04/12/2024 02:58

(Article begins on next page)

Rich Vehicle Routing with Auxiliary Depots and Anticipated Deliveries: An Application to Pharmaceutical Distribution

Raphael Kramer[a,*], Jean-François Cordeau[b], Manuel Iori[a]

[a]*Dipartimento di Scienze e Metodi dell'Ingegneria, Università degli Studi di Modena e Reggio Emilia, Italy*
[b]*Department of Logistics and Operations Management, HEC Montréal, Canada*

Abstract

We present and solve a rich vehicle routing problem based on a practical distribution problem faced by a third-party logistics provider, whose aim is to deliver pharmaceutical products to healthcare facilities in Tuscany. The problem is characterized by having multiple depots, a heterogeneous fleet of vehicles, flexible time windows, periodic demands, incompatibilities between vehicles and customers, a maximum duration for the routes, and a maximum number of customers per route. A multi-start iterated local search algorithm making use of several neighborhoods is proposed to solve the problem. The algorithm has been tested on a large number of instances and obtained good results, both on the real case study and on a number of vehicle routing variants encountered in the literature.

Keywords: rich vehicle routing, pharmaceutical distribution, real-life logistics, iterated local search

1. Introduction

Vehicle routing constitutes a well-known class of combinatorial optimization problems that have been the subject of countless studies since the late 1950's. This is justified by their high potential of applications in many real-life situations, especially in the distribution of goods and in supply chain management, see, e.g., Golden et al. [1], Schmid et al. [2], and Toth and Vigo [3]. For instance, applications in the solid waste, beverage, food, dairy, and newspaper industries can be found in Golden et al. [4] and Coelho et al. [5]. In its basic definition, the capacitated *Vehicle Routing Problem* (VRP) aims at designing a set of minimum cost routes starting from and ending at a single depot, each route consisting of a sequence of customers to be visited once, in such a way that the total demand of the customers in each route does not exceed the vehicle capacity.

With respect to the basic model, VRPs faced by companies are often characterized by additional constraints, concerning operational aspects such as labor laws, market and environmental regulations, political decisions, contracts with customers, and so on. These details usually characterize the

---

*Corresponding author: Raphael Kramer. Dipartimento di Scienze e Metodi dell'Ingegneria, Università degli Studi di Modena e Reggio Emilia, via Amendola 2, 42122 Reggio Emilia, Italy. *E-mail address:* raphael.kramer@unimore.it

problem as a *Rich Vehicle Routing Problem* (RVRP). As defined by Lahyani et al. [6], RVRPs extend the academic problems by considering additional strategic and tactical aspects, as well as operational restrictions. In addition, according to Caceres-Cruz et al. [7], a RVRP must represent the most relevant attributes of a real-life vehicle routing distribution system, such as heterogeneity, multi-periodicity, and diversity of users and policies. A list and discussion of many of these characteristics can be found in Drexl [8].

In this paper, we present and solve a RVRP based on a practical distribution problem faced by an Italian *third-party logistics* (3PL) provider, whose aim is to deliver pharmaceutical products to hospitals and healthcare facilities in Tuscany. The problem is characterized by having: (i) multiple depots from where the vehicles may depart; (ii) a heterogeneous fleet of vehicles involving different capacities; (iii) customer time windows; (iv) time window flexibility, which can be obtained by installing warehouses dedicated to short pharmaceutical storage at some hospitals; (v) periodic demands distributed over a 6-day period; (vi) incompatibilities between vehicles and customers, due to, e.g., mountainous territory or city centers; (vii) a maximum time duration for each route; and (viii) a maximum number of customers that can be served per route.

To the best of our knowledge the proposed RVRP has never been addressed in the literature. The combination of these constraints makes the problem very interesting to study. Not only it models a real-world situation, but it also generalizes many other VRPS, such as, e.g., the VRP with time windows, the multi-depot VRP, and the multi-depot VRP with time windows. Because of these generalizations, it is easy to see that the proposed RVRP is NP-hard. In addition, the real-life instances that we solve are large and involve a few hundred customers. Hence, we decided to adopt a heuristic method. In particular, we developed an *Iterated Local Search* (ILS) that invokes, in sequence, constructive algorithms, local search procedures, and a perturbation phase. To speed up the local search and improve convergence towards good solutions, the ILS makes use of auxiliary data structures and accepts infeasible solutions at the expense of additional penalty costs. It also uses a multi-start method to further diversify the search. Due to the generality of the RVRP that it solves, the ILS is capable of effectively addressing many other VRP variants.

The remainder of this paper is organized as follows. Section 2 presents a brief review of the literature on vehicle routing problems associated with our case study. Section 3 gives a formal description of the problem. In Section 4 the proposed ILS is described in details. The algorithm is evaluated in Section 5 by means of extensive computational experiments, and conclusions are finally drawn in Section 6.

2. Literature review

The literature review presented in this section is divided into two parts: the first part focuses on practical VRP applications for pharmaceutical and healthcare distribution; whereas the second part is dedicated to recent RVRP applications and solution methodologies.

Liu et al. [9] studied a periodic *pickup-and-delivery problem* (PDP) for delivering drugs or medical devices from depots to patients' homes, or for delivering blood samples from patients' homes to laboratories, with the objective of minimizing the length of the longest route. They proposed a *tabu search* (TS) algorithm making use of multiple local search procedures, and computationally evaluated it on real-life data and classical instances from the literature. A non-periodic variant of the problem seeking the minimization of the total routing cost was earlier presented in Liu et al. [10] and solved by means of TS and genetic algorithms.

Magalhães and de Sousa [11] presented a pharmaceutical distribution case study arising in a dynamic multi-period environment, aiming at minimizing the delivery time and making use of a heterogeneous fleet of vehicles. In addition to the classical VRP challenges, decisions related with the postponement of deliveries while waiting for the arrival of new demands are also considered. The authors adopted a heuristic algorithm based on the cluster-first route-second principle, in which clusters are determined according to customer locations and vehicle capacities, while routes are built by using the least-cost insertion criterion. Once all routes are built, a local search procedure based on 2-opt exchanges is performed. This approach was able to reduce the average delivery time by about 8% when compared to the manual procedure adopted by the company.

Still in the context of pharmaceutical distribution, Ceselli et al. [12] studied a combined facility location and vehicle routing problem for delivering drugs (vaccines or treatments) either to citizens or to *distribution centers* (DC), with the aim of maximizing the total demand satisfied within a given deadline. In this application, DCs are supply facilities where customers have to go by their own means to have their demands attended. The opening of DCs is conditioned by a budget limit, and the customers that can be served by them are restricted by range and capacity constraints. In addition, customers visited at home are served by a heterogeneous fleet of vehicles, departing from multiple depots. The authors presented a set covering formulation and solved it by means of a branch-and-price-and-cut algorithm. Computational experiments were performed on artificial and realistic data considering the delivery of medicines in a scenario of bioterrorism emergency. A similar pharmaceutical distribution problem with DCs and traditional routes was earlier studied by Escuín et al. [13], who focused on a time-dependent environment in which a given number of DCs must be located to cover the largest number of customers by means of vehicle routes. Additional routes departing form a main depot are designed to supply DCs and customers not serviced by any DC. A mathematical model and an algorithm based on variable neighborhood search and TS were proposed, and a real case involving 211 pharmacies and 6 DCs was solved.

Motivated by the increasing competition in the pharmaceutical industry and the necessity for well-designed supply chains, Martins et al. [14] proposed an optimization-simulation approach for the so-called wholesalers network redesign problem, by taking into account both operational costs and service level. In a first stage, strategic-tactical decisions (e.g., location and capacity of warehouses, and allocation of customers to warehouses) are taken by solving a mixed integer

programming model. Then, in a second stage, a discrete event simulation model is adopted to evaluate and analyze the impact of the first stage solution in the wholesaler's daily activities. This approach was applied to redesign an existing supply chain network, and showed that simply changing a warehouse to a cross-docking may lead to savings of about 4% at the operational level. An attended home healthcare VRP with time windows and fuzzy demands was addressed by Shi et al. [15]. They developed a hybrid genetic algorithm integrated with stochastic simulation methods, and reported experimental results on a number of benchmark instances from the literature. Other recent studies on pharmaceutical distribution problems can be found in Fei et al. [16], Nsamzinshuti et al. [17], and Cieśla and Mrówczyńska [18].

2.2. RVRP applications and solution methodologies

The books of Golden et al. [1] and Toth and Vigo [3] are main references for the theory and practice on RVRPs. Since their publication, several interesting applications have been reported in the literature and solved with accurate methodologies. This section reviews recent applications that are related to our case study although not explicitly devoted to the solution of healthcare problems.

Inspired by the problem faced by a transport carrier, Mancini [19] dealt with a multi-depot periodic VRP with heterogeneous fleet, site-dependent constraints, and maximum duration for the routes. Customers have to be visited only once in the time period, and routes may finish in a depot different from their departure one. The author proposed a mathematical formulation and an *adaptive large neighborhood search* (ALNS) algorithm. The ALNS makes use of four destroy operators and repairs the solutions by solving to optimality a restricted mathematical model. Computational experiments performed on instances ranging from 50 to 75 customers showed that the ALNS was able to produce solutions which were 22% better on average than those obtained by the mathematical model (within a time limit of 1 hour).

A VRP with multiple time windows, heterogeneous fleet, and site-dependent incompatibility constraints was studied by Amorim et al. [20] in the context of food distribution in Portugal. Vehicles are classified as refrigerated trucks or dry trucks, and products are classified as dry, fresh, or cold. The vehicles allowed to visit a customer depend on the type of product demanded and on the customer's location. Moreover, customers may require to be serviced only during the morning (e.g., hospitals), or not to be serviced at lunch time. To solve the problem the authors used an ALNS metaheuristic composed by four destroy operators and six repair operators. The ALNS was able to improve the solution adopted by the company, reducing the costs by about 17%. Detti et al. [21] also present an application involving a heterogeneous fleet and incompatibility constraints. The authors study a PDP for transporting patients from pickup locations to healthcare facilities, and solve it by means of a *mixed-integer linear programming* model and by heuristic algorithms based on TS and *variable neighborhood search* (VNS).

Other studies on RVRPs with multi-depot and/or time window constraints have been presented

by Alinaghian and Shokouhi [22] and by Bruck et al. [23]. The former studied a problem with multi-compartment vehicles that has applications in fuel distribution and oil collection, and proposed a mathematical model and a hybrid ALNS algorithm to solve the problem. The latter developed an integrated heuristic approach to solve a practical attended home service problem faced by an Italian service provider of gas, electricity and water.

De Armas et al. [24] proposed a general VNS to minimize the total traveled distance and balance the difference between the longest and shortest routes of an RVRP with multiple time windows, heterogeneous fleet, site-dependent constraints, and customer priorities. The proposed algorithm was evaluated by solving real instances provided by a company from the Canary Islands, and also by solving benchmark instances for the VRP with time windows. It improved the solutions adopted by the company, generating more balanced routes and reducing the total traveling distance by about 30%. The approach was later extended in De Armas and Melián-Batista [25] to deal with a dynamic version of the problem in which customer demands are dynamically revealed over time.

The aforementioned works illustrate the variety and specificities of problems derived from real-life contexts. Some researchers proposed general framework algorithms with the purpose to address a wide class of VRP variants. Motivated by the request of a company developing software planning-tools for 3PL providers, Ceselli et al. [26] developed a column generation algorithm for solving vehicle routing problems that include several features, such as time windows, multiple depots, split deliveries, maximum route length and duration, drivers' resting periods, incompatibilities between goods, depots, vehicles, and customers, among others. The objective function is also flexible and may consider the total distance traveled, the number of pallets loaded, the total weight, the total volume, the total value, and the number of stops. The pricing problem is a resource-constrained elementary shortest-path, which they solved by means of a bounded bidirectional dynamic program. Relevant heuristic frameworks for solving practical RVRPs containing many of the characteristics cited before, or, in general, multi-attribute VRPs were presented by Derigs and Vogel [27], Vidal et al. [28], and Penna et al. [29], among others.

3. Problem description

We first provide a formal definition of the general problem that we are studying in Section 3.1. We then explain the details of the case study in Section 3.2.

3.1. General problem definition

We are given a graph $G = (N, A)$, with a set of nodes N and set a of arcs $A = \{(i,j) : i,j \in N, i \neq j\}$. A traveling distance d_{ij} and a traveling time t_{ij} are associated with each arc $(i,j) \in A$, and a time window $[e_i, l_i]$ with each node $i \in N$, with e_i being the earliest arrival time and l_i the latest. Vehicles can arrive earlier than e_i, but have to wait until e_i before starting their delivery operation. The set of nodes is divided into depots ($\mathcal{D}$) and customers ($\mathcal{C}$). Furthermore, depots are divided in *main depots* ($\mathcal{M}$) and *auxiliary depots* ($\mathcal{A}$), and customers ($\mathcal{C}$) in *small customers* ($\mathcal{S}$)

and *hospitals* ($\mathcal{H}$), implying $N = \mathcal{D} \cup \mathcal{C} = (\mathcal{M} \cup \mathcal{A}) \cup (\mathcal{S} \cup \mathcal{H})$. The main characteristics of each node type are:

- *Main depots* ($\mathcal{M}$) are managed and supplied by a regional entity and have homogeneous time windows. They are the starting points of all pharmaceutical products.
- *Auxiliary depots* ($\mathcal{A}$) are managed and supplied by the 3PL operator in charge of delivering the pharmaceutical products. They can be supplied with products coming from a main depot, through the use of a large truck in an exclusive route (from the main depot to the auxiliary depot, without visiting other nodes). These routes can be performed one business day before the expedition of the products to the customers, but not earlier (this limit is imposed by the regional entity). The time windows of the auxiliary depots are wider than those of the main depots, allowing a better operability.
- *Small customers* ($\mathcal{S}$) include healthcare facilities, clinics, and other related establishments, and are characterized by quite large time windows and low volume of demands.
- *Hospitals* ($\mathcal{H}$) are characterized by tighter time windows and much higher demands than the small customers. A subset $\mathcal{H}^*$ of hospitals differ from small customers also in the possibility of accepting *anticipated deliveries*, as they are equipped with warehouses where the pharmaceutical products can be stored for up to one day in advance with respect to their requested delivery.

Each customer may require deliveries in multiple days. Let $\mathcal{P} = \{1, 2, \ldots, |\mathcal{P}|\}$ denote the set of periods for which transportation must be planned. The demand to be delivered to a customer $i \in \mathcal{C}$ in a period $p \in \mathcal{P}$ consists of a volume q_{ip} and is associated with a service time s_{ip}. Each customer should be visited exactly once in each period having $q_{ip} > 0$.

Let $\mathcal{V} = \{1, \ldots, |\mathcal{V}|\}$ be the set of vehicle types. Each vehicle type $v \in \mathcal{V}$ has a capacity Q_v, and a cost h_v proportional to the traveled distance. Customer/vehicle incompatibility constraints should be taken into account. For instance, customers located in mountainous territory or city centers cannot be visited by large trucks. Formally, by assuming that $\mathcal{V}$ is sorted in decreasing order of capacity, the subset of vehicles allowed to visit a customer $i \in \mathcal{C}$ is denoted by $\mathcal{V}_i = \{\nu_i, \nu_i + 1, \ldots, |\mathcal{V}|\}$, where ν_i is the largest vehicle allowed to visit i. Vehicles can be located either at a main depot or at an auxiliary depot, and must return to their depot at the end of the route.

Based on labor regulations imposed on the 3PL operator, the total duration of each route cannot exceed a limit H, and the number of customers to be visited in a route cannot exceed an input parameter K.

The capacity of the vehicle used to supply the auxiliary depots is Q_1 (i.e., the largest truck), and the cost of each exclusive route to supply the auxiliary depot from the closest main depot is ω_k, for $k \in \mathcal{A}$.

Let us denote by s a generic solution respecting all problem constraints and by $\mathcal{R}_p(s)$ the set of routes adopted in period p by solution s. Let also $\sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_{|\sigma|}\}$ denote a generic route starting and ending at the same depot (i.e., $\sigma_1 = \sigma_{|\sigma|}$), $v(\sigma)$ the type of vehicle associated with the

6

route σ, and $L_{pk}(s)$ the total demand to be delivered in period p by vehicles departing from the auxiliary depot $k \in \mathcal{A}$ in solution s. The cost of s can be computed as

$$z(s) = \sum_{p \in \mathcal{P}} \sum_{\sigma \in \mathcal{R}_p(s)} \sum_{i=1}^{|\sigma|-1} h_{v(\sigma)} d_{\sigma_i, \sigma_{i+1}} + \sum_{p \in \mathcal{P}} \sum_{k \in \mathcal{A}} \omega_k \left\lceil \frac{L_{pk}(s)}{Q_1} \right\rceil. \tag{1}$$

The first term in the function computes the cost that depends on vehicle type and traveling distance, while the second term computes the cost to supply the auxiliary depots, where $\left\lceil \frac{L_{pk}(s)}{Q_1} \right\rceil$ is the number of trips required to supply the depot k at period p. The objective of our RVRP consists in routing the vehicles on each period to fulfill customer demands, while respecting constraints and minimizing (1). Following the notation commonly adopted to classify VRPs (see [3]), this RVRP can be classified as a multi-depot, heterogeneous fleet, periodic and site-dependent VRP with time windows.

3.2. Case study

The RVRP that we study was motivated by a public tender issued by the Italian state to select a 3PL operator to distribute pharmaceutical products to healthcare facilities in Tuscany. Companies interested in participating in the tender were asked to submit a technical proposal for the provision of the service, specifying the details of the proposed distribution systems, including location of the auxiliary depots and of the warehouses within the hospitals (if any) and design of the possible routes to be implemented to satisfy all demands. Some data concerning historical demands by customers were provided. We filled these data by including coordinates and transportation distances. We also clustered many of the customers into single locations (for example, all departments of a hospital are clustered into a single delivery point for the vehicle). The case study also takes into account the conditions imposed by the regional entity and the customers (e.g., customer demands per period, location and time windows of the main depots), and the decisions taken by the 3PL on the basis of tactical and strategic issues (e.g., the location and time windows of the auxiliary depots, the subset of hospitals to be equipped with warehouses).

The problem that we obtained is graphically depicted in Figure 1. It involves two main depots, two auxiliary depots, a few thousand customers clustered in a few hundred nodes that are spread in an area (Tuscany) of about 23000 km^2. The area is divided into three separate regions, each of which forms an instance of our problem. The three regions are called center, northwest and southeast in the following, and are depicted with three different colors in the figure. Although we consider the option that customers can be supplied with products coming from any of the two main depots and the two auxiliary depots, the instances are separated one from the other. Hence, a route serving customers of a region cannot serve customers of other regions.

Customer demands are measured in number of packages and are distributed aver a 6-day time horizon. Demands are quite high from Monday to Friday, and lower on Saturday. Six types of vehicles can be used to attend the customer deliveries, with capacities raging from 144 to 726
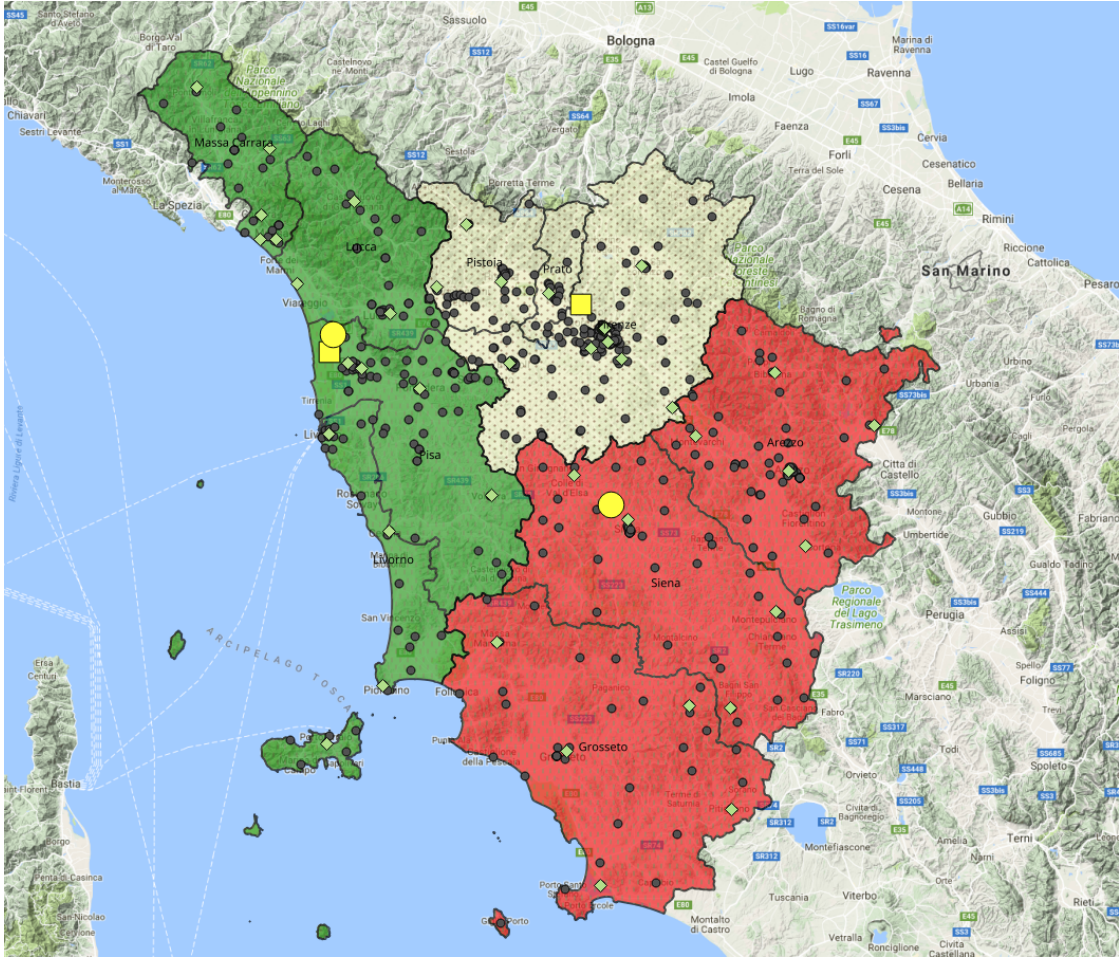
Figure 1: Region delimitations and node locations for the case study instances. Large yellow squares are the main depots installed by the regional entity; large yellow circles are the auxiliary depots planned by the 3PL company; small gray circles are the small customers; and green diamonds are the hospitals.

packages. Regarding the vehicle-customer incompatibilities, these depend on the size of vehicles, and thus the definition of the largest vehicle allowed to visit each customer is enough to represent these constraints. They particularly affect the southern and northern parts of the territory, which are quite mountainous, and the city centers (especially Florence). The maximum number of visits of a route cannot exceed 15, and the duration cannot exceed 8 hours. Finally, the time windows are set as follows: main depots from 5:15 to 23:00; auxiliary depots from 4:00 to 0:00; small customers from 7:00 to 14:00; and hospitals from 7:00 to 10:00 if no warehouse is installed, but an additional time window between 4:00 to 23:00 of the previous business day is allowed if the warehouse is installed.

4. Proposed Algorithm

To solve the RVRP of Section 3, we propose a *Multi-Start Iterated Local Search* (MS-ILS) algorithm, which invokes for a certain number of iterations a classical ILS method (see, e.g.,

Lourenço et al. [30]) based on a constructive procedure, local search, and a perturbation mechanism. To speed-up the local search (LS) phase and improve convergence towards good solutions, MS-ILS makes use of auxiliary data structures and accepts infeasible solutions (but penalizes them in the objective function). The details of the method are provided in the next Sections 4.1 to 4.4.

4.1. Preliminaries and main algorithm framework

The solutions evaluated during the LS phase are obtained by concatenating subsequences of vertices, and their costs are computed in amortized $\mathcal{O}(1)$ time by using the auxiliary data structures proposed by Vidal et al. [31]. For every subsequence, the following information is stored: total distance; minimum duration; cumulated load; earliest arrival time; latest arrival time; minimum *time-warp* use (i.e., units of time violating the time window constraints) to fix time window violations; largest vehicle allowed to attend the customers; and number of anticipated deliveries performed. These structures are initialized before the LS for every route, and updated every time the route is modified.

Due to the time window constraints and the maximum allowed duration for the routes, the LS and perturbation procedures may have difficulty in generating feasible solutions, thus compromising the convergence towards good solutions. We thus opted to accept infeasible solutions, but penalize them with an additional cost in the objective function. Let us define, for a given route r and period p, T_{pr} as the excess of time with respect to the total allowed route duration, and W_{pr} as the total time window violation. The cost function of a generic solution s is then

$$z'(s) = z(s) + \sum_{p \in \mathcal{P}} \sum_{r \in \mathcal{R}_p(s)} \lambda_1 T_{pr} + \sum_{p \in \mathcal{P}} \sum_{r \in \mathcal{R}_p(s)} \lambda_2 W_{pr}, \qquad (2)$$

where $z(s)$ is computed using (1), and λ_1 and λ_2 are the penalization costs associated to route duration and time window violations, respectively.

An informal MS-ILS pseudo-code is given in Algorithm 1. The algorithm makes use of three solutions: s_{best} is the incumbent, s_i is the best solution at a given iteration i, and s is the current solution. It starts by initializing s_{best} at line 2. At each iteration i, s is generated (line 4) and copied into s_i. Then, a LS is performed over s (line 7) and, if it improves s_i, the obtained solution is copied to s_i (line 9). Thereafter, a perturbation is performed on s_i (line 11), and the modified solution is stored in s. This ILS loop (lines 6-13) is repeated for η_{ils} consecutive iterations without improving s_i. When the ILS stop criterion is reached, if s_i is better than s_{best}, the incumbent solution is updated (line 14). After η_{iter} re-starts, the algorithm returns the best found solution (16).

4.2. Constructive procedure

An initial solution is constructed iteratively for each period by using Algorithm 2. Let LC_p be the set of customers to be visited in period p. Initially, for each hospital i belonging to LC_p we

Algorithm 1 Multi-Start Iterated Local Search

```
 1: function MS-ILS(Instance)
 2:     s_best ← ∅; z'(s_best) ← ∞                          ▷ Initialization of an empty solution
 3:     for i = 1, ..., η_iter do
 4:         s_i ← s ← GenInitSol(Instance)                   ▷ Constructive procedure
 5:         iterILS ← 1
 6:         while iterILS ≤ η_ils do
 7:             s ← LocalSearch(Instance, s)                  ▷ Local Search
 8:             if z'(s) < z'(s_i) then                       ▷ Acceptance criterion
 9:                 s_i ← s; iterILS ← 0
10:             end if
11:             s ← Perturbation(Instance, s_i)              ▷ Perturbation
12:             iterILS ← iterILS + 1
13:         end while
14:         if z'(s_i) < z'(s_best) then s_best ← s_i         ▷ Incumbent solution
15:     end for
16:     return s_best;
17: end function
```

create a route starting from the closest depot and associated with the largest vehicle allowed to visit i (line 6). Then, the remaining customers are inserted in the existing routes iteratively in a random order (lines 9–16). They are inserted according to a greedy strategy, i.e., in the route and position that leads to the lowest cost increase (line 11). In this step, only violations of the time window constraints are accepted, but penalized according to (2). Customers that could not be inserted in any existing route are served by new routes, which are created at line 13. The procedure terminates when all the customers have been assigned to a route for each period in which they need to be served, i.e., when $LC_p = \emptyset$ for all $p \in \mathcal{P}$.

Algorithm 2 Generate Initial Solution

```
 1: function GENINITSOL(Instance)
 2:     s ← ∅                                                ▷ Initialization of an empty solution
 3:     for p = 1 ... |P| do
 4:         LC_p ← Initialize the list of customers with demands on day p
 5:         for i ∈ LC_p ∩ H do
 6:             s ← s ∪ CreateRoute(p, i, closestDepot_i, ν_i)    ▷ ν_i is the largest vehicle allowed to visit i
 7:             LC_p ← LC_p \ {i}
 8:         end for
 9:         while LC_p ≠ ∅ do
10:             i ← SelectRandomCustomer(LC_p)
11:             GreedyInsertion(s, p, i)
12:             if i was not inserted then
13:                 s ← s ∪ CreateRoute(p, i, closestDepot_i, ν_i)
14:             end if
15:             LC_p ← LC_p \ {i}
16:         end while
17:     end for
18:     return s
19: end function
```

4.3. Local Search

The local search (LS) procedure is composed by seven neighborhoods based on elementary moves. Each elementary move generates a new solution by performing a single modification to the current solution. The implemented neighborhoods are:

1. *Relocate intra-route*: modify the position of a customer in a route;
2. *Swap intra-route*: exchange the positions of two customers in the same route;
3. *Relocate inter-route*: move a customer from its current route to a different route in the same period;
4. *Swap inter-route*: swap two customers served by two different routes in the same period;
5. *Relocate TP*: move a hospital from a route in period p, to a route in period $p-1$ (or period $p+1$), thus creating (or deleting) an anticipated delivery;
6. *2-opt*: invert the order in which a subset of customers is visited in a given route;
7. *restricted 2-opt**: exchange the last subsequence of customers of two different routes in the same period.

The list comprises both inter-route neighborhoods (3, 4, 5, and 7) and intra-route ones (1, 2, and 6). The neighborhoods are invoked according to the *randomized variable neighborhood descent* (RVND) scheme proposed by Subramanian [32], as shown in Algorithm 3. For each period p, a list NL of inter-route neighborhoods is initialized (line 3). Then, iteratively, a neighborhood $N \in NL$ is randomly selected (line 5) and the best solution s' that it can produce is identified (line 6). If the cost (computed according to (2)) of s' does not improve the cost of the current solution s, then N is removed from NL (line 11). Otherwise, s' is possibly improved by means of intra-route neighborhoods and is set as the new current solution (line 8), and NL is reinitialized (line 9). When NL is empty, the RVND procedure terminates and the best found solution is returned. The intra-route search at line 8 works as follows: it selects a random neighborhood from a pre-initialized list of neighborhoods and identifies the best solution that it can produce. In case of improvement, it replaces the current solution with the new one, and then executes a new local search by using the same neighborhood. When no improvement is obtained or $\tau_{\max}$ moves have been performed with the same neighborhood, then the neighborhood is removed from the list and a new neighborhood is randomly selected. The procedure terminates when the list of neighborhoods is empty.

4.4. Perturbation

The main role of the perturbation procedure is to modify a local optimal solution to allow the evaluation of new solutions in a subsequent local search phase, potentially leading to improvements of the incumbent solution. To this aim, two procedures were adopted, but only one is executed at each call according to a random choice. The first procedure executes φ random swap moves involving two different routes. The second one randomly selects a route starting from an auxiliary depot and forces it instead to start from a main depot – the one leading to the lowest cost. These two procedures have the capacity to generate solutions that cannot be obtained by the elementary

Algorithm 3 RVND

```
 1: function RVND(Instance, s)
 2:     for p = 1 ... |𝒫| do
 3:         NL ← list of all inter-route neighborhoods
 4:         while NL ≠ ∅ do
 5:             N ← SelectRandomNeighborhood(NL)
 6:             s′ ← BestNeighbor(s, N)
 7:             if z′(s′) < z′(s) then
 8:                 s ← IntraRouteSearch(s′)
 9:                 NL ← list of all inter-route neighborhoods
10:             else
11:                 NL ← NL \ N
12:             end if
13:         end while
14:     end for
15:     return s
16: end function
```

moves considered in the LS. In particular, the second perturbation guides the search to solutions that avoid the transportation of freight from main to auxiliary depots.

5. Computational Experiments

The performance of the proposed MS-ILS algorithm is evaluated by solving the realistic instances provided by the public tender, and also by solving artificial instances that we generated according to the characteristics of the realistic ones. Details on the instances are provided in Section 5.1. The experiments are divided into three parts. In the first part (Section 5.2), we evaluate the performance of the MS-ILS on the considered instances. In the second part (Section 5.3), we perform a sensitivity analysis that aims at estimating the gains that can be obtained by changing the number and locations of hospital warehouses and auxiliary depots. The last part (Section 5.4) reports additional computational experiments obtained by the MS-ILS on three VRP variants from the literature.

All our algorithms were coded in C++ and executed on a single thread of an Intel Core i5-5200U 2.2 GHz with 16 GB of RAM, running under Linux Mint 17.2 64-bit. To compute the shortest distance between each pair of nodes, we used the digital map of the region available at the free OpenStreetMap database (http://www.openstreetmap.org/). We read the map with the C library *shapelib* (http://shapelib.maptools.org/) and compute the shortest paths by using the library *lemon* (http://lemon.cs.elte.hu/trac/lemon). The following parameter values were adopted for the MS-ILS: $\eta_{\text{iter}} = 20$, $\eta_{\text{ils}} = 20$, and $\tau_{\max} = 100$. In all the experiments, each instance was solved 10 times.

5.1. Testbed instances

A set of three realistic instances and 45 artificial instances was considered. The realistic instances are based on the real data from Tuscany, as discussed in Section 3.2, and contain, respectively, 159, 161, and 232 nodes. In these instances the location of the auxiliary depots and the hospitals to be equipped with warehouses were decided a priori by the 3PL company and were provided as input to the MS-ILS. The artificial instances have 100, 150, 200, 250, or 300 nodes. For each size, we created nine instances by varying the number of main depots and auxiliary depots, by considering all pairs with $|\mathcal{M}| \in \{2, 3, 4\}$ and $|\mathcal{A}| \in \{2, 3, 4\}$. The coordinates of the nodes were randomly generated in a square region of $|N|^2$ area, with main depots located in the periphery and auxiliary depots located far from the borders (and close to the center). All instances have a time horizon of six periods, six types of vehicles with capacity 726, 372, 312, 226, 166, or 144, maximum duration of routes equal to 1/3 of the total duration of a single period, and maximum number of visits per route equal to 15. To stimulate further research on the problem, the artificial instances have been made available at http://www.or.unimore.it/resources/RVRP_PharmDist/home.html.

5.2. Computational results on realistic and artificial instances

As the problem results from a public tender, it has never been solved in reality so we cannot compare our solutions with real ones. We thus gain insight in the performance of our MS-ILS by comparing its best and average performance over the 10 runs with that produced by the constructive greedy that we provided in Algorithm 2. Since greedy strategies are commonly adopted by decision-makers, this can also give some indication of how much could be saved in the total cost by using the proposed method. The results for the realistic instances are reported in Table 1. The group of columns under "instance" contain the name and some input characteristics of each instance. The other columns refer to the incumbent solution found by the 10 attempts of the greedy and by the average and best solutions found by the MS-ILS. They have the following meaning: "cost" gives the total cost computed using (1) (the algorithms provided feasible solutions for all the instances, so the additional penalty function in (2) was always zero for the incumbent solutions); "$|\mathcal{R}|$" gives the total number of routes used; "$\#\overleftarrow{q}$" gives the number of anticipated deliveries; "time(s)" gives the computational time in seconds; and "gap(%)" reports the percentage gap of MS-ILS with respect to the best greedy solution found (computed as $100(z_{\text{MS-ILS}} - z_{\text{greedy}})/z_{\text{MS-ILS}}$).

The results displayed in Table 1 show a rare use of hospital warehouses for anticipated deliveries in the realistic instances. This is justified because the hospitals selected by the 3PL to host a warehouse have high value of demands (close to Q_1), so they are often served by single-visit routes or by routes containing just one or two customers with low demands. Anticipating these large deliveries does not allow the algorithm to produce consistent savings, because it cannot find good subsets of customers that would fit in the same vehicle. From the transportation cost point of view, this observation suggests that hospitals with smaller demands are more attractive to host a warehouse.

13

Table 1: Detailed results on the realistic instances.

instance						best greedy sol.		average MS-ILS solution					best MS-ILS solution			
name	$\|N\|$	$\|\mathcal{M}\|$	$\|\mathcal{A}\|$	$\|\mathcal{H}\|^{(a)}$	$\|\mathcal{S}\|$	cost	$\|\mathcal{R}\|$	cost	$\#\overleftarrow{q}$	$\|\mathcal{R}\|$	time(s)	gap(%)	cost	$\#\overleftarrow{q}$	$\|\mathcal{R}\|$	gap(%)
Center	232	2	2	17(9)	211	11489.1	137.0	7972.8	0.3	124.6	199.2	-30.6	7924.9	0.0	125	-31.0
Northwest	159	2	2	16(5)	139	15201.3	109.0	10633.1	0.2	87.2	83.1	-30.0	10464.0	0.0	88	-31.2
Southeast	161	2	2	14(0)	143	20490.9	82.0	16704.4	0.0	75.0	32.2	-18.5	16593.1	0.0	74	-19.0
avg.						15727.1	109.3	11770.1	0.2	95.6	104.8	-26.4	11660.7	0.0	95.7	-27.1

(a): in parentheses number of hospitals equipped to accept anticipated deliveries

A portion of the best solution obtained for the Center instance is depicted in Figure 2. An intersection of three routes can be noticed at the right of the main depot. This is imputed to the many constraints characterizing the addressed RVRP, as the vehicle-customers incompatibilities. Indeed, it can be noticed, on the right part of the figure, that these customers are located in a mountainous region and thus only small vehicles (with the lowest capacity) are allowed to visit them.
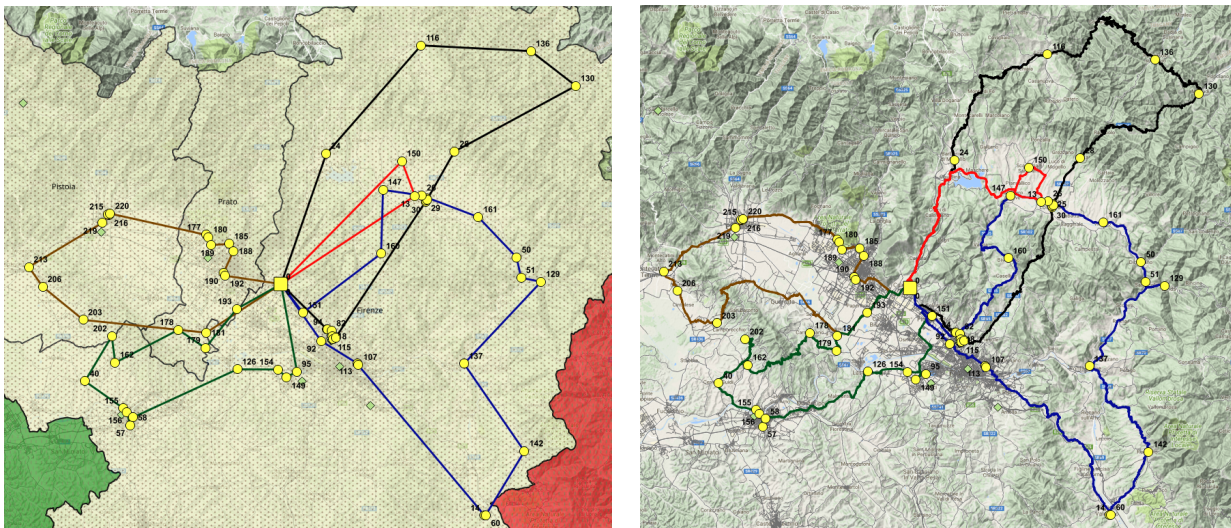
Figure 2: Example of five routes for the best solution of instance *Center*, on Friday. Left figure shows the routes in a simplified version. Right figure shows the routes considering real roads. Both figures represent the same sequence of visits.

The results for the artificial instances are reported in Table 2. Columns have the same meanings as those of Table 1, but now, for the sake of conciseness, each row presents aggregate results for a group of nine instances having the same number of vertices. Detailed results for each single instance are given in Appendix A. The results confirm that the MS-ILS achieves very important improvements (about 23% on average) with respect to the best solution found by the greedy heuristic. As expected, the computational time required to solve an instance increases with the instance size. However, the size of the instance does not affect the percentage gap, which varies very little. This shows the robustness of the MS-ILS algorithm. It is worth noting that in this

14

second round of tests the algorithm made a consistent use of anticipated deliveries. On average, about 17 anticipated deliveries were used by 133 routes, showing a good use of the warehouses installed at the hospitals.

Table 2: Aggregated results on the artificial instances.

instance group					best greedy sol.		average MS-ILS solution					best MS-ILS solution			
$|N|$	$|\mathcal{M}|$	$|\mathcal{A}|$	$|\mathcal{H}|^{(a)}$	$|\mathcal{S}|$	cost	$|\mathcal{R}|$	cost	$\#\overleftarrow{q}$	$|\mathcal{R}|$	time(s)	gap(%)	cost	$\#\overleftarrow{q}$	$|\mathcal{R}|$	gap(%)
100	2-4	2-4	11	81-85	8241.7	107.9	6487.1	10.1	79.0	11.3	-20.9	6404.9	11.8	78.0	-21.9
150	2-4	2-4	15	127-131	16614.1	139.4	12953.6	13.6	102.2	34.8	-21.9	12822.5	14.6	102.2	-22.7
200	2-4	2-4	21	171-175	27884.6	173.2	21585.1	15.0	133.8	60.2	-22.4	21321.7	15.3	132.8	-23.3
250	2-4	2-4	26	216-220	42844.2	209.9	31963.4	23.8	154.4	122.7	-25.0	31604.8	25.6	152.9	-25.8
300	2-4	2-4	30	262-266	57775.2	243.1	43273.7	22.1	193.6	174.6	-24.8	42883.6	24.0	194.9	-25.5
avg.					30671.9	174.7	23252.6	16.9	132.6	80.7	-23.0	23007.5	18.2	132.2	-23.8

(a): all hospitals equipped to accept anticipated deliveries

5.3. Sensitivity analysis

In this section we briefly report the outcome of additional tests that we conducted with the aim of better evaluating the characteristics of the problem that we deemed more interesting, namely, the possibility to anticipate the deliveries by means of warehouses in the hospitals and the use of the auxiliary depots.

Table 3 shows the results that we obtained by solving again the complete testbed of artificial instances, but this time neglecting the use of the warehouses in all the $|\mathcal{H}|$ hospitals. Apart from the details on the instances, the table reports values for the average MS-ILS solutions obtained under the original configuration of Table 2 (avg. MS-ILS sol.), values for the new more restricted configuration with no anticipated deliveries (avg. MS-ILS sol. – no warehouses), and percentage gaps between the solutions. No important change can be noticed on the number of routes. The new configuration is slightly easier to solve, because of the reduced number of transportation options, and this leads to a slight reduction in the computational time. The new configuration, however, obtains solution costs that are on average 2-5% worse than those obtained when the anticipated deliveries are allowed. This gap is very stable among the different groups of instances, ranging from 2.0% to 2.9%, and shows that the option of using larger time windows can indeed lead to consistent savings. In the best case (instance *n100m3a3*), a reduction of 5.46% can be obtained.

Similar additional tests aimed at evaluating the importance of artificial depots were also carried out. We considered again the artificial instances, but removed all auxiliary depots. The results are shown in Table 4, under the same format as Table 3. A considerable difference can be noticed between the two configurations. The existence of auxiliary depots with wide time windows allow indeed to obtain much better solution costs, having an average improvement of 6% with respect to the configuration in which no auxiliary depot exists. A greater use of anticipated deliveries can also be noticed in the configuration with no auxiliary depots.

Table 3: Evaluating the use of warehouses in hospitals to anticipate deliveries.

				instance				avg. MS-ILS sol.		avg. MS-ILS sol. – no warehouses			
$|N|$	$|\mathcal{M}|$	$|\mathcal{A}|$	$|\mathcal{H}|$	$|\mathcal{S}|$	cost	$\#\overleftarrow{q}$	$|\mathcal{R}|$	time(s)	cost	$|\mathcal{R}|$	time(s)	gap(%)
100	2-4	2-4	11	81-85	6487.1	10.1	79.0	11.3	6665.3	80.8	9.2	2.7
150	2-4	2-4	15	127-131	12953.6	13.6	102.2	34.8	13310.8	105.1	25.0	2.7
200	2-4	2-4	21	171-175	21585.1	15.0	133.8	60.2	22030.8	136.6	43.4	2.0
250	2-4	2-4	26	216-220	31963.4	23.8	154.4	122.7	32916.3	160.3	81.9	2.9
300	2-4	2-4	30	262-266	43273.7	22.1	193.6	174.6	44232.7	199.6	172.9	2.2
avg.					23252.6	16.9	132.6	80.7	23831.2	136.5	66.5	2.5

Clearly, the use of auxiliary depots, as well as that of warehouses in hospitals, comes at a cost. Savings in routing costs are obtained by means of installation of costly facilities. As in our case study we were not given estimations of these installation costs, but were instead provided with precise decisions on which depots and warehouses were adopted, we avoided studying the balance between routing and installation costs. We expect that studying the combination of such facility location and vehicle routing decisions (see, e.g., Prodhon and Prins [33] and Drexl and Schneider [34]) could possibly lead to further cost savings.

Table 4: Evaluating the use of auxiliary depots.

				instance				avg. MS-ILS sol.		avg. MS-ILS sol. – no aux. depots				
$|N|$	$|\mathcal{M}|$	$|\mathcal{A}|$	$|\mathcal{H}|$	$|\mathcal{S}|$	cost	$\#\overleftarrow{q}$	$|\mathcal{R}|$	time(s)	cost	$\#\overleftarrow{q}$	$|\mathcal{R}|$	time(s)	gap(%)
100	2-4	2-4	11	81-85	6487.1	10.1	79.0	11.3	6789.2	11.4	65.0	17.4	4.6
150	2-4	2-4	15	127-131	12953.6	13.6	102.2	34.8	13773.0	16.3	86.8	58.3	6.0
200	2-4	2-4	21	171-175	21585.1	15.0	133.8	60.2	23334.5	19.9	112.1	167.0	7.9
250	2-4	2-4	26	216-220	31963.4	23.8	154.4	122.7	33165.2	28.8	134.7	281.1	3.6
300	2-4	2-4	30	262-266	43273.7	22.1	193.6	174.6	46763.2	31.2	165.9	447.8	8.0
avg.					23252.6	16.9	132.6	80.7	24765.0	21.5	112.9	194.3	6.0

5.4. Computational results on VRP variants from the literature

To better evaluate the quality of the proposed MS-ILS and show its wide range of applicability, we performed additional computational experiments on some well-known VRP variants from the literature. In particular, we solved benchmark instances of the *VRP with time windows* (VRPTW), the *Multi-Depot VRP* (MDVRP), and the *MDVRP with time windows* (MDVRPTW). Due to the high number of instances, in this section we present only aggregate results, but refer to Appendix A for detailed information.

The results that we obtained on the VRPTW, the MDVRP, and the MDVRPTW are reported in Tables 5, 6, and 7, respectively, and correspond to the benchmark instances of Solomon [35], Cordeau et al. [36], and Cordeau and Laporte [37], respectively. The VRPTW is solved by considering

Euclidean distances truncated to the first digit. Given that the number of vehicles associated to each depot is an input value for both MDVRP and MDVRPTW instances, we set MS-ILS to accept infeasible solutions where vehicle capacities can be exceeded, but we added a penalization cost proportional to the excess of capacity used. The columns in the three tables have the same meaning, and report the cost of the solutions found ("cost"), the number of routes ("$|\mathcal{R}|$"), the computing time in seconds ("time(s)"), and the average percentage gap from the *best-known solution* (BKS) value ("gap").

Since the MS-ILS was designed to solve a richer and more complex problem, one cannot expect it to improve state-of-the-art algorithms for the "simpler" VRP variants addressed. Nevertheless, it is nice to notice that MS-ILS can achieve solutions that are not far from the BKS in reasonable computing times. The results on the VRPTW are very satisfactory: in a matter of a few seconds MS-ILS finds solutions that are on average less than 1.5% worse than the BKS value, and this gap is reduced to just 0.74% when considering the best MS-ILS solutions. The metaheuristic is less effective on the MDVRP and MDVRPTW test beds, maybe because these instances can be quite large and involve up to 360 nodes. Still it can provide feasible solutions in a matter of a few seconds or few minutes for all instances, and achieve gaps that are just 2-3% away from the BKS values.

Table 5: Aggregate results for the VRPTW – Solomon [35] instances.

instance group			BKS*		average MS-ILS solution				best MS-ILS solution		
group	$|N|$	#inst	cost	$|\mathcal{R}|$	cost	$|\mathcal{R}|$	time(s)	gap(%)	cost	$|\mathcal{R}|$	gap(%)
c1	100	9	826.70	10.00	827.48	10.00	2.93	0.09	826.70	10.00	0.00
c2	100	8	587.38	3.00	587.66	3.00	3.48	0.05	587.38	3.00	0.00
r1	100	12	1173.61	13.25	1192.31	13.72	5.23	1.74	1182.93	13.42	0.87
r2	100	11	872.51	5.45	890.52	4.84	5.46	2.04	880.04	5.18	0.84
rc1	100	8	1334.49	12.63	1374.65	13.69	5.16	3.07	1359.33	13.50	1.91
rc2	100	8	1000.68	6.25	1019.58	5.90	5.13	1.94	1008.59	6.00	0.82
avg.			973.24	8.64	989.39	8.72	4.63	1.51	981.39	8.71	0.74

*Best known solutions collected from Kramer et al. [38]

Table 6: Aggregate results for the MDVRP – Cordeau et al. [36] instances.

instance group			BKS*		average MS-ILS solution				best MS-ILS solution		
group	$|N|$	#inst	cost	$|\mathcal{R}|$	cost	$|\mathcal{R}|$	time(s)	gap(%)	cost	$|\mathcal{R}|$	gap(%)
p	≤ 99	6	948.27	8.50	950.11	8.68	1.10	0.31	948.27	8.50	0.00
	100-199	7	1613.75	14.71	1634.09	14.99	6.97	1.46	1622.70	14.86	0.80
	200-299	7	3856.66	24.86	3969.35	25.50	59.80	2.92	3935.99	25.43	2.04
	≥ 300	3	5751.92	35.33	5865.99	36.93	219.52	2.02	5830.27	36.33	1.39
pr	≤ 99	3	1086.07	6.00	1090.97	6.00	3.17	0.38	1087.33	6.00	0.10
	100-199	3	1842.32	12.33	1888.89	12.63	26.40	2.53	1870.01	12.67	1.49
	≥ 200	4	2502.24	20.75	2638.33	21.03	109.62	5.30	2611.27	21.00	4.22
avg.			2425.22	17.33	2485.32	17.77	50.29	2.07	2466.92	17.64	1.38

*Best known solutions collected from Vidal et al. [39]

Table 7: Aggregate results for the MDVRPTW – Cordeau and Laporte [37] instances.

instance group			BKS*		average MS-ILS solution				best MS-ILS solution		
group	$|N|$	#inst	cost	$|\mathcal{R}|$	cost	$|\mathcal{R}|$	time(s)	gap(%)	cost	$|\mathcal{R}|$	gap(%)
pr	≤ 99	6	2429.41	18.00	2507.55	17.75	138.41	2.70	2481.46	17.83	1.75
	100-199	6	2027.07	16.00	2106.80	14.92	89.63	3.72	2077.80	14.83	1.97
	≥ 200	8	2216.57	16.50	2336.50	16.43	74.41	5.08	2287.87	16.38	2.80
avg.			2223.57	16.80	2318.91	16.37	98.18	3.96	2282.92	16.35	2.23

*Best known solutions collected from Vidal et al. [31]

6. Conclusion

This paper introduced and formalized a multi-period vehicle routing problem with time windows inspired by a real situation arising in the field of pharmaceutical distribution. The problem is characterized by the existence of auxiliary depots and the possibility of anticipating the deliveries for a subset of customers. The interest for the problem is motivated by the fact that not only it is a real case study, but it also generalizes many other vehicle routing variants from the literature. To obtain good quality solutions we developed a multi-start iterated local search algorithm that makes use of several neighborhoods. Computational experiments were performed on a set of realistic and artificial instances and showed that anticipated deliveries and auxiliary depots play an important role in reducing the transportation costs. Additional experiments showed that the proposed algorithm is also capable of producing good quality solutions on related vehicle routing problems. Future research will focus on the study of anticipated deliveries in other multi-period vehicle routing problems.

Acknowledgments

References

[1] B. Golden, S. Raghavan, E. Wasil (Eds.), The Vehicle Routing Problem: Latest Advances and New Challenges, Operations Research/Computer Science Interfaces Series, Springer, 2008.

[2] V. Schmid, K. Doerner, G. Laporte, Rich routing problems arising in supply chain management, European Journal of Operational Research 224 (2013) 435–448.

[3] P. Toth, D. Vigo (Eds.), Vehicle Routing: Problems, methods, and applications, 2nd ed., SIAM, Philadelphia, 2014.

[4] B. Golden, A. Assad, E. Wasil, Routing vehicles in the real world: applications in the solid waste, beverage, food, dairy and newspaper industries, in: P. Toth, D. Vigo (Eds.), The Vehicle Routing Problem, Monographs on Discrete Mathematics and Applications, SIAM, Philadelphia, 2002, pp. 245–286.

[5] L. Coelho, J. Renaud, G. Laporte, Road-based goods transportation: a survey of real-world logistics applications from 2000 to 2015, INFOR: Information Systems and Operational Research 54 (2016) 79–96.

[6] R. Lahyani, M. Khemakhem, F. Semet, Rich vehicle routing problems: from a taxonomy to a definition, European Journal of Operational Research 241 (2015) 1–14.

[7] J. Caceres-Cruz, P. Arias, D. Guimarans, D. Riera, A. Juan, Rich vehicle routing problem: Survey, ACM Computing Surveys 47 (2014) 32:1–32:28.

[8] M. Drexl, Rich vehicle routing in theory and practice, Logistics Research 5 (2012) 47–63.

[9] R. Liu, X. Xie, T. Garaix, Hybridization of tabu search with feasible and infeasible local searches for periodic home health care logistics, Omega 47 (2014) 17–32.

[10] R. Liu, X. Xie, V. Augusto, C. Rodriguez, Heuristic algorithms for a vehicle routing problem with simultaneous delivery and pickup and time windows in home health care, European Journal of Operational Research 230 (2013) 475–486.

[11] J. Magalhães, J. P. de Sousa, Dynamic VRP in pharmaceutical distribution — a case study, Central European Journal of Operations Research 14 (2006) 177–192.

[12] A. Ceselli, G. Righini, E. Tresoldi, Combined location and routing problems for drug distribution, Discrete Applied Mathematics 165 (2014) 130–145.

[13] D. Escuín, C. Millán, E. Larrodé, Modelization of time-dependent urban freight problems by using a multiple number of distribution centers, Networks and Spatial Economics 12 (2012) 321–336.

[14] S. Martins, P. Amorim, G. Figueira, B. Almada-Lobo, An optimization-simulation approach to the network redesign problem of pharmaceutical wholesalers, Computers & Industrial Engineering 106 (2017) 315–328.

[15] Y. Shi, T. Boudouh, O. Grunder, A hybrid genetic algorithm for a home health care routing problem with time window and fuzzy demand, Expert Systems with Applications 72 (2017) 160–176.

[16] T. Fei, L.-Y. Zhang, W. Zhang, Distribution routing optimization in special drug based on improved ant colony algorithm, Journal of Discrete Mathematical Sciences and Cryptography 19 (2016) 607–621.

[17] A. Nsamzinshuti, F. Cardoso, M. Janjevic, A. Ndiaye, Pharmaceutical distribution in urban area: an integrated analysis and perspective of the case of Brussels-Capital Region (BRC), Transportation Research Procedia 25 (2017) 747–761.

[18] M. Cieśla, B. Mrówczyńska, Problem of medicines distribution on the example of pharmaceutical wholesale, in: S. Zawiślak, J. Rysiński (Eds.), Graph-Based Modelling in Engineering, Springer International Publishing, 2017, pp. 51–65.

[19] S. Mancini, A real-life multi depot multi period vehicle routing problem with a heterogeneous fleet: Formulation and adaptive large neighborhood search based matheuristic, Transportation Research Part C: Emerging Technologies 70 (2016) 100–112.

[20] P. Amorim, S. Parragh, F. Sperandio, B. Almada-Lobo, A rich vehicle routing problem dealing with perishable food: a case study, TOP 22 (2014) 489–508.

[21] P. Detti, F. Papalini, G. de Lara, A multi-depot dial-a-ride problem with heterogeneous vehicles and compatibility constraints in healthcare, Omega 70 (2017) 1–14.

[22] M. Alinaghian, N. Shokouhi, Multi-depot multi-compartment vehicle routing problem, solved by a hybrid adaptive large neighborhood search, Omega 76 (2018) 85–99.

[23] B. P. Bruck, J.-F. Cordeau, M. Iori, A practical time slot management and routing problem for attended home services, Omega (2018). In press.

[24] J. De Armas, B. Melin-Batista, J. Moreno-Prez, J. Brito, GVNS for a real-world rich vehicle routing problem with time windows, Engineering Applications of Artificial Intelligence 42 (2015) 45–56.

[25] J. De Armas, B. Melián-Batista, Variable neighborhood search for a dynamic rich vehicle routing problem with time windows, Computers & Industrial Engineering 85 (2015) 120–131.

[26] A. Ceselli, G. Righini, M. Salani, A column generation algorithm for a rich vehicle-routing problem, Transporta-

tion Science 43 (2009) 56–69.

[27] U. Derigs, U. Vogel, Experience with a framework for developing heuristics for solving rich vehicle routing problems, Journal of Heuristics 20 (2014) 75–106.

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

[29] P. Penna, A. Subramanian, L. Ochi, T. Vidal, C. Prins, A hybrid heuristic for a broad class of vehicle routing problems with heterogeneous fleet, Annals of Operations Research (2017). Forthcoming.

[30] H. Lourenço, O. Martin, T. Stützle, Iterated local search: Framework and applications, in: M. Gendreau, J.-Y. Potvin (Eds.), Handbook of Metaheuristics, International Series in Operations Research and Management Science, 2 ed., Springer, 2010, pp. 362–397.

[31] T. Vidal, T. Crainic, M. Gendreau, C. Prins, A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows, Computers & Operations Research 40 (2013) 475–489.

[32] A. Subramanian, Heuristic, Exact and Hybrid Approaches for Vehicle Routing Problems, Ph.D. thesis, Universidade Federal Fluminense, Computing Graduate Program, Niterói, Brazil, 2012.

[33] C. Prodhon, C. Prins, A survey of recent research on location-routing problems, European Journal of Operational Research 238 (2014) 1–17.

[34] M. Drexl, M. Schneider, A survey of variants and extensions of the location-routing problem, European Journal of Operational Research 241 (2015) 283–308.

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

[36] J.-F. Cordeau, M. Gendreau, G. Laporte, A tabu search heuristic for periodic and multi-depot vehicle routing problems, Networks 30 (1997) 105–119.

[37] J.-F. Cordeau, G. Laporte, A tabu search algorithm for the site dependent vehicle routing problem with time windows, INFOR: Information Systems and Operational Research 39 (2001) 292–298.

[38] R. Kramer, A. Subramanian, T. Vidal, L. Cabral, A matheuristic approach for the pollution-routing problem, European Journal of Operational Research 243 (2015) 523–539.

[39] T. Vidal, T. Crainic, M. Gendreau, N. Lahrichi, W. Rei, A hybrid genetic algorithm for multidepot and periodic vehicle routing problems, Operations Research 60 (2012) 611–624.

Appendix A. Detailed computational results

In this appendix, we provide the detailed computational results that we obtained on the instances that were discussed only in an aggregate form in the paper. In particular:

- Table A.8 reports detailed results for the artificial instances that we reported in aggregate form in Table 2 of the paper;

- Table A.9 refers to the VRPTW instances previously discussed in Table 5;

- Table A.10 refers to the MDVRP instances previously discussed in Table 6;

- Table A.11 refers to the MDVRPTW instances previously discussed in Table 7.

All columns in the tables reported in this appendix have the same meanings as those in the corresponding tables in the paper.

Table A.8: MS-ILS detailed results on the artificial instances (aggregate results in Table 2 of the paper)

instance group						best greedy sol.		average MS-ILS solution					best MS-ILS solution			
name	$\lvert N\rvert$	$\lvert\mathcal{M}\rvert$	$\lvert\mathcal{A}\rvert$	$\lvert\mathcal{H}\rvert$	$\lvert\mathcal{S}\rvert$	cost	$\lvert\mathcal{R}\rvert$	cost	$\#\overleftarrow{q}$	$\lvert\mathcal{R}\rvert$	time(s)	gap(%)	cost	$\#\overleftarrow{q}$	$\lvert\mathcal{R}\rvert$	gap(%)
n100m2a2	100	2	2	11	85	9286.4	92	7339.8	5.7	75.4	8.3	-21.0	7240.8	13	78.0	-22.0
n100m2a3	100	2	3	11	84	7839.8	96	6364.5	4.1	74.5	9.2	-18.8	6328.2	5	72.0	-19.3
n100m2a4	100	2	4	11	83	9599.3	119	7205.8	13.1	80.9	12.4	-24.9	7106.2	16	77.0	-26.0
n100m3a2	100	3	2	11	84	8917.1	103	6275.4	9.1	63.9	19.9	-29.6	6221.8	9	65.0	-30.2
n100m3a3	100	3	3	11	83	7978.5	103	5920.8	16.6	68.5	10.3	-25.8	5834.5	20	64.0	-26.9
n100m3a4	100	3	4	11	82	8328.7	120	6525.1	15.2	84.3	9.7	-21.7	6389.7	19	83.0	-23.3
n100m4a2	100	4	2	11	83	8139.3	112	6817.4	11.0	94.0	13.1	-16.2	6734.8	9	95.0	-17.3
n100m4a3	100	4	3	11	82	7085.2	101	5786.2	9.2	69.3	9.2	-18.3	5741.5	9	68.0	-19.0
n100m4a4	100	4	4	11	81	7000.6	125	6149.6	6.7	100.4	9.8	-12.2	6046.8	6	100.0	-13.6
n150m2a2	150	2	2	15	131	18769.2	126	14292.4	17.2	83.8	38.4	-23.9	14165.0	22	80.0	-24.5
n150m2a3	150	2	3	15	130	15082.3	127	11718.7	8.5	96.5	25.4	-22.3	11625.0	8	98.0	-22.9
n150m2a4	150	2	4	15	129	18932.2	132	14053.1	9.8	97.2	35.8	-25.8	13768.0	9	101.0	-27.3
n150m3a2	150	3	2	15	130	14991.6	136	11485.7	13.9	94.1	56.6	-23.4	11424.1	14	93.0	-23.8
n150m3a3	150	3	3	15	129	15873.4	144	12271.1	12.8	97.4	38.3	-22.7	12217.5	11	95.0	-23.0
n150m3a4	150	3	4	15	128	18608.0	154	14270.2	15.6	118.7	28.1	-23.3	14145.7	16	116.0	-24.0
n150m4a2	150	4	2	15	129	15506.4	136	11992.8	15.9	100.3	23.8	-22.7	11834.8	21	102.0	-23.7
n150m4a3	150	4	3	15	128	18155.8	148	15040.3	11.0	119.0	33.4	-17.2	14878.0	16	118.0	-18.1
n150m4a4	150	4	4	15	127	13607.6	152	11458.6	17.3	112.4	33.3	-15.8	11344.3	14	117.0	-16.6
n200m2a2	200	2	2	21	175	29270.5	164	21771.7	10.9	134.1	74.8	-25.6	21414.1	13	140.0	-26.8
n200m2a3	200	2	3	21	174	34719.0	179	24971.9	18.5	119.4	66.7	-28.1	24558.8	29	118.0	-29.3
n200m2a4	200	2	4	21	173	29103.0	180	23321.1	18.1	133.7	63.9	-19.9	23048.2	22	133.0	-20.8
n200m3a2	200	3	2	21	174	27508.8	140	21143.6	15.7	116.8	82.8	-23.1	20923.7	14	112.0	-23.9
n200m3a3	200	3	3	21	173	26070.8	173	18578.1	15.0	119.6	50.4	-28.7	18253.3	7	122.0	-30.0
n200m3a4	200	3	4	21	172	28491.8	171	23290.2	10.1	139.3	38.3	-18.3	23003.1	5	139.0	-19.3
n200m4a2	200	4	2	21	173	25652.5	190	19935.2	22.4	139.3	54.7	-22.3	19809.8	14	139.0	-22.8
n200m4a3	200	4	3	21	172	26159.4	165	21894.0	8.1	146.8	55.8	-16.3	21701.1	12	140.0	-17.0
n200m4a4	200	4	4	21	171	23985.5	197	19360.5	15.8	154.9	54.4	-19.3	19182.9	22	152.0	-20.0
n250m2a2	250	2	2	26	220	43649.7	207	33466.2	21.8	163.8	109.7	-23.3	32948.9	28	162.0	-24.5
n250m2a3	250	2	3	26	219	54802.8	217	36269.6	30.8	145.2	109.5	-33.8	35803.0	25	144.0	-34.7
n250m2a4	250	2	4	26	218	43214.9	229	32413.6	30.3	163.2	121.5	-25.0	32256.5	28	162.0	-25.4
n250m3a2	250	3	2	26	219	44154.0	205	33377.2	21.0	157.6	123.4	-24.4	33150.0	23	159.0	-24.9
n250m3a3	250	3	3	26	218	45406.7	201	33396.2	19.5	147.6	105.5	-26.5	33062.7	28	146.0	-27.2
n250m3a4	250	3	4	26	217	39457.7	209	31126.9	14.4	155.0	139.0	-21.1	30659.3	15	155.0	-22.3
n250m4a2	250	4	2	26	218	42672.5	197	32553.0	18.2	156.8	88.3	-23.7	32015.7	21	152.0	-25.0
n250m4a3	250	4	3	26	217	38557.3	209	28505.2	33.9	137.7	134.2	-26.1	28160.2	37	136.0	-27.0
n250m4a4	250	4	4	26	216	33681.8	215	26563.0	24.1	162.4	173.0	-21.1	26386.6	25	160.0	-21.7
n300m2a2	300	2	2	30	266	67360.2	230	49190.8	22.7	171.8	172.1	-27.0	48878.2	19	174.0	-27.4
n300m2a3	300	2	3	30	265	60720.3	246	47589.3	15.3	223.4	248.3	-21.6	47361.1	14	226.0	-22.0
n300m2a4	300	2	4	30	264	62095.7	247	45641.5	19.4	197.3	233.6	-26.5	45120.6	22	203.0	-27.3
n300m3a2	300	3	2	30	265	55336.4	225	43702.6	17.0	192.1	168.4	-21.0	43401.9	19	190.0	-21.6
n300m3a3	300	3	3	30	264	51794.5	246	40092.6	24.0	189.9	182.4	-22.6	39626.9	35	195.0	-23.5
n300m3a4	300	3	4	30	263	61396.6	254	43666.3	32.3	175.3	163.4	-28.9	42893.5	36	170.0	-30.1
n300m4a2	300	4	2	30	264	57267.1	245	39284.4	27.6	184.6	129.4	-31.4	38954.9	29	185.0	-32.0
n300m4a3	300	4	3	30	263	58205.0	248	41940.4	23.6	189.9	165.2	-27.9	41684.8	22	198.0	-28.4
n300m4a4	300	4	4	30	262	45801.2	247	38355.3	16.8	217.7	108.8	-16.3	38030.3	20	213.0	-17.0
avg.						30671.9	174.7	23252.6	16.9	132.6	80.7	-23.0	23007.5	18.2	132.2	-23.8

Table A.9: MS-ILS detailed results for the VRPTW (aggregate results in Table 5 of the paper)

instance	BKS*		average MS-ILS solution				best MS-ILS solution		
name	cost	$|\mathcal{R}|$	cost	$|\mathcal{R}|$	time(s)	gap	cost	$|\mathcal{R}|$	gap
c101	827.3	10	827.30	10.00	2.21	0.00	827.3	10	0.00
c102	827.3	10	827.30	10.00	2.89	0.00	827.3	10	0.00
c103	826.3	10	826.30	10.00	3.67	0.00	826.3	10	0.00
c104	822.9	10	829.92	10.00	4.14	0.85	822.9	10	0.00
c105	827.3	10	827.30	10.00	2.37	0.00	827.3	10	0.00
c106	827.3	10	827.30	10.00	2.44	0.00	827.3	10	0.00
c107	827.3	10	827.30	10.00	2.59	0.00	827.3	10	0.00
c108	827.3	10	827.30	10.00	2.95	0.00	827.3	10	0.00
c109	827.3	10	827.30	10.00	3.12	0.00	827.3	10	0.00
c201	589.1	3	589.10	3.00	2.38	0.00	589.1	3	0.00
c202	589.1	3	589.10	3.00	3.00	0.00	589.1	3	0.00
c203	588.7	3	588.70	3.00	4.45	0.00	588.7	3	0.00
c204	588.1	3	590.38	3.00	4.99	0.39	588.1	3	0.00
c205	586.4	3	586.40	3.00	3.19	0.00	586.4	3	0.00
c206	586.0	3	586.00	3.00	3.16	0.00	586.0	3	0.00
c207	585.8	3	585.80	3.00	3.31	0.00	585.8	3	0.00
c208	585.8	3	585.80	3.00	3.38	0.00	585.8	3	0.00
r101	1637.7	20	1639.42	20.00	5.35	0.11	1637.7	20	0.00
r102	1466.6	18	1470.06	18.00	4.60	0.24	1466.8	18	0.01
r103	1208.7	14	1221.28	14.60	5.20	1.04	1215.0	14	0.52
r104	971.5	11	998.32	11.50	5.27	2.76	986.8	11	1.57
r105	1355.3	15	1371.46	15.80	5.47	1.19	1361.9	16	0.49
r106	1234.6	13	1257.19	13.70	5.47	1.83	1249.7	13	1.22
r107	1064.6	11	1086.25	11.90	5.25	2.03	1071.8	11	0.68
r108	932.1	10	957.38	10.70	5.05	2.71	954.5	10	2.40
r109	1146.9	13	1166.43	13.40	5.26	1.70	1161.8	14	1.30
r110	1068.0	12	1095.77	12.20	5.24	2.60	1085.2	12	1.61
r111	1048.7	12	1072.25	12.00	5.22	2.25	1053.5	12	0.46
r112	948.6	10	971.90	10.80	5.41	2.46	950.4	10	0.19
r201	1143.2	8	1175.53	6.60	5.02	2.83	1157.7	7	1.27
r202	1029.6	8	1045.12	5.70	4.72	1.51	1036.4	7	0.66
r203	870.8	6	881.07	5.50	5.43	1.18	875.9	6	0.59
r204	731.3	5	743.49	4.50	5.20	1.67	734.4	4	0.42
r205	949.8	5	968.05	4.90	5.45	1.92	957.5	5	0.81
r206	875.9	5	896.88	4.40	5.88	2.40	882.7	5	0.78
r207	794.0	4	812.34	4.00	5.93	2.31	795.8	4	0.23
r208	701.0	4	713.85	3.40	5.62	1.83	705.5	4	0.64
r209	854.8	5	872.46	4.90	5.17	2.07	860.9	5	0.71
r210	900.5	6	925.99	5.30	5.53	2.83	919.4	6	2.10

Table A.9: MS-ILS detailed results for the VRPTW – continued from previous page

instance	BKS*		average MS-ILS solution				best MS-ILS solution		
name	cost	$|\mathcal{R}|$	cost	$|\mathcal{R}|$	time(s)	gap	cost	$|\mathcal{R}|$	gap
r211	746.7	4	760.96	4.00	6.10	1.91	754.2	4	1.00
rc101	1619.8	15	1660.33	17.30	5.01	2.50	1651.2	17	1.94
rc102	1457.4	14	1485.74	15.00	5.07	1.94	1479.7	15	1.53
rc103	1258.0	11	1322.95	12.20	5.36	5.16	1311.0	12	4.21
rc104	1132.3	10	1164.34	11.00	5.10	2.83	1155.1	11	2.01
rc105	1513.7	15	1553.04	16.50	5.32	2.60	1520.9	16	0.48
rc106	1372.7	13	1405.24	13.70	5.05	2.37	1392.0	14	1.41
rc107	1207.8	12	1259.27	12.70	4.48	4.26	1230.7	12	1.90
rc108	1114.2	11	1146.30	11.10	5.92	2.88	1134.0	11	1.78
rc201	1261.8	9	1278.02	8.30	4.98	1.29	1269.5	8	0.61
rc202	1092.3	8	1110.08	7.20	5.11	1.63	1095.6	7	0.30
rc203	923.7	5	944.92	5.10	5.79	2.30	936.6	5	1.40
rc204	783.5	4	793.51	4.00	5.52	1.28	783.5	4	0.00
rc205	1154.0	7	1166.92	7.10	5.37	1.12	1156.8	7	0.24
rc206	1051.1	7	1076.00	5.80	4.44	2.37	1064.5	5	1.27
rc207	962.9	6	990.33	5.40	4.56	2.85	972.0	7	0.95
rc208	776.1	4	796.86	4.30	5.30	2.67	790.2	5	1.82
avg.	973.24	8.64	989.39	8.72	4.63	1.51	981.4	8.71	0.74

*Best known solutions collected from Kramer et al. [38].

Table A.10: MS-ILS detailed results for the MDVRP (aggregate results in Table 6 of the paper)

instance			BKS*		average MS-ILS solution				best MS-ILS solution		
name	$|N|$	$|\mathcal{M}|$	cost	$|\mathcal{R}|$	cost	$|\mathcal{R}|$	time(s)	gap(%)	cost	$|\mathcal{R}|$	gap(%)
p01	50	4	576.87	11	580.27	11.00	0.44	0.59	576.87	11	0.00
p02	50	4	473.53	5	475.44	5.60	0.45	0.40	473.53	5	0.00
p03	75	5	641.19	11	646.27	11.30	1.49	0.79	641.19	11	0.00
p12	80	2	1318.95	8	1319.61	8.20	1.39	0.05	1318.95	8	0.00
p13	80	2	1318.95	8	1318.95	8.00	1.25	0.00	1318.95	8	0.00
p14	80	2	1360.12	8	1360.12	8.00	1.60	0.00	1360.12	8	0.00
p04	100	2	1001.04	15	1026.81	15.90	2.77	2.57	1018.26	16	1.72
p05	100	2	750.03	8	756.37	8.00	2.82	0.85	752.41	8	0.32
p06	100	3	876.50	16	890.88	16.00	2.91	1.64	887.27	16	1.23
p07	100	4	881.97	16	901.47	16.00	3.18	2.21	896.62	16	1.66
p15	160	4	2505.42	16	2545.27	16.10	12.99	1.59	2522.99	16	0.70
p16	160	4	2572.23	16	2577.27	16.00	10.09	0.20	2572.23	16	0.00
p17	160	4	2709.09	16	2740.56	16.90	14.02	1.16	2709.09	16	0.00
p18	240	6	3702.85	23	3791.66	24.60	43.97	2.40	3751.43	24	1.31
p19	240	6	3827.06	24	3854.47	24.00	39.63	0.72	3840.91	24	0.36
p20	240	6	4058.07	24	4126.95	25.60	63.01	1.70	4097.35	25	0.97
p08	249	2	4372.78	25	4564.23	25.70	65.70	4.38	4529.47	25	3.58
p09	249	3	3858.66	26	3995.24	26.00	60.22	3.54	3962.77	26	2.70
p10	249	4	3631.11	26	3784.20	26.20	69.30	4.22	3752.51	27	3.34
p11	249	5	3546.06	26	3668.74	26.40	76.74	3.46	3617.47	27	2.01
p21	360	9	5474.84	34	5667.70	36.90	205.05	3.52	5616.86	36	2.59
p22	360	9	5702.16	36	5763.53	36.00	170.67	1.08	5734.18	36	0.56
p23	360	9	6078.75	36	6166.75	37.90	282.83	1.45	6139.76	37	1.00
pr01	48	4	861.32	4	861.32	4.00	0.61	0.00	861.32	4	0.00
pr07	72	6	1089.56	6	1090.76	6.00	2.00	0.11	1089.56	6	0.00
pr02	96	4	1307.34	8	1320.85	8.00	6.90	1.03	1311.11	8	0.29
pr03	144	4	1803.80	11	1826.23	10.90	19.01	1.24	1809.25	11	0.30
pr08	144	6	1664.85	12	1721.03	12.00	13.32	3.37	1700.32	12	2.13
pr04	192	4	2058.31	14	2119.42	15.00	46.88	2.97	2100.47	15	2.05
pr09	216	6	2133.20	17	2216.33	17.10	42.61	3.90	2196.22	17	2.95
pr05	240	4	2331.20	19	2444.73	20.00	91.00	4.87	2412.63	20	3.49
pr06	288	4	2676.30	23	2794.84	23.00	183.63	4.43	2776.55	23	3.75
pr10	288	6	2868.26	24	3097.40	24.00	121.24	7.99	3059.69	24	6.67
avg.			2425.22	17.33	2485.32	17.77	50.29	2.07	2466.92	17.64	1.38

*Best known solutions collected from Vidal et al. [39].

Table A.11: MS-ILS detailed results for the MDVRPTW (aggregate results in Table 7 of the paper)

instance			BKS*		average MS-ILS solution				best MS-ILS solution		
name	$\|N\|$	$\|\mathcal{M}\|$	cost	$\|\mathcal{R}\|$	cost	$\|\mathcal{R}\|$	time(s)	gap(%)	cost	$\|\mathcal{R}\|$	gap(%)
pr01	48	4	1074.12	8	1079.61	8.00	1.06	0.51	1074.12	8	0.00
pr02	96	4	1762.61	12	1779.59	11.90	9.14	0.96	1768.38	12	0.33
pr03	144	4	2373.61	16	2418.96	15.70	32.15	1.91	2407.90	16	1.44
pr04	192	4	2815.11	20	2930.50	19.60	106.10	4.10	2900.21	20	3.02
pr05	240	4	2962.25	24	3091.25	23.60	200.55	4.35	3041.84	23	2.69
pr06	288	4	3588.78	28	3745.42	27.70	481.47	4.36	3696.28	28	3.00
pr07	72	6	1418.22	12	1426.90	10.00	4.72	0.61	1418.22	10	0.00
pr08	144	6	2096.73	18	2160.57	16.40	37.46	3.04	2119.80	16	1.10
pr09	216	6	2712.56	24	2794.71	22.70	145.75	3.03	2774.10	22	2.27
pr10	288	6	3464.65	30	3667.60	28.40	343.90	5.86	3636.82	29	4.97
pr11	48	4	1005.73	4	1055.46	4.00	0.14	4.94	1012.46	4	0.67
pr12	96	4	1464.50	8	1535.59	8.00	5.78	4.85	1505.38	8	2.79
pr13	144	4	2001.81	12	2027.66	12.00	21.76	1.29	2015.77	12	0.70
pr14	192	4	2195.33	16	2281.64	15.60	57.90	3.93	2256.69	15	2.80
pr15	240	4	2433.15	20	2587.97	20.00	98.02	6.36	2545.83	20	4.63
pr16	288	4	2836.67	24	2986.55	23.90	180.70	5.28	2932.45	24	3.38
pr17	72	6	1236.24	6	1305.20	6.00	0.69	5.58	1237.18	6	0.08
pr18	144	6	1788.18	12	1851.02	12.00	22.58	3.51	1807.18	12	1.06
pr19	216	6	2257.13	18	2343.35	17.90	98.79	3.82	2331.08	18	3.28
pr20	288	6	2984.01	24	3308.59	24.00	114.86	10.88	3176.76	24	6.46
avg.			2223.57	16.80	2318.91	16.37	98.18	3.96	2282.92	16.35	2.23

*Best known solutions collected from Vidal et al. [31].

26