To cite this article:

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*, 924-952.

An optimization approach for designing routes in metrological control services: A case study

Jose Carlos Molina; Ignacio Eguia; Jesus Racero

Abstract This paper is the first to tackle the problem of designing routes in service companies that are responsible for the metrological control of measuring equipments at customer sites. This real-world problem belongs to the well-known Rich Vehicle Routing Problems which combine multiple attributes that distinguish them from traditional vehicle routing problems. The attributes include 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. This routing and scheduling problem is modeled with linear programming techniques and solved by a variable neighborhood descent metaheuristic based on a tabu search algorithm with a holding list. A real-life case study faced by a company in the region of Andalusia (Spain) is also presented in this work. The performance of the metaheuristic is compared with the literature for the standard fixed heterogeneous vehicle routing problem. Results obtained on a real case instance improve the solutions implemented by the company.

Keywords Rich VRP, VRP with time windows, Fixed Heterogeneous Fleet, Variable Neighborhood Descent

1 Introduction

The efficient and effective movement of goods or drivers to provide services is one of the most important logistical activities in today´s competitive companies since it can increase efficiency and productivity in many different ways. An effective management has a significant impact on both service quality and product cost reduction, achieving the company´s differentiation in a competitive market. Transportation plays a central role in distribution and services and has an influence in customer satisfaction levels.

Among the logistical activities, the vehicle routing problem (VRP) is one of the most widely researched. The classical VRP tries to minimize the total distance travelled by a set of vehicles while satisfying the demand of a given set of customers and considering the assumption that each vehicle serves a single route during any planning period. Although there are several variants and specializations of the VRP, the problem often does not include all the constraints and requirements of real companies. These requirements have received relatively little attention in the literature for years. Nevertheless, the new needs of companies have forced researchers to consider more complex and efficient approaches and this is how the Rich Vehicle Routing Problems (RVRPs) emerged.

RVRPs are VRPs from a real-life scenario, that deal with additional constraints which aim to take into account the particularities of the vehicle routing distribution system more precisely. RVRPs combine multiple attributes that might include dynamism, stochasticity, heterogeneity, multi-periodicity, integration with other related activities, diversity of users and policies, legal and contractual issues, environmental issues, and more (Caceres et al., 2015). These attributes complement the traditional VRP formulations and lead to a variety of Multi-Attribute Vehicle Routing Problems (MAVRPs).

The aim of this work is to tackle a real-world RVRP that matches with the current metrological control practice in companies and to propose solution methods that can solve real-life instances. Metrological control services cover all administrative and technical activities which are conducted to ensure that measuring instruments, apparatus and equipment perform their tasks properly and in accordance with the industry regulations and requirements. Metrological control can be performed at local laboratories, such as Motor Ordinance Test (MOT) checks, or at customer sites, such as fuel dispenser checks, scale tests, etc.

The problem of designing routes in real service companies that are responsible for the metrological control of measuring equipments basically consist of the following. Everyday orders to be served at customers sites are received by the company. The distribution business of these companies is affected by high seasonality throughout the year. For this reason, the number of orders received in a week can exceed the companies′ capabilities. In order to implement these activities, companies have a set of own workers, called verifiers, who visit the customers to verify and calibrate their measuring equipments using a fixed fleet of vehicles located in a central laboratory and a set of available measure instruments (patterns) which depends on the type of service to be performed. The weekly planning consists on selecting the customer orders to be performed and designing the routes of customers to be visited by each verifier every day considering constraints on time windows, working regulations, verifiers' skills, customer priorities, vehicle-verifier-order constraints and limited number of available patterns and vehicles.

This metrological control services problem has been presented to the authors by a public company in Andalusia, Spain, which aims to perform the legal metrological control in this region. It is important to note that the relevance of this sector continues increasing in Andalusia. The net turnover in 2014 reached the amount of 108.42 million euros, representing an increase over the previous year of 3.58%.

This paper is organized as follows: Section 2 reviews the literature about related problems. Section 3 describes the details of the tackled problem. The solution methodology including the mathematical formulation and algorithmic details to solve the problem are explained in Section 4. The experimental results of the case study are given in section 5 and finally, the conclusions are presented in the last section.

2 Literature review

In MAVRPs, Vidal et al. (2013) distinguish three main classes of attributes, relative to their impact on different aspects of the problem solution. This classification is related to the methodology for solving the problem. These classes are the assignment of customers and routes to resources, the sequence choices, and the evaluation of fixed sequences. In Lahyani et al. (2015) RVRPs are considered according to the Scenario Characteristics (SCs) and to the Problem Physical Characteristics (PPCs). Under each of these two classes, attributes are listed in an arborescent way with the company decision levels (strategic, tactical and operational). The first and second levels (strategic and tactical) include decisions related to SCs, while the operational level is associated with the PPCs. The strategic level includes decisions related to locations, the number of depots used, and the data type. The tactical level defines the order type and the visit frequencies of customers over a given time horizon. Finally, the operational level considers vehicle and driver schedules and attributes are specified for customers, vehicles, drivers, and roads.

From the point of view of the SCs the problem is defined as a multi-period vehicle routing and driver scheduling problem with a single depot, single trips and service delivery tasks where capacity constraints are not presented. The problem is deterministic and static since data are not random and it is assumed that no changes can affect the problem as car breakdowns, traffic jams, new order placements, unexpected withdrawal of orders, etc. On the other hand, the attributes related to the PPCs that are taken into consideration in this work are summarized in the following items:

- **Fixed Heterogeneous fleet**: The Heterogeneous VRP with a limited number of vehicles (HVRP) was proposed by Taillard (1999) and it consists of optimizing a set of routes for a fixed fleet of vehicles with different characteristics which serve some customers with known demands from a central depot. Every customer is visited exactly once by a vehicle and all vehicles start and finish their routes at a depot, without exceeding its capacity. A simplification of this problem is the multiple traveling salesman problem (m-TSP) where the capacity restrictions are removed. In general, the m-TSP is a generalization of the Travelling Salesman Problem (TSP) where tours are designed for *m* salespersons that start and end their trip at the depot. A survey on m-TSP, its variants, some practical applications and solving approaches can be found in Bektas (2006) and Yuan et al. (2013).
- **Time windows:** When hard time windows (TW) are considered, additional restrictions are introduced to force customers to be served by a vehicle in a predefined time window [EWi , LWi], where EWi and LWi are the earliest and latest time to start the service respectively. If a vehicle arrives before EWi, a waiting time must be taken into account until the beginning of the time window; if the vehicle arrives later than LWi, the solution is infeasible. Examples of solving HVRP with TW include Paraskevopoulos et al. (2008), Rieck & Zimmermann (2010) and Koç et al. (2015). The company under study establishes a route duration of eight hours for the verifiers. It is a parameter which must be greater or equal to the total driver working hours in one day. It is often defined as a time window at the depot.

- **Patterns restrictions:** In addition to the attributes analyzed in Vidal et al. (2013), in this problem routes must be designed accordingly to the daily limitation on the number of measure patterns in the central laboratory. The different routes designed in a specific day have to share these scarce resources. Therefore, the problem presents resource synchronization between tours and can accordingly be formulated as follows: "the total consumption of a specified resource by all vehicles must be less than or equal to a specified limit". Drexl (2012) presents an elaborated and recent survey on vehicle routing problems with multiple synchronization constraints.
- **Driver-order restrictions:** There are also driver–order limitations, which indicate that orders must (not) be served by exactly one verifier from a specific set of qualified verifiers. These constraints depend on the verifier qualification. An example of applying these restrictions can be found in Rieck & Zimmermann (2010). Moreover, Cappanera et al. (2011) defined the Skill VRP which originates from a real application in Field Service and is related to Home Care applications. The problem consists of defining a set of tours, each one operated by a technician, to fulfill service requirements asking for a particular skill. The authors formulated and tested three models with an increasing level of variables disaggregation obtaining very good linear programming bounds.
- **Vehicle-order restrictions:** In the problem under study, each order type must be served by a specified vehicle type. This is due to vehicle requirements and patterns incompatibility between different types of orders. An example of applying these restrictions can be found in Oppen & Løkketangen (2008). The site dependent VRP is a variant that takes into account these types of incompatibilities between customers and vehicles types (Baldacci et al., 2008).

- **Order priorities:** In addition to the previous attributes, the company assigns priority tags to some orders to determine which of them must be served first. Depending on these priorities, some orders can be postponed until the next planning horizon increasing their priority tag, thus, older orders have higher priorities. Some papers in the literature include customer priorities, for instance Cornillier et al. (2009) and De Armas et al. (2015).
- **Unserved orders:** The company under study often cannot serve all orders received in a week because it usually deals with a great number of orders exceeding its capabilities. It forces the company to postpone some of the orders to the next planning period. This results in the consideration of a generalized method to handle both feasible and infeasible problem instances. For handling both problem types, a holding list (HL) containing unserved customers is introduced in the developed algorithm. This idea was first introduced by Lau et al. (2003) and subsequently implemented by other authors as Jiang et al. (2014) or Lim and Zhang (2007).

Due to the combinatorial complexity of RVRPs, solving real-life sized problems by heuristics or metaheuristics are often preferred since they can find good solutions within a reasonable amount of computational time. A great number of heuristic and metaheuristic approaches are proposed to solve the HF-VRP in the scientific literature. For further information on HF-VRP a survey can be found in Baldacci et al., (2008), Irnich et al., (2014), and Koç et al. (2016). Lahyani et al. (2015) also presents information about the use of metaheuristics in RVRPs.

Mladenovic and Hansen (1997) proposed an optimization technique called Variable Neighborhood Search (VNS). It is a metaheuristic for solving optimization problems based on systematic changes of neighborhood during the search. A variant of the VNS arises when the search is performed in a deterministic way. The metaheuristic is known as Variable Neighborhood Descent (VND). On the contrary, if the search is randomly performed we have the reduced VNS. Interesting new variants of the VNS are presented in Hansen and Mladenovic (2003).

Applications of VNS, or hybrids of VNS combined with other metaheuristics, are diverse and numerous and have proved their effectiveness (Paraskevopoulos et al., 2008; Hansen et al., 2010; De Armas et al., 2015). For further information, Hansen et al. (2010) present an extensive review on VNS applications.

This paper makes two main scientific contributions. First, to the best of our knowledge, it does not exist in the scientific literature any practical VRP application which combines the attributes listed above. Another major difference considered in our implementation, compared to the large majority of papers on VRP literature is the differentiation between a vehicle and its driver. Customer orders are fulfilled by exactly one technician depending on their skill with a specified type of vehicle. Hence, routes for drivers and vehicles must be synchronized. Moreover, we want to draw attention to the importance of tackling resource synchronization constraints found in other real world VRP applications such as home care scheduling problems. The paper also introduces a mixed-integer linear programming model for a multi-period vehicle routing and driver scheduling problem that considers resource synchronization constraints. The model is based on an extension of a traditional HVRPTW model, incorporating some new decisions variables. The second contribution is a VND algorithm with a tabu structure for the local search which aims to efficiently solve this RVRP. The latter combines some new neighborhood structures definitions and a holding list that prevent the algorithms from being trapped at local optimal and to explore a larger search space.

3 Problem description

In this section the metrological control services problem is presented through a real-life company that is responsible to perform the legal metrological control in the region of Andalusia (Spain). Section 3.1 describes the problem scenario while section 3.2 presents the current methodology for designing routes in the company.

3.1 Problem scenario

The company performs metrological control services in eight provincial laboratories, which are regulated by a central reference laboratory located in Seville. In addition to the provincial laboratories, and in order to perform the services that must be provided at customer locations, the company owns 34 well-equipped vans and 6 trucks. The purpose of this paper is to elaborate a methodology to be used by the company for designing routes to give service to this type of customers.

The company receives metrological control job orders from the customers. These orders are internally classified into types of metrological services and stored in a database. Each type of service consists on a different set of operations which requires special patterns and involves a different service time at the customer location (see Table 1 and Table 2).

The company requires verifiers, who are suitably qualified people for performing each type of service. However, each verifier is only specialized in a set of types of services. Verifiers also need a set of patterns to complete each type of service. These patterns are physical elements that serve as a standard reference for measurements to be performed during the service. The patterns are loaded on vehicles but not all combinations of patterns are possible, as there are bulky patterns that cannot be carried together.

There is also a limited number of each type of pattern available at the laboratory. The quantity of the different patterns to perform the services does not depend on the laboratory and is shown in Table 2.

Provincial laboratories have their own fleet of vehicles that is used for handling the orders. The fleet consists on vans and trucks that vary in number depending on the province. There are two types of vans: standard vans and special vans, the latter with an installation of special pipes for gas analyzers and opacimeters. In addition, there are also trucks for verifying large tonnage scales. Some types of service must only be performed with one type of vehicle. Every day, each vehicle is assigned to a verifier and a set of patterns and a feasible route is designed. Other constraints to be considered are the limited working hours for each verifier which should not exceed the hours of the workday, i.e., 8 hours. In addition, there are orders that must be served in a predefined time window. This is the case of small businesses (with only one scale, one fuel dispenser, etc.), in which the service must be performed early in the workday, so that it doesn´t interfere with the opening hours. Otherwise, this would imply a financial loss for customers.

Table 1. Type of services and patterns

Type of service	Denomination	Group of service	Type of vehicle	Service times (min)	Pattern codes needed
T1	Checking fuel dispensers	Hydrocarbons (H)	Van	40	3,16,17,18,19,20,28
T2	Checking fuel pumps	Hydrocarbons (H)	Van	20	3,16,17,18,19,20,28
T3	Weighing up to 60 Kg	Mass (M)	Van	40	5,20
T4	Weighing up to 500 Kg	Mass (M)	Van	130	10,20
T5	Weighing more than 500 Kg	Mass (M)	Truck	130	9,11,20
T6	Checking tire pressure gauges	Pressure (P)	Van	30	2,14,20,25
T7	Calibration of Opacimeters	Gases (G)	Van	30	3,20,22,23,26
T8	Verification of Opacimeters	Gases (G)	Van	45	3,20,22,23,26
T9	Verification of volumetric meters for tanker trucks	Volume (V)	Van	260	3,13,20,21,24,27
T10	Verification of Temperature recorders	Temperature (T)	Van	60	1,3,20,26,28
T11	Calibration of Exhaust Gas Analyzers	Gases (G)	Special van	30	20

T12	Verification of Exhaust Gas Analyzers	Gases (G)	Special van	45	20

3.2 Problems and objectives

As it was already mentioned, every day the company receives the customer orders to be served as soon as possible, and places them into a dynamic list of unserved customers. To manage the large amount of orders, the company's logistic department assigns responsibility of carrying out the requested orders of its province to each provincial laboratory. In general, when the company receives a job order, it includes the following information:

- Company name.
- Address, telephone and email.
- Type and amount of services to be performed.
- Date of the order receipt.
- Time window for the service.

Table 2. Patterns description and available quantity

Pattern code	Denomination	Available quantity	Pattern code	Denomination	Available quantity
1	Thermal shock absorber	4	15	150 grams weight	1
2	Bottle of liquid nitrogen	6	16	10 liter glass flask	7
3	Chronometer	6	17	2 liter glass flask	7
4	Optical filter kit	6	18	20 liter glass flask	7
5	Truck weights kit	8	19	5 liter glass flask	7
6	Weights kit from 1 gr to 10 Kg	1	20	Environmental meter	14
7	Weights kit from 1 mg to 10 Kg	3	21	Petrol and diesel fuel volume meter	1
8	Weights kit from 1 mg to 500 mg	5	22	Smoke generator engine for opacimeters	1
9	Weights kit from 100 Kg to 200 Kg	1	23	Reference opacimeter	1
10	Weights kit from 100 mgr to 10 Kg	4	24	Wireless temperature sensor	4
11	20 Kg Weights kit	1	25	Reference tire pressure gauges	7

12	Weights kit for pallets	1	26	Portable PC and software	1
13	Metal volumetric flasks of 2,5,10 and 20 liters	1	27	Wireless temperature measuring system	2
14	Tire pressure gauge	1	28	Thermometer	9

Currently, due to the large number of orders and the limited number of its own technical verifiers and vehicles, very frequently some of the orders have to be postponed to the following planning horizon. Moreover, due to the complexity of the problem, the vehicle routes are manually designed by the heads of laboratories according to their experience and considering some simple rules, without applying any methodology for optimizing them. These planning rules are the following:

- Customer orders are classified by type of service and geographical area according to the ZIP code.
- Routes with nearby orders, with only one type of service, are designed. Thus, patterns incompatibilities and problems in the driver qualification are avoided. Moreover, the company takes into account the order priorities and time window restrictions in the design of the routes. In general, customers with small businesses must be served in the early hours of the route and the oldest orders get the highest priority.

This approach presents several disadvantages. The first problem is that routes are manually designed by the heads of laboratories according to their experience, losing an important amount of time in the planning process. Schedules and routes are planned one week before, and that task takes more than five hours. Moreover, the cost associated to inefficient planning must also be taken into account; this cost includes fuel costs, lost time for the verifiers, over-time working hours, unsatisfied customers, etc.

The manual design of vehicle routes is not the only problem the company has. The goal of the company is to be the best operating in terms of quality and it wants to achieve high operational efficiency. In order to provide the best quality service to customers and due to the large number of orders received in a day, two objective functions must be considered by the company in a hierarchical form: maximizing the number of prioritized served orders is in the top level (primary) and minimizing the total traveled distances is in the bottom (secondary). Therefore, it is important for the company to have a tool to help the heads of laboratories to make the right decisions concerning the daily vehicles assignment to the verifiers and the optimization of the routes.

Although exact algorithmic methodologies are not appropriate when solving real-life large VRP instances, a large amount of mathematical formulations, relaxations and recent exact methods are presented in the scientific literature for the VRP and its variants (Toth and Vigo, 2014).

To the best of our knowledge, there is no mathematical model dealing with a multi-objective vehicle routing and driver scheduling problem with a fixed heterogeneous fleet, multi-period, time windows, holding list and resource synchronization constraints in the scientific literature. Before getting down to the details of this formulation, let us suggest that this model should not be used to solve all RVRP instances. Nevertheless it is useful to describe and understand all constraints clearly and it can also be used as a basis for formulating more RVRPs applications with additional attributes.

The attributes of the RVRP presented in this paper are frequently faced by companies belonging to other application sectors as the home care providers industry or the industrial maintenance services providers. Therefore, this work contributes to solving RVRPs in other potential areas considering their specific attributes.

The problem under study is defined on a graph $G=\{\mathcal{N}, \mathcal{A}\}$ with $\mathcal{N}=\{0,1,\ldots,N\}$ as a set of nodes for a planning period, where node 0 represents the depot, and $\mathcal{A}$ is a set of arcs defined between each pair of nodes. A set of K heterogeneous vehicles and a fictitious vehicle denoted by K+1, is represented by $\mathcal{K}=\{1,\ldots K, K+1\}$ and is available from the depot. The fictitious vehicle is usually called “phantom vehicle” (Lau et al., 2003) and it will travel to those nodes that cannot be served in the established planning period. The set of verifiers that are to be assigned to vehicles is denoted by $\mathcal{S}$={1,…,S} and the planning horizon expressed in days is given by $\mathcal{T}$={1,…,T}.

The notation adopted is the following:

- PR_i : priority in node i.
- $[EW_i, LW_i]$: earliest and latest time to begin the service at node i.
- ST_i : service time in node i.
- TD_{ij} : distance from node i to node j ($i \neq j$).
- TT_{ij} : driving time between the nodes i and j.
- $TMax_s$: maximum allowable driving time for verifier s.
- NJ_s : maximum number of workdays for verifier s.
- SW_{is} : equal to 1 if verifier s can provide a service at node i, 0 otherwise.
- WJ_{st} : equal to 1 if verifier s can work on day t, 0 otherwise.
- SV_{ik} : equal to 1 if node i can be served with vehicle k, 0 otherwise.
- NP_p : number of patterns p available in a day.
- SP_{ip} : equal to 1 if node i needs the pattern p to be served, 0 otherwise.

The problem uses the following decision variables:

- X_{ijkt} : binary variable, equal to 1 if vehicle $k \in \{1,\dots,K+1\}$ travels from nodes i to j ($i \neq j$) on day t; $X_{ijK+1t}=1$ represents the list of unserved orders i.
- Y_{ikt} : starting service time at node $i \in \{0,1,\dots,N\}$ performed by a vehicle $k \in \{1,\dots,K\}$ on day t; y_{0kt} is the time of arrival at the laboratory.
- Z_{skt} : binary variable, equal to 1 if verifier $s \in \{1,\dots,S\}$ is assigned to vehicle $k \in \{1,\dots,K\}$ on day t.

- W_{pkt} : binary variable, equal to 1 if pattern p ∈ {1,…,P} is assigned to vehicle k ∈ {1,…,K} on day t.

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

$$\sum_{j=1}^{N} X_{0jkt} \leq 1 \quad \forall k \neq K+1, \forall t \tag{1}$$

$$\sum_{\substack{j=0 \\ j\neq i}}^{N} X_{jikt} - \sum_{\substack{j=0 \\ j\neq i}}^{N} X_{ijkt} = 0 \quad \forall i, \forall k \neq K+1, \forall t \tag{2}$$

$$\sum_{t=1}^{T}\sum_{k=1}^{K+1}\sum_{\substack{j=0 \\ j\neq i}}^{N} X_{ijkt} = 1 \quad \forall i \neq 0 \tag{3}$$

$$Y_{ikt} + ST_i + TT_{ij} \leq Y_{jkt} + (\sum_{s=1}^{S} TMax_s) \cdot (1 - X_{ijkt}) \quad \forall i \neq 0, \forall j, i \neq j, \forall k \neq K+1, \forall t \tag{4}$$

$$TT_{0j} \leq Y_{jkt} + (\sum_{s=1}^{S} TMax_s) \cdot (1 - X_{0jkt}) \quad \forall j \neq 0, \forall k \neq K+1, \forall t \tag{5}$$

$$EW_i \cdot \sum_{j=0}^{N} X_{ijkt} \leq Y_{ikt} \leq LW_i \cdot \sum_{j=0}^{N} X_{ijkt} \quad \forall i \neq 0, \forall k \neq K+1, \forall t \tag{6}$$

$$\sum_{j=1}^{N} X_{0jkt} = \sum_{s=1}^{S} Z_{skt} \quad \forall k \neq K+1, \forall t \tag{7}$$

$$Y_{0kt} \leq TMax_s + (\sum_{r=1}^{S} TMax_r) \cdot (1 - Z_{skt}) \quad \forall s, \forall k \neq K+1, \forall t \tag{8}$$

$$\sum_{t=1}^{T}\sum_{k=1}^{K} Z_{skt} \leq NJ_s \quad \forall s \tag{9}$$

$$\sum_{k=1}^{K} Z_{skt} \leq WJ_{st} \quad \forall s, \forall t \tag{10}$$

$$\sum_{t=1}^{T}\sum_{\substack{j=0 \\ j\neq i}}^{N} X_{jikt} \leq SV_{ik} \quad \forall i \neq 0, \forall k \neq K+1 \tag{11}$$

$$\sum_{\substack{j=0\\j\neq i}}^{N} X_{jikt} + 1 - SW_{is} \leq 1 + 1 - Z_{skt} \quad \forall i \neq 0, \forall k \neq K+1, \forall s, \forall t \tag{12}$$

$$\sum_{\substack{j=0\\j\neq i}}^{N} X_{jikt} + SP_{ip} \leq 1 + W_{pkt} \quad \forall i \neq 0, \forall k \neq K+1, \forall p, \forall t \tag{13}$$

$$\sum_{k=1}^{K} W_{pkt} \leq NP_{p} \quad \forall p, \forall t \tag{14}$$

$$\sum_{p=1}^{P} W_{pkt} \leq P \cdot \sum_{j=1}^{N} X_{0jkt} \quad \forall k \neq K+1, \forall t \tag{15}$$

$$\sum_{i=0}^{N} Y_{ikt} \leq N \cdot (\sum_{r=1}^{S} TMax_{r}) \cdot \sum_{j=1}^{N} X_{0jkt} \quad \forall k \neq K+1, \forall t \tag{16}$$

Constraints (1) mean that each vehicle departs from the laboratory each day once or it doesn't, that is, no more than K vehicles (fleet size) depart from the depot. Constraints (2) are the flow conservation on each node. Constraints (3) guarantee that each customer is visited exactly once in the planning period. It can be observed that unserved orders are visited by the phantom vehicle. Starting service times are calculated in constraints (4) and (5), where Y_{0kt} is the ending time of the tour for vehicle k on day t. These constraints also avoid sub-tours. Time windows are imposed by constraints (6). Constraints (7) mean that each vehicle departing from the laboratory is assigned a verifier. Any verifier cannot exceed its maximum allowable driving time in constraints (8). Constraints (9) are used to restrict the maximum number of workdays for a verifier. Constraints (10) guarantee that each verifier can be assigned to only one vehicle in the allowed workday. Incompatibilities between services and vehicles and services and verifiers are modeled in constraints (11) and (12) respectively. Constraints (13) ensure the use of patterns for every served order and constraints (14) limit the available number of each one in each workday. Constraints (15) and (16) avoid assigning patterns and calculating starting service times if a vehicle does not depart from the laboratory in a workday. Further-more, the objective of the route planning is a hierarchical objective function, where maximizing the total number of prioritized served orders is considered as primary objective, and minimizing the total traveled distance as secondary. Thus, if multiple solutions serving the same number of orders exist, the model must choose those with shorter distances. For this purpose δ is introduced in the objective function. This value represents a small positive number to ensure that the second term is of a lower order of magnitude than the first one

$$MAX \quad \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{j=1}^{N}\sum_{\substack{i=0 \\ i\neq j}}^{N} PR_j X_{ijkt} - \delta \cdot \sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{j=0}^{N}\sum_{\substack{i=0 \\ i\neq j}}^{N} TD_{ij} X_{ijkt} \tag{17}$$

4 Solution method

In the remainder of this section, the VND algorithm with tabu search and holding list for solving the tackled problem is presented. This tool will guide heads of laboratories on how to design the routes and schedules allowing the consideration of their preferences and specific constraints.

4.1 The Variable Neighborhood Descent Tabu Search Algorithm with a Holding List

In this section, the basic VND algorithm is adapted to solve the HVRPTW. The algorithm is improved by introducing tabu search for the local search and by adopting some local search procedures including several new neighborhood structures definitions and a holding list that prevents the algorithm from being trapped at local optimal and allows exploring a larger search space.

The algorithm starts with an initial solution. For this purpose, in a first phase a large number of initial solutions are produced using a semi-parallel construction heuristic for different combinations of parameter values, selecting only a small set F of high quality solutions for further improvement. Then, the post-optimization procedure is repeated for all the initial solutions of the subset F, re-starting from a new initial solution once a region has been extensively explored. The algorithm finishes when all solutions are examined or an upper bound limit γ, with respect to the computational time consumption, is reached. The use of a multi-start strategy is found to be useful in providing diversity and obtaining high quality solutions.

Compared to the solution approach for solving the HVRPTW described in Paraskevopoulos et al. (2008), there are several differences in our implementation. First, our approach handles both feasible and infeasible problem instances. Infeasible problems occur when some customers cannot be assigned to certain vehicles due to problem restrictions. In this situation, the basic idea is to introduce a holding list in the algorithm that contains all unserved customers. This idea was first introduced by Lau et al. (2003) and subsequently implemented by other authors, such as Jiang et al. (2014) or Lim and Zhang (2007). The problem introduces the additional objective of maximizing the number of served customers. One possible way is dealing with a multi-objective problem and defining weights for the different objective functions. However, our approach is to define a hierarchical cost structure where serving more customers is always better regardless of the objective function considered. Another major difference involves the deterministic scheme of the VND. Despite the shaking mechanism being used to allow a more efficient and effective intensification local search, random moves are avoided in our approach to future applications in real companies. Other differences are introduced in the heuristic metrics and in the definition of the neighborhood structures where the search space is increased to perform a more thorough search.

The semi-parallel construction heuristic is based on the insertion framework described in Paraskevopoulos et al. (2008) but introducing some differences. The heuristic builds a route for every feasible combination of vehicle, verifier and day taking into account incompatibilities between vehicle type and verifier qualifications and daily pattern limits. As capacity constraints are not considered and time windows restrictions are only to be met in a small set of orders, the new greedy function that evaluates the cost of inserting an order u between i and j served by vehicle of type k and a verifier s on the workday t is denoted in (18) where the α weights define the relative contribution of each individual metric to the overall selection.

$$\phi_{ij}^{ukst} = \alpha_1 \cdot C_{ij,u}^1 + \alpha_2 \cdot C_{ij,u}^2 + \alpha_3 \cdot C_u^3 \quad (18)$$

The first metric (19) is a measure of the driving time increase caused when an order u is inserted between two consecutive orders (i,j) in a route (Solomon, 1987). Similarly to the metric for measuring carried load increments (Paraskevopoulos et al. 2008), metric $C_{ij,u}^2$ (20) is introduced in this problem to give priority to the insertion of feasible orders with large services times on the route and maximize the use of the driver working time. Metric C_u^3 is introduced to give priority to the insertion of remote customers in the route construction. This component is a variant of the criteria proposed by Mole and Jameson (1976) and also used in Solomon (1987). It is expressed in equation (21), where 0 represents the depot or laboratory and G, the furthest unassigned order in terms of time.

$$C_{ij,u}^1 = TT_{iu} + TT_{uj} - TT_{ij} \quad (19)$$

$$C^2_{ij,u} = TMax_s - \sum_{i \in R} ST_i - \sum_{(i,j) \in R} TT_{ij} - ST_u \tag{20}$$

$$C^3_u = TT_{0G} - TT_{0u} \tag{21}$$

Finally, a new metric (β) is introduced for the routes selection considering the hierarchical cost structure of the problem, where maximizing the number of prioritized orders and minimizing the total travelled distances are the primary and secondary objectives respectively. To evaluate it, a small positive scalar ρ (Equation 22) is introduced to ensure that the term representing distances is of a lower order of magnitude than the term representing priorities. Thus, in case of existing multiple routes serving the same number of orders, the metric will choose those with shorter distances. Otherwise, the higher the value of β, the higher amount of prioritized orders is served. At the end of the procedure, if there are any unassigned orders left, an additional "phantom vehicle" is generated to serve them. The phantom vehicle contains the list of the orders that are not served in the current solution (Lau et al. 2003).

$$\beta_{\mathrm{k,s,t}} = \sum_{\mathrm{i \in R}} \mathrm{PR_i} \Big/ 1 + \rho \cdot \sum_{\mathrm{(i,j) \in R}} \mathrm{TD_{ij}} \qquad \rho = \sum_{i=1}^{N} PR_i \Big/ 10^3 \cdot \sum_{j=0}^{N} \sum_{\substack{i=0 \\ i \neq j}}^{N} TD_{ij} \tag{22}$$

In the second phase the solutions are attempted to be improved by a Variable Neighborhood Descent Tabu Search algorithm with a Holding List (VNDTS_HL). The algorithm starts by defining a set of neighborhood structures N_k ($k = 1 \ldots k_{max}$), where N_k is the k^{th} neighborhood. The iterative process starts from an initial solution s. Then, a local search based on tabu search is performed to determine a new solution *s´* in N_k. 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_{k+1}. This is repeated until all neighborhood structures are examined ($k = k_{max}$).

4.1.1 Tabu search

The tabu search (TS) is a metaheuristic that consists of moving successively, in each iteration from one solution *s* to the best or first improving solution of its neighborhood, prohibiting movements stored in the tabu list (TL) or, if so, satisfying some aspiration criterion (i.e. improving the best found solution in the problem).

The TL stores in a stated list the solutions explored throughout the search or, more commonly, some relevant attributes of these solutions. In this problem, the TL consists of storing the exchanged nodes and their initial positions before moving to other solution. The main purposes are to prevent the return to the most recent visited solutions in order to avoid cycling and to drive the search towards regions of the solution space not yet explored. The termination condition used is the maximum number of iterations without observing improvement.

<u>Variable Neighborhood Descent Tabu Search algorithm</u>

```
F ← Semi-parallel_Insertion_Heuristic;
Define a set of neighborhood structures N_k, k=1, 2,…,k_max;
For all solutions s of set F do:
   While (CPU time consumed ≤ γ) do:
      Set k ← 1;
      While (k ≤ k_max) do:
       s' ← TabuSearch (s, N_k);
       If f (s') improves f (s) then
          s ← s'; k ← 1;
       Else
```

```
            k ← k+1;
          EndIf
        EndWhile
        UpdateBestSolution(s);
      EndWhile
EndFor
```

4.1.2 Neighborhood structures

Five neighborhoods structures (k_{max}=5) are used in this algorithm to solve the RVRP under study. As opposed to other metaheuristics based on VNS, where the neighborhood structures are defined by a single local search operator, in our implementation some of them are defined by a set of two operators. Although the computational effort required to search the neighborhood becomes greater, the quality of the solution of the neighborhood may improve considerably. In this work, the following operators are used: Relocate (Savelsbergh, 1992), Exchange (Kindervater and Savelsbergh, 1997), 2-opt (Croes, 1958), 3-opt (Lin, 1965), CROSS-exchange (Taillard et al., 1997), double and triple insertion (Brandão, 2011). The operators are applied with the best accept strategy, i.e., the best improving move is first identified and then accepted. Moreover, the solution includes a holding list containing the list of the orders that are not served. The holding list 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 within Relocate, Exchange and CROSS-exchange operators. These movements are:

- Relocate from holding list: Transferring orders from HL to an existing route.
- Relocate to holding list: Transferring orders from an existing route to the HL.

- Exchange with holding list: Exchanging orders from an existing route with another group of orders in the HL.

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

The neighborhood structures are briefly described below and are performed in the following sequence:

- *Mixture*: This neighborhood structure is defined by Relocate and Exchange operators. It is applied only on pairs of routes (inter-route) and it aims to generate a feasible solution by removing an order from a route and insert it into another route or swapping a pair of orders from two different routes.
- *Crossings*: This neighborhood structure is also applied only on pairs of routes and is defined by the CROSS-exchange operator. CROSS-exchange swaps segments of orders between two routes. The different segments may contain an arbitrary number of orders. Due to the typically vast number of neighbors that would result, the segment length is limited to two orders in this problem. Thus, set of 2-2 and 1-2 swaps are defined.
- *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 and 3-OPT operators.

- *Interchanges*: This neighborhood structure is similar to the mixture type but 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 moves. In a double or triple insertion move, the operation is similar to the single insertion one except for removing respectively two or three consecutive orders belonging to the same route.

5 Computational Results

This section describes the computational experiments carried out to validate the effectiveness of the algorithms developed to solve the presented RVRP. The algorithms were developed in C++ and run on a 3.30 GHz Intel® Core(TM) i5-2400 CPU.

First, the parameters used within the algorithm are described. Secondly, a comparative analysis with the best results from the literature corresponding to the standard HVRPTW benchmark instances is performed. Next, real data corresponding to a weekly planning of the company under study is presented and solutions provided by the algorithms are compared with the solutions implemented by the company. Finally, the results obtained in a simplified planning problem by using CPLEX and the VNDTS_HL is presented.

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, we initially used the parameters suggested by Paraskevopoulos et al. (2008) for the VNDTS_HL.

The first phase of the proposed methodology consists on the calculation of the set *F* with different initial solutions obtained by varying the construction parameters of the heuristic α_1, α_2 and α_3, and selecting the best solutions. The main focus of the problem under study is on both improving the intra-route sequence of orders and the vehicle route travel time utilization. For these reasons, the values of α_1 and α_2 ranged between 0.3-0.6 via increments of 0.01. Experimentation suggests a size of 30 solutions for the set *F*. Other parameters that affect the VNDTS_HL are the maximum number of iterations for the TS and the TL size. Based on a set of previous experiments these values were set to 30 and 15 respectively. The algorithm is run once for every instance, with a maximum computer time of 1200 seconds, since the purpose of these experiments is to evaluate the proposed methodology as a powerful tool for fast and effective fleet scheduling in real-life situations. Table 3 summarizes the parameters used by the algorithm.

Table 3. Parameter settings

Heuristic parameters				Metaheuristic parameters			
α_1	α_2	α_3	$\Delta\alpha$	*F*	Nº Iterations	TL	γ (sec)
0.3-0.6	0.3-0.6	0-0.4	0.01	30	30	15	1200

5.2 Comparative with the literature

The HVRPTW benchmark data sets of Paraskevopoulos et al. (2008) is a subset of the FSMVRPTW instances, in which the fleet size is set equal to that obtained in the best known solutions of Liu and Shen (1999). In total there are 24 benchmark instances 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.

Note that the algorithm proposed in this paper is not designed to solve the HVRPTW as capacity constraints are not taken into account. In order to conduct the experiments, the semi-parallel construction heuristic proposed by Paraskevopoulos et al. (2008) is used to generate the set of initials solutions. Next, initial solutions are improved by a modification of the proposed VNDTS_HL algorithm which only considers capacity and time windows constraints.

Table 4 summarizes the average results obtained by VNDTS_HL compared to the current state-of-the-art solution approaches for the HVRPTW. 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 first column of the table shows the Paraskevopoulos instance category. Then, the total costs (TC) and the percentage deviation (% Dev) of the costs of each method with respect to VNDTS_HL are described. The last rows indicate the average percentage deviation over all problem instances.

Table 4. Comparison between different approaches for HVRPTW instances

Instance set	LS		ReVNTS		HEA		VNDTS_HL
	TC	% Dev.	TC	% Dev.	TC	% Dev.	TC
R1A (4)	4825.25	-11.45	4316.31	0.31	4298.68	0.72	4329.68
C1A (4)	8280.50	-9.91	7553.89	-0.27	7547.09	-0.18	7533.79
RC1A (4)	5486.00	-7.90	5114.13	-0.58	5093.54	-0.18	5084.46
R2A (4)	4248.25	-20.33	3521.06	0.27	3500.53	0.85	3530.62
C2A (4)	7160.25	-7.07	6729.50	-0.63	6687.11	0.00	6687.19
RC2A (4)	5647.25	-16.09	4854.09	0.14	4849.64	0.27	4864.18
Average		-12.13		-0.15		0.24	

The VNDTS_HL algorithm provides high quality solutions with average cost reductions from -12.16% to 0.22% and a worst case performance of 0.85%. For C1, C2 and

RC1 instances, the average results show substantially reduction of the total costs indicating the effectiveness of the proposed VNDTS_HL solution approach in this type of problems. It is important to note that the problem tackled in this paper usually presents a semi-clustered distribution of orders and a short scheduling horizon that favor the applicability of the proposed VNDTS_HL approach. Note that this paper is not aimed at overcoming the best results for the HVRPTWs, but rather at proposing an intelligent algorithm that works effectively when instances with real-world constraints are solved.

5.3 Real instance provided by the company

To conclude the computational experiments carried out in this paper, the solutions provided by the heuristic and the VNDTS_HL algorithm are compared to the solution implemented by the company for the real data. Additionally a set of three experiments has been carried out to show the different solutions obtained by the algorithm when different restrictions are taken into consideration.

The data belongs to the provincial Laboratory of Seville for the first week of September 2014. The choice of this period is related to the fact that is a representative week of the year with respect to the received orders. The fifty orders received from the different customers and their characteristics are shown in Table 5 and the geographical location of service orders and the provincial laboratory can be observed in Figure 2. All orders are assigned the same priority and as observed in Table 5, there are some of them to be served in a predefined time window in the early hours of the workday (those belonging to small business as already mentioned). The costs of travelling between every two orders (distances and travel times) have been obtained using the application Open Street Maps.

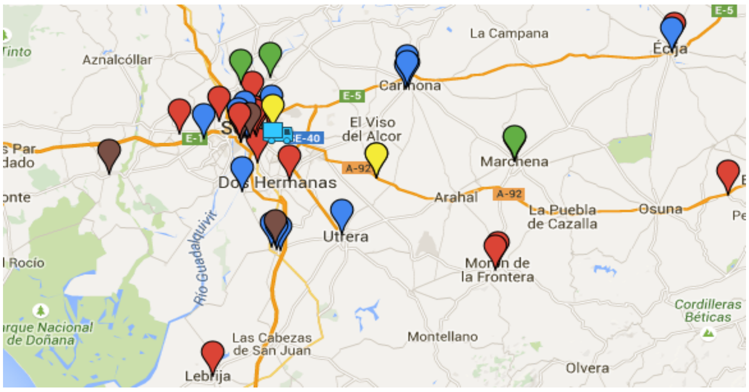

Figure.-2 Geographical location of service orders and provincial laboratory

The planning period is set to four days, from Monday to Thursday. On Fridays, verifiers must carry out administrative tasks at the central laboratory. The provincial laboratory of Seville currently has five verifiers responsible for the metrological services at customer locations, thus a maximum of twenty routes are expected to be planned. Those days with a lower number of worked hours, will be completed with administrative tasks. Table 6 shows the verifiers available during the planning period and their service type qualifications.

Table 5. Service types, total service times and time windows for the received order nodes.

Order Nº	Services types (quantity)	Group	Total S.T.(min)	TW (min)	Order Nº	Services types (quantity)	Group	Total S.T.(min)	TW (min)
1	T1(2)+T2(16)	H	400		20	T1(2)+T2(18)	H	440	
2	T1(2)+T2(6)	H	200		21	T1(1)	H	40	
3	T1(2)+T2(18)	H	440		22	T3(10)	M	400	
4	T1(2)+T2(8)	H	240		23-24	T3(2)	M	80	
5	T1(4)+T2(7)	H	300		25-27	T3(1)	M	40	[0-60]
6	T1(4)+T2(5)	H	260		28	T6(1)	P	30	
7	T1(1)+T2(8)	H	200	[0-60]	29	T6(2)	P	60	
8	T1(1)+T2(1)	H	60	[0-60]	30	T6(1)	P	30	
9	T1(3)+T2(5)	H	220		31-34	T9(1)	V	260	
10	T1(3)+T2(14)	H	400		35-37	T12(1)	G	45	
11	T1(2)+T2(3)	H	140		38	T1(3)+T2(6)	H	240	
12	T1(1)+T2(2)	H	80	[0-60]	39	T3(4)	M	160	
13	T1(3)+T2(10)	H	320		40	T3(3)	M	120	
14	T1(3)+T2(14)	H	400		41-45	T3(1)	M	40	
15	T1(1)+T2(7)	H	180	[0-60]	46	T3(3)	M	120	

16	T1(2)+T2(6)	H	200	47	T3(1)	M	40
17	T1(2)+T2(9)	H	260	48-49	T6(1)	P	30
18	T1(3)+T2(6)	H	240	50	T9(1)	V	260
19	T1(2)+T2(7)	H	220				

Table 6. Verifiers and service type qualifications

Verifier	Service type qualifications
1	T1-T2-T9-T10
2	T1-T2-T3-T4-T5-T9-T10
3	T1-T2-T3-T4-T5-T10
4	T1-T2-T3-T4-T5-T10
5	T1-T2-T6-T7-T8-T10-T11-T12

The provincial laboratory of Seville has an own fleet of vehicles consisting of five standard vans (nº 1 to 5), one special van with special pipes for gas analyzers and opacimeters (nº 6), and one truck for large tonnage scales (nº 7). The relation between type of service and vehicle type is shown in Table 1. The quantity of the different patterns available to perform the services is presented in Table 2.

Table 7. Head of laboratory solution

Day	Route	Group	Vehicle	Verifier	Time (min)	Dist. (Km)	Required Patterns
1	0-31-0	V	1	1	265.78	4.17	3-13-20-21-24-27
	0-10-0	H	2	2	413.72	11.53	3-16-17-18-19-20-28
	0-23-39-43-47-42-0	M	3	3	479.92	176.00	5-20
	0-7-21-0	H	4	4	334.28	130.48	3-16-17-18-19-20-28
	0-30-49-29-28-48-0	P	5	5	326.77	176.65	2-14-20-25
2	0-32-0	V	1	1	265.78	4.17	3-13-20-21-24-27
	0-20-0	H	2	2	459.23	17.79	3-16-17-18-19-20-28
	0-27-26-24-46-0	M	3	3	361.99	102.28	5-20
	0-15-11-0	H	4	4	467.14	220.67	3-16-17-18-19-20-28
	0-36-35-37-0	G	6	5	255.76	154.04	20
3	0-3-0	H	1	1	458.68	17.62	3-16-17-18-19-20-28
	0-33-0	V	2	2	265.78	4.17	3-13-20-21-24-27
	0-19-9-0	H	3	3	473.63	36.09	3-16-17-18-19-20-28
	0-25-41-45-44-40-0	M	4	4	354.70	83.90	5-20
	0-18-16-0	H	5	5	456.31	13.35	3-16-17-18-19-20-28
4	0-34-0	V	1	1	265.78	4.17	3-13-20-21-24-27
	0-22-0	M	2	2	419.33	14.92	5-20
	0-12-17-0	H	3	3	441.53	140.32	3-16-17-18-19-20-28

	0-8-5-0	H	4	4	458.31	130.28	3-16-17-18-19-20-28
	0-4-2-0	H	5	5	459.46	18.24	3-16-17-18-19-20-28
-	UNROUTED ORDERS				1-6-13-14-38-50		

The solution produced by the head of laboratory for the real planning problem (*a*) is shown in Table 7. As observed, each route performs only one type of service to avoid patterns incompatibilities and problems in the verifier qualification. Table 8 presents the routes of the weekly planning obtained by the VNDTS_HL algorithm. Note that the limitation to a single unit in the pattern number 21 causes that only one route belonging to the Volume group can be performed per day.

Table 8. Detailed solutions for the algorithm

Day	Route	Group	Veh.	Verif.	Time (min)	Dist. (Km)	Required Patterns
	0-22-0	M	1	3	419.33	14.92	5-20
	0-31-16-0	H+V	2	1	474.10	11.91	3-13-16-17-18-19-20-21-24-27-28
1	0-20-0	H	3	2	459.23	17.79	3-16-17-18-19-20-28
	0-7-40-0	H+M	4	4	415.92	123.55	3-5-16-17-18-19-20-28
	0-36-35-37-0	G	6	5	255.76	154.04	20
	0-6-48-28-29-0	H+P	1	5	464.47	87.43	2-3-14-16-17-18-19-20-25-28
	0-3-0	H	2	1	458.68	17.62	3-16-17-18-19-20-28
2	0-42-46-39-0	M	3	3	433.53	172.97	5-20
	0-27-8-24-33-0	H+M+V	4	2	465.50	22.48	3-5-13-16-17-18-19-20-21-24-27-28
	0-14-0	H	5	4	420.35	19.00	3-16-17-18-19-20-28
	0-49-17-0	H+P	1	5	390.85	136.79	2-3-14-16-17-18-19-20-25-28
	0-21-4-11-0	H	2	1	454.02	32.40	3-16-17-18-19-20-28
3	0-25-44-45-41-18-0	H+M	3	3	468.29	79.09	3-5-16-17-18-19-20-28
	0-26-1-0	H+M	4	4	478.36	40.23	3-5-16-17-18-19-20-28
	0-12-23-34-0	H+M+V	5	2	433.79	10.84	3-5-13-16-17-18-19-20-21-24-27-28
	0-30-13-0	H+P	1	5	440.90	120.00	2-3-14-16-17-18-19-20-25-28
	0-32-2-0	H+V	2	1	476.25	15.20	3-13-16-17-18-19-20-21-24-27-28
4	0-5-47-43-0	H+M	3	2	475.76	119.12	3-5-16-17-18-19-20-28
	0-19-9-0	H	4	3	473.63	36.09	3-16-17-18-19-20-28
	0-10-0	H	5	4	413.71	11.53	3-16-17-18-19-20-28
-	UNROUTED ORDERS				15-38-50		

In order to analyze the capabilities of the algorithm, we have also created three additional cases that may appear in this type of company. Case study (*b*) analyzes the situation when order priorities are changed; particularly the unserved orders in the real planning solution provided by the algorithm (orders nº 15, 38 and 50) are assigned a double priority. Problem (*c*) represents the problem when the number of vans is reduced in one unit. This is the case when company's vehicles need regular preventive maintenances or repairs. Similarly, problem (*d*) represents the problem when one specific verifier (nº3) is missing. The objective is to compare the results obtained by the head of laboratory´s current procedure to those obtained from the algorithms described in this paper, assessing the increments of served orders and the savings obtained in terms of distance. The performance of the head of laboratory method and the proposed methodology for the different cases are compared in Table 9.

Table 9. Obtained solutions

Case study	Head of laboratory			Heuristic			Metaheuristic			
	Served Orders	Dist. (Km)	Routes	Served Orders	Dist. (Km)	Routes	Served Orders	Dist. (Km)	Routes	CPU Time (min)
REAL PLANNING	44	1460.83	20	47	1689.15	20	47	1242.99	20	17.40
WITH ORDER PRIORITIES	44	1664.32	20	46	1704.79	20	47	1328.75	20	17.53
1 VEHICLE LESS	41	1410.50	17	43	1772.55	17	43	999.67	17	20.00
1 VERIFIER LESS	40	1398.97	16	42	1755.80	16	42	931.21	16	9.81

Looking at the results obtained for the real planning problem (*a*), the number of served orders can be even increased just by applying the semi parallel construction heuristic. The heuristic outperforms the current method with an increment of three served orders, around 6.8%, but this fact increases the total traveled distance by 15.63%. These results can be explained by the interdependency of the first and secondary objectives. It is usual to obtain a solution with an increased traveled distance when serving more orders. Nevertheless, by also allowing the use of the VNDTS_HL post-optimization algorithm the results show the same number of served orders as the heuristic and a 14.91% savings in distance compared to the current practice.

The solution obtained by the VNDTS_HL algorithm for problem (*b*), incorporates the orders with double priority with an increment of 85.76 kilometers (6.9%) with respect to the solution obtained for the real case. On the other hand, the results obtained for problems (*c*) and (*d*) are close to the solution obtained for problem (*a*) if routes with fewer served orders are removed.

In terms of total CPU time consumption (Table 9), during the first phase the heuristic consumed on average 50–60 seconds while using the VNDTS_HL took less than 23 minutes in the worst case. This value is considered as a short computation time in a weekly planning.

In summary, the results of the proposed methodology show a better planning distribution with substantial increments in the total number of served orders and also significant decreases in the total distance traveled. Routes with more than one type of order allow increasing the number of prioritized served orders with a significant reduction on the total distance traveled.

5.4 Comparison with CPLEX results

In this section, a comparison of the results obtained by using CPLEX and the VNDTS_HL is presented. For comparison purposes and in order to obtain any optimal solution by CPLEX, the real-life problem presented in Section 5.3 is simplified and reduced. In particular, the available fleet is reduced to vans nº 1, 2 and 6, the latter with special pipes for gas analyzers and opacimeters. Consequently, the set of available verifiers is also reduced to two people (verifier nº 2 and 5). A set of 8 instances of different node sizes are generated. Instances with 12, 14, 16, and 18 orders have been solved in a planning period of 2 days while the planning horizon in instances with 20, 30, 40 and 50 orders is increased to 4 days. Table 10 shows the orders belonging to each instance and the number of required variables and constraints. Stopping rules consider a limit time of 7200 seconds for CPLEX while for the VNDTS_HL the same parameters values are kept.

Table 10. Experimental instances

Nº Orders	12	14	16	18	20	30	40	50
Planning period	2 days	2 days	2 days	2 days	4 days	4 days	4 days	4 days
Order numbers	11-12; 26-32; 35-37	9-12; 26-32; 35-37	8-12; 25-32; 35-37	8-12; 24-33; 35-37	1-2; 8-12; 24-33; 35-37	1-17; 24-33; 35-37	1-22; 24-33; 35-42	1-50
Nº Var.	1696	2160	2688	3280	7872	16352	28032	42912
Nº Constr.	3464	4192	4968	5792	13246	23366	35886	50806

For every instance, the results obtained by CPLEX and the VNDTS_HL are compared, and a difference with respect to the solution reported by CPLEX (which for some instances correspond to the optimal solution) is estimated. As the objective of the route planning is a hierarchical objective function composed of two criteria, two different deviations are obtained as described in (23), where Z_f correspond to the value obtained by the VNDTS_HL for the objective function *f.* The deviation of the secondary objective function (minimizing distances) is only calculated when the deviation for the primary objective function (maximizing the number of prioritized served orders) is equal to zero. Thus, positive gaps are obtained when CPLEX finds better solutions.

$$Dev_1 = \frac{CPLEX_1 - Z_1}{CPLEX_1} * 100 \quad ; \quad Dev_2 = \frac{Z_2 - CPLEX_2}{CPLEX_2} * 100 \qquad (23)$$

Table 11. Obtained solutions

	CPLEX	Heuristic	Metaheuristic

Nº Orders	Planning period	S.O.D.(Km)		GAP(%)	T. (sec)	S.O.D.(Km)		T. (sec)	S.O.D.(Km)		CPU Time (sec)	%Dev_1	%Dev_2
12	2 days	12	293.70	0.00	5.94	12	293.70	0.23	12	293.70	0.31	0.00	0.00
14		12	253.22	0.00	5244	11	301.21	0.36	12	253.22	0.98	0.00	0.00
16		13	271.33	7.78	7200	11	311.66	0.38	13	271.33	1.33	0.00	0.00
18		14	285.50	21.58	7200	12	319.15	0.48	14	285.50	1.91	0.00	0.00
20	4 days	20	369.72	0.29	7200	20	393.95	1.26	20	376.45	6.00	0.00	1.82
30		15	674.90	103.07	7200	20	988.12	3.01	21	369.74	43.80	**-40.00**	-
40		14	575.48	191.83	7200	21	1263.37	7.29	26	535.71	125.00	**-85.71**	-
50		10	320.81	407.52	7200	24	1267.15	11.55	31	565.59	176.24	**-210.00**	-

Table 11 shows a summary of all results obtained. The first line of the table lists the methods used to solve the instances. The first and second columns of the table indicate the number of orders of each instance and the planning period expressed in days respectively. Then, the number of served orders (S.O.), the total distance traveled (D.) and the computational time (T.) of the solution provided by each method are shown. The optimality gap tolerance (GAP) found by CPLEX in the specified computational time is also indicated. Finally, an analysis of the quality of the VNDTS_HL in terms of the percentage deviations (% Dev) obtained with respect to CPLEX according to (23), are presented in the last two columns.

As observed in Table 11, the GAPs found by CPLEX in the specified computational time increase as long as the number of orders of the instance also increase in a particular planning horizon. Therefore, as the number of nodes increases, it becomes more difficult to obtain optimal solutions by CPLEX, which have been only found for the two smallest size instances (12 and 14 orders).

The solutions obtained by the VNDTS_HL for the smallest size instances (with a planning period of two days) match with those found by CPLEX. Moreover, the algorithm is more effective in terms of computational effort. Due to the quality of the solutions do not decrease when the size of the instances increased, the VNDTS_HL may be expected a good performance for bigger size instances in which solutions provided by

CPLEX cannot be compared. This fact can be observed in instances with 30, 40 and 50 orders where the VNDTS_HL yields important percentage deviations in the number of served orders with respect to CPLEX.

6 Conclusions

This work deals with a Rich Vehicle Routing Problem (RVRP) to solve the problem faced by a real service company that is responsible of the metrological control of measuring equipments in the region of Andalusia. In this problem, a weekly routing plan and driver scheduling is made for a set of verifiers and a fleet of heterogeneous vehicles to serve customers that need to be visited for the verification and calibration of a set of measuring equipment. Moreover, the company has different types of orders that have to be assigned to the verifiers, depending on their skills and need a set of measure patterns to carry out the service. In addition, a large amount of received service orders per week is considered allowing postponing some of them to the next planning. The attributes considered in this RVRP include a fixed heterogeneous fleet of vehicles, time windows for customers and depot, resource synchronization between tours, driver-order and vehicle-order constraints, order priorities and unserved customers.

The routing and scheduling problem is modelled with linear programming techniques. Compared with classic models from the literature for solving the HVRPTW, additional constraints and binary variables are added to tackle the RVRP under study. In order to obtain high quality solutions, a semi-parallel insertion heuristic and a VNDTS_HL algorithm are designed. These two methods are applied to design routes maximizing the number of served orders (main objective) while minimizing the total distance traveled (second objective).

Experimental results are presented with a comparative analysis with the best results from the literature corresponding to the standard HVRPTW benchmark instances. The proposed method provided high quality solutions with average cost reductions, especially in cluster and semi-cluster instances. In addition, based on a real instance provided by the company, the performance of the current planning method and the proposed methodology were compared. It was observed that both procedures outperformed the company implementation. The VNDTS_HL showed promising increments in the total number of served orders. More specifically, the VNDTS_HL algorithm applied to the real planning increments the number of served orders, around 6.8%, compared to the current method used by the head of the laboratory. A 14.91% reduction of the total distance travelled was also observed using the same available resources. The VNDTS_HL algorithm is also validated in three additional cases that may appear in this type of company.

Finally, a set of instances is generated to compare the results obtained by using CPLEX and the VNDTS_HL. Numerical results show that the VNDTS_HL presents reasonable gaps with respect to the solution found by CPLEX in smaller size instances becoming an alternative method for solving large instances that CPLEX cannot solve efficiently.

To conclude, the computational experience performed in this work validates the effectiveness of the proposed approach. A new tool has been developed to achieve a more efficient management of the resources in the company under study. This methodology will assist head of laboratories in their decisions, identifying the alternative with the highest number of prioritized orders and also reducing distances, costs and the amount of time in the planning process.

Acknowledgments

This research has been fully funded by the Andalusia Government through grants P10-TEP-6332.

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.

Bektas, T. (2006). The multiple traveling salesman problem: an overview of formulations and solution procedures. Omega, 34(3), 209-219.

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

Caceres-Cruz, J., Arias, P., Guimarans, D., Riera, D., & Juan, A. A. (2015). Rich Vehicle Routing Problem: Survey. ACM Computing Surveys (CSUR), 47(2), 32.

Cappanera, P., Gouveia, L., & Scutellà, M. G. (2011). The skill vehicle routing problem. In Network Optimization (pp. 354-364). Springer Berlin Heidelberg.

Cornillier, F., Boctor, F. F., Laporte, G., & Renaud, J. (2009). The petrol station replenishment problem with time windows. Computers & Operations Research, 36(3), 919–935.

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.

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.

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. (1997). Vehicle routing: handling edge exchanges. In: 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.

Lahyani, R., Khemakhem, M., & Semet, F. (2015). Rich vehicle routing problems: From a taxonomy to a definition. European Journal of Operational Research, 241(1), 1-14.

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.

Lin, S. (1965). Computer solutions of the traveling salesman problem. The Bell System Technical Journal, 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.

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.

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

Mole, R. H., & Jameson, S. R. (1976). A sequential route-building algorithm employing a generalised savings criterion. Journal of the Operational Research Society, 27(2), 503-511.

Oppen, J., & Løkketangen, A. (2008). A tabu search approach for the livestock collection problem. Computers & Operations Research, 35(10), 3213-3229.

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.

Rieck, J., & Zimmermann, J. (2010). A new mixed integer linear model for a rich vehicle routing problem with docking constraints. Annals of Operations Research, 181(1), 337–358.

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

Solomon M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. Operations Research 35(2), 254–265.

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.

Taillard, E.D. (1999). A heuristic column generation method for the heterogeneous fleet VRP. RAIRO-Operations Research, 33(1), 1-14.

Toth, P., & Vigo, D. (Eds.). (2014). Vehicle routing: problems, methods, and applications (Vol. 18). SIAM.

Vidal, T., Crainic, T. G., Gendreau, M., & Prins, C. (2013). Heuristics for multi-attribute vehicle routing problems: a survey and synthesis. European Journal of Operational Research, 231(1), 1-21.

Yuan, S., Skinner, B., Huang, S., & Liu, D. (2013). A new crossover approach for solving the multiple travelling salesmen problem using genetic algorithms. European Journal of Operational Research, 228(1), 72-82.