207**	2950	1393.55	4343.55	-0.10%	2900	1439.35	***4339.35**
HR111	2670	1696.32	4366.32	-0.88%	2650	1678.41	***4328.41**	**HRC208**	2900	1376.15	4276.15	-0.15%	2900	1369.55	***4269.55**
Min				-3.97%											
Average				-0.51%											
Max				1.59%											
Av. FC	1728,66			-1.35%	1705.71										
Av. VC		1376.92		0.40%		1382.39									

Acknowledgments

The research of Molina, Eguia and Racero has been fully funded by the Andalusia Government through grants P10-TEP-6332.

Appendix A. Abbreviations and symbols used in the paper

Abbreviation	Description
BKS	best known solution
CR	consumable resource
FC	fixed costs
FSMVRP	fleet size and mix vehicle routing problem
FSMVRPTW	fleet size and mix vehicle routing problem with time windows
GENI	generalized insertion
HEA	hybrid evolutionary algorithm of Koç et al. (2015)
HF-VRP	heterogeneous fleet vehicle routing problem
HL	holding list
HVRP	heterogeneous vehicle routing problem
HVRPTW	heterogeneous vehicle routing problem with time windows
HVRPTW-LR	heterogeneous vehicle routing problem with time windows and a limited number of resources
LS	Liu and Shen (1999)
LST	Lau et al. (2003)
LZ	Lim and Zhang (2007)
LW	Lim and Wang (2004)
m-VRPTW	vehicle routing problem with time windows and a limited number of vehicles
PCR	renewable penalty cost
PCC	consumable penalty cost
ReVNTS	reactive variable neighborhood tabu search algorithm of Paraskevopoulos et al. (2008)
RR	renewable resource
RRM	resources readjustment mechanism
SBS/RS	Shuttle-Based Storage and Retrieval Systems
TC	total cost
TL	tabu list
TS	tabu search
TS_imp	tabu search implemented
TSJ	tabu search of Jiang et al. (2014)
TW	time windows
VC	variable costs
VND	variable neighborhood descent
VNDTS-HL	variable neighborhood descent tabu search algorithm with holding list
VNS	variable neighborhood search
VRP	vehicle routing problem
VRPTW	vehicle routing problem with time windows

Symbol	Definition
%Dev	percentage deviation of the costs with respect to the best known solution
c	consumable resource
C	number of consumable resources
$C_{ij,u}^{n}$	metric *n* of the greedy function
D_i	load demanded by node *i*
EW_i	earliest time to begin the service at node *i*
F	set of initial solutions
K	number of vehicles
LW_i	latest time to begin the service at node *i*
Mix	composition fleet
N	number of nodes
N_λ	set of neighborhood structures
NR_r	number of renewable resource *r* available in the route planning
NC_c	quantity of consumable resource *c* available in the route planning
Q_k	capacity of vehicle *k*
r	renewable resource
r_min_k	resource less used on route performed by vehicle k
R	number of renewable resources
s	solution
SC_{ic}	quantity of consumable resource *c* demanded by node *i*
SR_{ir}	binary value equal to 1 if node *i* needs the renewable resource *r* to be served
ST_i	service time in node *i*
t_{size}	tabu tenure
t_{min}	lower value of the tabu tenure
t_{max}	higher value of the tabu tenure
TD_{ij}	travel distance from node *i* to node *j*
$TMax_k$	latest returning time to the depot of vehicle *k*
TT_{ij}	travel time from node *i* to node *j*
α_i	weight that defines the relative contribution of the metric *i*
β_k	variable cost of vehicle *k*
λ	neighborhood structure
λ_{max}	number of neighborhood structures
ξ	conversion factor
γ	maximum computer time
Φ^{uk}_{ij}	greedy function that measures the cost of inserting node *u* between nodes *i* and *j* in vehicle *k*
ρ	small positive scalar

Appendix B. HVRPTW-LR test cases

	Test cases (A)	Test cases (B)	Test cases (C)		Test cases (A)	Test cases (B)	Test cases (C)
	Available Fleet	Available quantity of RRs(r_1, r_2, r_3)	Available quantity of CRs		Available Fleet	Available quantity of RRs(r_1, r_2, r_3)	Available quantity of CRs
R101-LR	B^7 C^9 D^1	(15 15 15)	1100	**R201-LR**	A^4	(3 3 3)	1100
R102-LR	B^3 C^{13}	(14 14 14)	1100	**R202-LR**	A^4	(3 3 3)	1100
R103-LR	B^4 C^{12}	(9 9 9)	1100	**R203-LR**	A^4	(2 2 2)	1100
R104-LR	B^6 C^{11}	(8 8 8)	1100	**R204-LR**	A^4	(2 2 2)	1100
C101-LR	B^8	(7 7 7)	1360	**C201-LR**	A^4	(3 3 3)	1360
C102-LR	A^{14}	(12 12 12)	1360	**C202-LR**	A^1 C^2	(2 2 2)	1360
C103-LR	A^{14}	(7 7 7)	1360	**C203-LR**	C^1 D^1	(1 1 1)	1360
C104-LR	A^{14}	(7 7 7)	1360	**C204-LR**	A^4	(2 2 2)	1360
RC101-LR	A^1 B^6 C^6	(11 11 11)	1300	**RC201-LR**	C^1 E^2	(2 2 2)	1300
RC102-LR	A^5 B^2 C^7	(12 12 12)	1300	**RC202-LR**	A^1 C^1 D^1 E^1	(3 3 3)	1300
RC103-LR	A^{11} B^2 C^5	(7 7 7)	1300	**RC203-LR**	A^1 C^4	(3 3 3)	1300
RC104-LR	A^1 B^8 C^3 D^1	(6 6 6)	1300	**RC204-LR**	A^9 B^2	(7 7 7)	1300

RRs utilization and CRs consumption at customer sites

RRs (Test cases B)			CRs (Test cases C)	
Node number range interval	RRs utilization at Even nodes	RRs utilization at Odd nodes	Node number range interval	CR consumption at customer i
1-25	---	r_1 & r_2	1-100	CR(i)=D(i)
26-50	r_1	r_1 & r_3		
51-75	r_2	r_2 & r_3		
76-100	r_3	r_1 & r_2 & r_3		

Appendix C. Computational results on HVRPTW-LR test cases

	(A) Fleet Reduction		(B) Limitation on RRs		(C) Limitation on CRs		(D)=(A)+(B)+(C)	
	NC	TC	NC	TC	NC	TC	NC	TC
R101-LR	90	3819.70	98	4928.07	88	3668.42	88	3756.00
R102-LR	90	3599.60	98	4581.91	88	3543.89	88	3564.45
R103-LR	90	3474.64	87	3588.64	88	3197.42	87	3588.64
R104-LR	90	3196.07	84	3250.79	88	3054.78	84	3250.79
C101-LR	84	7031.17	97	9863.74	88	8435.34	83	7082.25
C102-LR	89	5459.32	100	7704.01	88	5419.97	88	5419.97
C103-LR	90	5525.27	88	5712.69	88	5351.49	87	5783.41
C104-LR	89	5319.45	88	5454.87	88	5262.69	88	5672.28
RC101-LR	88	4384.32	95	5398.56	87	4379.19	86	4234.33
RC102-LR	90	4259.91	100	5361.92	87	4055.50	87	4055.50
RC103-LR	88	4078.71	88	4669.68	87	3788.01	80	4116.80
RC104-LR	88	3888.96	82	4125.32	87	3738.84	82	4125.32
R201-LR	92	3417.73	95	4667.29	88	3169.85	88	3610.76
R202-LR	92	3123.38	94	4164.54	88	2946.44	88	3450.48
R203-LR	92	2895.94	84	3671.93	88	2775.11	82	3534.73
R204-LR	92	2657.16	81	2893.13	88	2533.92	81	2999.94
C201-LR	95	5181.30	100	8384.32	88	4955.47	88	5423.21
C202-LR	95	5782.10	87	8278.95	88	5639.35	83	6373.50
C203-LR	70	5136.18	58	4899.07	88	7309.18	58	5589.27
C204-LR	95	4724.35	84	4975.28	88	4638.14	84	5057.74
RC201-LR	85	4266.03	82	5776.60	87	5408.39	78	4369.93
RC202-LR	87	4068.83	100	5614.59	87	4081.08	87	4081.08
RC203-LR	87	3598.16	90	4581.26	87	3577.35	87	3999.88
RC204-LR	87	3172.29	94	4100.71	87	3140.62	87	3608.30

References

Baldacci, R., Battarra, M., & Vigo, D. (2008). Routing a heterogeneous fleet of vehicles. In The vehicle routing problem: latest advances and new challenges (pp. 3-27). Springer US.

Borovinšek, M., Ekren, B. Y., Burinskienė, A., & Lerher, T. (2017). Multi-objective optimisation model of shuttle-based storage and retrieval system. Transport, 32(2), 120-137.

Brandão, J. (2006). A new tabu search algorithm for the vehicle routing problem with backhauls. European Journal of Operational Research, 173(2), 540-555.

Brandão, J. (2009). A deterministic tabu search algorithm for the fleet size and mix vehicle routing problem. European Journal of Operational Research, 195(3), 716-728.

Brandão, J. (2011). A tabu search algorithm for the heterogeneous fixed fleet vehicle routing problem. Computers & Operations Research, 38(1), 140-151.

Bräysy, O. (2001). Local search and variable neighborhood search algorithms for the vehicle routing problem with time windows. Acta Wasaensia 87, University Wasaenis, Vaasa .

Bräysy, O. (2003). A reactive variable neighborhood search for the vehicle-routing problem with time windows. INFORMS Journal on Computing, 15(4), 347-368.

Carlo, H. J., & Vis, I. F. (2012). Sequencing dynamic storage systems with multiple lifts and shuttles. International Journal of Production Economics, 140(2), 844-853.

Chen, P., Huang, H. K., & Dong, X. Y. (2010). Iterated variable neighborhood descent algorithm for the capacitated vehicle routing problem. Expert Systems with Applications, 37(2), 1620-1627.

Croes, G. A., (1958). A method for solving traveling-salesman problems. Operations Research, 6(6), 791-812.

De Armas, J., Melián-Batista, B., Moreno-Pérez, J. A., & Brito, J. (2015). GVNS for a real-world rich vehicle routing problem with time windows. Engineering Applications of Artificial Intelligence, 42, 45-56.

Drexl, M. (2012). Synchronization in vehicle routing—a survey of VRPs with multiple synchronization constraints. Transportation Science, 46(3), 297-316.

Dukic, G., Opetuk, T., & Lerher, T. (2015). A throughput model for a dual-tray Vertical Lift Module with a human order-picker. International journal of production economics, 170, 874-881.

Gendreau, M., Hertz, A., Laporte, G., & Stan, M. (1998). A generalized insertion heuristic for the traveling salesman problem with time windows. Operations Research, 46(3), 330-335.

Glover, F. (1989). Tabu search-part I. ORSA Journal on Computing, 1(3), 190-206.

Hansen, P., & Mladenovic, N. (2003). A tutorial on variable neighborhood search. Groupe d'études et de recherche en analyse des décisions, HEC Montréal.

Hansen, P., Mladenović, N., & Pérez, J. A. M. (2010). Variable neighborhood search: methods and applications. Annals of Operations Research, 175(1), 367-407.

Hoff, A., Andersson, H., Christiansen, M., Hasle, G., & Løkketangen, A. (2010). Industrial aspects and literature survey: Fleet composition and routing. Computers & Operations Research, 37(12), 2041-2061.

Ioannou, G., Kritikos, M., & Prastacos, G., (2001). A greedy look-ahead heuristic for the vehicle routing problem with time windows. Journal of the Operational Research Society, 52(5), 523-537.

Irnich, S., Schneider, M., & Vigo, D. (2014). Four variants of the vehicle routing problem. Vehicle Routing: Problems, Methods, and Applications, 18, 241-271.

Jiang, J., Ng, K. M., Poh, K. L., & Teo, K. M. (2014). Vehicle routing problem with a heterogeneous fleet and time windows. Expert Systems with Applications, 41(8), 3748-3760.

Kindervater, G.A.P.,& Savelsbergh, M.W.P., (1998). Vehicle routing: handling edge exchanges. In: Aarts, E.,Lenstra, J.K. (eds.) Local Search in Combinatorial Optimization. pp. 337–360. Wiley, UK.

Koç, Ç., Bektaş, T., Jabali, O., & Laporte, G. (2015). A hybrid evolutionary algorithm for heterogeneous fleet vehicle routing problems with time windows. Computers & Operations Research, 64, 11-27.

Koç, Ç., Bektaş, T., Jabali, O., & Laporte, G. (2016). Thirty years of heterogeneous vehicle routing. European Journal of Operational Research, 249(1), 1-21.

Kritikos, M. N., & Ioannou, G. (2013). The heterogeneous fleet vehicle routing problem with overloads and time windows. International Journal of Production Economics, 144(1), 68-75.

Lau, H. C., Sim, M., & Teo, K. M. (2003). Vehicle routing problem with time windows and a limited number of vehicles. European Journal of Operational Research, 148, 559-569.

Lerher, T., Ekren, B. Y., Sari, Z., & Rosi, B. (2015). Simulation analysis of shuttle based storage and retrieval systems. International Journal of Simulation Modelling, 14(1), 48-59.

Lerher, T. (2016). Travel time model for double-deep shuttle-based storage and retrieval systems. International Journal of Production Research, 54(9), 2519-2540.

Lerher, T., Borovinsek, M., Ficko, M., & Palcic, I. (2017). Parametric study of throughput performance in SBS/RS based on simulation. International Journal of Simulation Modelling, 16(1), 96-107.

Lim, A., & Wang, F. (2004). A smoothed dynamic tabu search embedded GRASP for m-VRPTW. In Tools with Artificial Intelligence, 2004. ICTAI 2004. 16th IEEE International Conference on (pp. 704-708). IEEE.

Lim, A., & Zhang, X. (2007). A two-stage heuristic with ejection pools and generalized ejection chains for the vehicle routing problem with time windows. INFORMS Journal on Computing, 19(3), 443-457.

Lin, S. (1965). Computer solutions of the traveling salesman problem. Bell System Technical Journal, The, 44(10), 2245-2269.

Liu F.H. & Shen S.Y. (1999). The fleet size and mix vehicle routing problem with time windows. Journal of the Operational Research Society, 50(7),721-732, 1999.

Mladenović, N., & Hansen, P. (1997). Variable neighborhood search. Computers & Operations Research, 24(11), 1097-1100.

Molina, J. C., Eguia, I., & Racero, J. (2018). An optimization approach for designing routes in metrological control services: a case study. Flexible Services and Manufacturing Journal, 30(4), 924-952.

Molina, J. C., Eguia, I., & Racero, J. (2019). Reducing pollutant emissions in a waste collection vehicle routing problem using a variable neighborhood tabu search algorithm: a case study. TOP, 27(2), 253-287.

Paraskevopoulos, D. C., Repoussis, P. P., Tarantilis, C. D., Ioannou, G., & Prastacos, G. P. (2008). A reactive variable neighborhood tabu search for the heterogeneous fleet vehicle routing problem with time windows. Journal of Heuristics, 14(5), 425-455.

Rosi, B., Grasic, L., Dukic, G., Opetuk, T., & Lerher, T. (2016). Simulation-based performance analysis of automated single-tray vertical lift module. International Journal of Simulation Modelling, 15(1), 97-108.

Savelsbergh, M. W., (1992). The vehicle routing problem with time windows: Minimizing route duration. ORSA Journal on Computing, 4(2), 146-154.

Schrimpf, G., Schneider, J., Stamm-Wilbrandt, H., & Dueck, G. (2000). Record breaking optimization results using the ruin and recreate principle. Journal of Computational Physics, 159(2), 139-171.

Solomon MM (1987) Algorithms for the vehicle routing and scheduling problems with time window constraints. Operational Research 35(2), 254-265.

Steuer, R. E. (1986). Multiple criteria optimization: theory, computation, and applications. Wiley.

Taillard, É., Badeau, P., Gendreau, M., Guertin, F., & Potvin, J. Y., (1997). A tabu search heuristic for the vehicle routing problem with soft time windows. Transportation Science, 31(2), 170-186.

Vidal, T., Crainic, T. G., Gendreau, M., Lahrichi, N., & Rei, W. (2012). A hybrid genetic algorithm for multidepot and periodic vehicle routing problems. Operations Research, 60(3), 611-624.

Vidal, T., Crainic, T. G., Gendreau, M., & Prins, C. (2014). A unified solution framework for multi-attribute vehicle routing problems. European Journal of Operational Research, 234(3), 658-673.

Wang, X. P., Xu, C. L., & Hu, X. P. (2008). Genetic algorithm for vehicle routing problem with time windows and a limited number of vehicles. In Management Science and Engineering. ICMSE 2008. 15th Annual Conference Proceedings., International Conference on (pp. 128-133). IEEE.

Wei, L., Zhang, Z., & Lim, A. (2014). An adaptive variable neighborhood search for a heterogeneous fleet vehicle routing problem with three-dimensional loading constraints. IEEE Computational Intelligence Magazine, 9(4), 18-30.

Yepes, V., & Medina, J. R. (2002). Optimización de rutas con flotas heterogéneas y múltiples usos de vehículos VRPHEMSTW. In Actas del V Congreso de Métodos Numéricos en Ingeniería. Madrid (pp. 3-6).