A decomposition approach for the periodic consistent vehicle routing problem with an application in the cleaning sector

Bilal Messaoudi[a,b] , Ammar Oulamara[a,b,c,*], Said Salhi[c]
(a) University of Lorraine, LORIA laboratory,
Campus Scientifique - BP 239, 54506 Vandoeuvre-les-Nancy, France.
(b) Antsway SA, 92, rue du sergent Blandan, ARTEM, 54000 Nancy
(c) Centre for Logistics & Heuristic Optimisation (CLHO),
Kent Business School, University of Kent, Canterbury CT2 7FS, UK

Abstract

This study is inspired by a challenging logistic problem encountered in the cleaning service sector. The company wishes to solve the consistent vehicle routing problem over a three-month planning horizon. The company has a heterogeneous vehicle fleet to guarantee multiple frequencies of visits to its customers. The objective is to minimize the number of vehicles used and the total distance traveled. This problem is a generalization of the periodic vehicle routing problem. We decompose the problem into two sub-problems, namely, the planning and routing optimization sub-problems. We construct a mathematical model for the former and a large neighborhood search for the latter. We evaluate the performance of our approach using the results of the industrial partner and instances from the literature on problems that are closely related to our case study. Our approach is found to be effective and robust. Our results outperform the existing company's plan in terms of solution quality, and staff convenience, and speed. We also discovered new best solutions on some of the instances from the literature.

Keywords. routing, driver consistency, mixed-integer linear programming, large neighborhood search

1 Introduction

The vehicle routing problem (VRP) is used to model several real applications, mainly in the transportation area. Since the introduction of the classical VRP by [17], several variants have been investigated and some are motivated by real life applications. One of these variants is the Periodic Vehicle Routing Problem, short for PVRP. Here, a planning horizon of several days is usually considered alongside a set of customers that are to be visited more than once with different frequencies. These visits are performed evenly at the route planning level such that customer' demands are spread in a balanced manner over the planning horizon allowing (i) an efficient management of company' resources such as drivers and vehicles and (ii) the design pf basic cyclic routes in each corresponding planning period

*Corresponding author: ammar.oulamara@univ-lorraine.fr

[26]. It is also worth noting that the companies do now give considerable importance to the satisfaction of their customers. This quality of service is, for the case of routing, enhanced by a personalized service, where for example, a driver is dedicated to serve the same set of customers which allows to establish a personal relationship. Developing these customer-based aspects of relationship requires routing to be consistent over multiple time periods. This type of consistency can be either temporal in which customer is visited at roughly the same time of the day and at regular intervals [19], or driver-oriented, in which the same driver regularly visits the same customers [22]. For instance, in several studies including [30], [54] and [61] both temporal and driver consistencies are considered. The integration of consistency in routing problems combined with the periodicity challenge, led operations managers to develop routing plans that make the best use of the available resources while gaining a competitive advantage for their organization.

To this end, this paper addresses a new practical and complex industrial periodic VRP in the cleaning sector with consistency constraints. Here, consistency constraints relate to (a) the driver consistency in which each driver visits the same customers over the planning horizon and (b) day consistency in which visits to the same customer are scheduled in the same day of the week over the planning horizon.

The rest of the paper is organized as follows: in Section 2, we present the industrial case study and summarizes our contributions. Section 3 provides a brief review focusing on two categories, namely, the periodic VRP and consistency VRP problems. In Section 4, provides a formal description and the necessary notations and develop a mathematical model formulation of the problem. Our solution method is presented in Section 5. Computational results are provided and analysed in Section 6. Finally, some conclusions and research directions are outlined in Section 7.

2 Industrial application and contributions

We address a logistic problem faced by a French company that provides a washroom cleaning service for business clients, including the supply of products such as cleaning gel and toilet paper, among others.

Currently, the company serves about 6,000 customers over a 12 week cyclical planning horizon. Customers are geographically located in the Ile-De-France region (about 12,000 km^2 area in France). The company systematically assigns each customer to the same agent (driver). This is performed so that each agent becomes efficient at carrying out his/her other work. This may include identifying the toilets location at the customers' sites, especially those with several toilets scattered in different levels of the same building.

The company has a vehicle fleet of 40 heterogeneous vehicles and each vehicle starts and ends its route at the depot the time. The maximum duration of each route is imposed by regulations and cannot be violated. The company offers 14 different products. A customer's request consists of a list of products and for each product, a quantity and a delivery frequency over the 12-week planning horizon. A frequency represents the number of customer visits (product delivery) over the 12 weeks and is predefined by the value set 1, 2, 3, 6 and 12. Customers usually require more than one product. For example, a customer wants to have toilet paper delivered once a week (frequency equal to 12) and hand cleaning gel every two weeks (frequency equal to 6). The service time for a customer visit consists of the cleaning service time plus the product delivery service time. The objective is to meet customer demands with the smallest number of vehicles, while respecting the following operational constraints,

(i) *vehicle capacity* - the demand of customers on each route must not exceed the capacity of the vehicle,

(ii) *the duration of a route* - the total duration of a route must not exceed the duration of a working shift,

(iii) *same day of visit* - each customer is visited on the same day of the week (day consistency), and he/she must be visited no more than once a week, and finally

(iv) *same agent for a visit* - each customer is visited by the same agent (driver consistency) when a visit is scheduled for this customer.

To plan the delivery routes to its customers, the company currently uses a basic vehicle routing software. Due to the high level of customer service requirements and for the company to retain its competitive advantage, this software does no longer meet all the company's constraints. This weakness often results in requiring several modifications that are to be performed by hand. These changes can, in some circumstances, be very difficult to carry out manually. Besides, this often requires the assistance of an expert who is familiar with the functioning of the company. The quality of the solution that is generated, in most cases, falls short of the standard expected by the company. This is particularly reflected in terms of the total distance travelled and the number of vehicles used. In addition, for simplicity the management team wishes also to reduce the dependency on the human expertise.

The purpose of this work is to develop a new optimization tool based on a decomposition approach and metaheuristic that incorporates all constraints regarding the planning and the routing while minimizing the number of vehicles used and the total distance travelled. It is worth noting that minimizing the number of vehicles used every day of the planning horizon needs to take into account the important aspect of driver consistency with respect to balancing of the workload between days and weeks. As a by-product, one of the practical goals is also to reduce, or even avoid, the reliance on manual changes to the solution obtained. This is important as it may cause human errors and extra unnecessary costs and resources that can be reduced providing an additional competitive edge to the company.

More generally, this paper addresses a practical variant of a periodic routing problem with driver and day consistency constraints as described in the industrial case study. It is worth noting that our approach can be applied not only to this specific industrial application in the cleaning sector, but also in several other areas including home health care services as investigated by ([2]) where nurses always visit the same patients at regular periods. The objective is to minimize the number of vehicles used and the total distance travelled. The problem of this industrial case study falls under the category known as multi-level combinatorial optimization. This interesting logistical practical problem was introduced by [38] where a simple two-phase approach was presented. In phase one, the customers are first assigned to weeks and then to each day of the week. In the second phase, a basic routing heuristic is used to construct the delivery routes. An initial testing of this basic approach against the implementation of the company was positively received by Senior Management who invited us to carry out a deeper study by investigating thoroughly many of the aspects. This study, though is based on the same application, it has several differences as clearly outlined in the following contributions:

(i) We reexamine and formally introduce the two phase decomposition approach of [38]. In the first phase, a novel customers clustering approach based on location analysis is introduced for scheduling customers on weeks and days. In the second phase a more innovative and powerful routing method is developed. This is demonstrated by the massive reduction in the number of vehicles (from 5 to 17) when tested against the current implementation of the company. This will be shown in Section 5.2.

(ii) We transformed our approach into a more flexible and powerful technique that also tackle related routing problems efficiently. This is adapted accordingly and tested against the state-of-the-art methods on instances from problems in the literature that are closely related to our case study.

The results obtained are found to be encouraging including the discovery of new best solutions.

As will be shown in the section 3, to the best of our knowledge, this problem of periodic vehicle routing with consistent constraints has not been considered in the literature, it is characterized by the following characteristics: (i) double granularity of the planning horizon, namely, week and day granularity, (ii) each customer requires several products, and each product has its own frequency of visits, (iii) the days and weeks of customer visits are decision variables, and (iv) a double consistency, i.e., agent consistency and day consistency.

3 Literature review

In this section we provide a review on two related routing problems to ours, namely, the periodic vehicle routing problem and the consistent vehicle routing problem.

3.1 Periodic vehicle routing problem

The periodic vehicle routing problem (PVRP) is a generalization of the classical vehicle routing problem (VRP) where the planning horizon (e.g., one or several weeks) is composed of multiple periods (e.g., several days) and customers are visited several times according to either a set of visit alternatives based on the frequency of the requested products/services or a fixed set of periods specified by the customer. For example, if the planning horizon is one week (5 working days), and either the customer fixes the number of visits, e.g., needs to be visited twice a week with at least two days between two consecutive visits, then the possible pairs of visit days would be $(1, 4)$, $(1, 5)$ and $(2, 5)$ or the customer fixes the periods of visits e.g., days 1 and 4. The problem is to determine the visiting option for each customer simultaneously with the routing decision while minimizing the total cost over the planning horizon.

The PVRP has attracted a considerable amount of interest among researchers and practitioners [57]. This is mainly due to the wide range of real-world applications that fit into this class of routing. Among the applications, we can cite a few such as waste and garbage collection ([56], [37]), animal waste ([15]), home health care services ([2]), delivery of blood products to hospitals ([25]), retail stock supply ([43]), maintenance service ([8]), perishable products ([21]), and maritime surveillance [18]. Other real life applications are reported in [10].

The problem was introduced by [7] where the authors studied the routing problem related to municipal waste collection. They considered a one week planning horizon with the objective of minimizing both the number of used vehicles and the total travel time. Constructive heuristics were developed to solve independent VRPs for each day of the period. [48] proposed a formal definition of the problem and developed three cluster-based constructive methods with the aim to minimize the total distance travelled per week. In that study, the number of vehicles was considered given. [14] provided the first mathematical formulation for the PVRP and proposed a decomposition approach where customers are first assigned to days followed by solving a VRP for each day.

There is a lack of research on exact methods for the PVRP compared to its VRP counterpart. A special case of the PVRP is considered in [36] where a branch-and-cut method is developed. The objective is to minimize the total distance for a two-period travelling salesman problem in which some customers are visited in both periods while the others are visited in either one of the two periods. [20] studied a variant of the PVRP in which the service frequency is also a decision variable with the objective of maximizing service benefits and minimizing routing costs. They proposed an interesting hybrid solution method based on Lagrangian relaxation and a branch-and-bound procedure. [40] put forward an exact method for the PVRP with the objective of balancing the workload across vehicles and spacial compactness of the routes. The authors developed a Dantzig-Wolfe reformulation and a column generation approach to solve the relaxed problem. [5] proposed a new formulation in generating strong lower bounds for the problem which are then used to restrict the set of routes without affecting optimality. [46] addresses the periodic vehicle routing problem with time windows (PVRPTW). An exact branch-and-price-and-cut algorithm is proposed in which the pricing problems are elementary shortest-path problems with resource constraints. In [26] authors address a PVRP in which grocery retailer are supplied from a distribution center according to repetitive delivery patterns. The authors model the problem as a mixed integer program, then they propose a sequential heuristic procedure that construct a set of retailers clusters first, then in second step assign delivery pattern to customer and routes of vehicles. A numerical study shows that operational costs is significantly reduced. [3] introduces the flexible PVRP where each customer has a total demand that has to be satisfied by the end of the planning horizon and the quantity delivered at each visit is a decision variable and should not exceed the customer storage capacity. They propose a mathematical formulation for the problem, together with some valid inequalities. [4] extends the work of [3] and proposes a two-phase matheuristic approach, where an initial solution is built using a MILP model in phase one, followed by a Tabu Search in the second phase. It seems that no contribution to the exact solution of the classical PVRP has been achieved after [5]. The main works in the literature has so far been focused on the study of new variants inspired by applications [39] instead.

Meta-heuristics are the most adopted approaches for solving the PVRP and its variants. For instance, [49] presented a solution method that consists of an initial route design, followed by three different improvement phases aimed at escaping from the local optima. [16] presented a tabu search method for solving the periodic multi-depot vehicle routing problem (MDVRP) in which two types of neighborhood operators are proposed. [1] considered a periodic pick-up of raw materials for a manufacturer of automobile parts. They proposed a Scatter Search based on a two-phase approach for which the first phase assigns orders to days and the second constructs routes for each day. [25] put forward a variable neighborhood search (VNS) algorithm with the solution acceptance being based on simulated annealing, [58] developed a hybrid genetic algorithm (GA) for the PVRP, and [62] proposed an adaptive large neighborhood search (ALNS). Others recent works have been mostly on variants of the basic PVRP including the consideration of time windows (PVRPTW), where customers are allowed to be served within a specific time interval of each period. For instance, [41] proposed a hybrid genetic algorithm, and [60] developed a heuristic algorithm based on improved ant colony optimization (IACO) enhanced by simulated annealing (SA). Another variant that has mostly been considered in the literature is the multi-depot PVRP in which the customers are allowed to be served from multiple depots. Here, [12] developed a reactive greedy randomized adaptive search procedure, and [13] proposed a hybrid genetic algorithm. Interesting and informative surveys on periodic vehicle routing problems can be found in [10] and [39].

3.2 Consistent vehicle routing problem

In recent years, Vehicle Routing Problems (VRPs) with consistency features have received significant attention due to their practical importance. Three types of consistency are usually considered in the literature [59]; (i) driver consistency, (ii) time consistency and (iii) quantity-delivered consistency. Driver consistency imposes that the same driver visits the same customers on each day they require service over a planning horizon. Time consistency on the other hand requires visits to the same customers at approximately the same time on each day they require service. Finally, the quantity-delivered consistency constrains delivery quantities within lower and upper bounds at each visit to the same customer while satisfying the total quantity at the end of the planning horizon. The consistency constraints (driver and time consistencies) are particularly important in real applications. For example, in home healthcare service [47] where the operator (driver) knows the needs and preference of their patients (customers).

The consistency constraints (driver and time consistencies) were formally introduced in [24]. They developed a two-stage algorithm where the first stage constructs a template routes that consists only of those customers that require several visits followed by the generation of the daily schedules in stage two. [55] adopts the same template routes principle to propose a two-level tabu search algorithm, in which template routes are constructed at the high level, and then the daily schedules are optimised at the low tabu search level. In [30] a template-based Adaptive Large Neighbourhood Search algorithm (ALNS) is developed to solve the driver and time consistent periodic VRP problem. [61] and [31] also solve the same problem but using a variable neighborhood search (VNS) algorithm instead. In [53] a hierarchical Tabu Search framework is proposed in which an upper-level Tabu Search method is combined with variable neighbourhood descent algorithm. [38] considered a PVRP problem with day visits and driver consistencies. A heuristic decomposition approach is proposed to solve the problem with the objective of minimizing the number of vehicles and total cost. [19] considered the time consistency only, in which the day is discretized in time windows and imposes consistency by bounding the number of different time segments during which a customer is served. Here, a large neighborhood search heuristic is proposed to solve the PVRP problem. Exact approaches are also developed in the literature to the consistent periodic VRP problem. For example, [11] tackled a consistent VRP in a pharmaceutical distribution company, with multiple daily deliveries. The authors developed an interesting decomposition algorithm based on mathematical programming model. Other works include [22] who proposed an exact method based on column generation, and [42]who constructed an integer linear programming formulation. In the latter paper, several useful families of valid inequalities are constructed and an exact branch-and-cut algorithm is then developed to solve the problem. Other extensions for the consistent periodic VRP were also considered in the literature such as including time windows of visits and flexible driver consistency. For instance, [52] focused on maximizing the number of times a unique driver visits each customer and in both [9] and [34] a limit of the number of different drivers serving any customer was imposed. In [35] a time and collaborative consistency were considered where carriers can exchange customers who have to be serviced on a regular basis, and in [54] driver consistency PVRP with profit was considered. Table 1 summarizes some of the published work in this area.

In this study, we will be extending and enhancing the work of [38] mentioned in Section 1, and also comparing our results against those recently produced by [42] and [22] to assess the performance of the proposed algorithm.

Table 1 – Comparison of our paper with with related works in the literature.

Paper	Periods of the planing	Types of visits	Consistency Constraints	Others Constraints	Objective (Min)	Methods
[24]	Days	Known and fixed	Driver consist. Arrival time consist.	Vehicle capacity limit, Travel time limit	Total travel time	Record-to-Record
[55]	Days	Known and fixed	Driver consist. Arrival time consist.	Vehicle capacity limit, Travel time limit	Total travel time	Template-based Tabu search (TTS)
[30]	Days	Known and fixed	Driver consist. Arrival time consist.	Vehicle capacity limit, Travel time limit	Total travel time	Template-based ALNS
[29]	Days	Known and fixed	Limited number of drivers	Vehicle capacity limit, Travel time limit	Total travel time and Arrival time difference	Large neighbourhood search (ALNS)
[19]	Days	Known and fixed	Visiting time consist.	Vehicle capacity limit	Total travel time	Large neighbourhood search (LNS)
[61]	Days	Known and fixed	Driver consist. Arrival time consist.	Vehicle capacity limit, Travel time limit	Total travel time	Variable neighbourhood search (VNS)
[22]	Days	Known and fixed	Driver consist.	Vehicle capacity limit, Travel time limit	Total travel time	Exact method (column generation)
[42]	Days	Decision variables from a set of patterns	Driver consist.	Limited number of customers per route	Total travel cost	Exact method (branch-and-cut)
[54]	Days	Known and fixed	Driver consist Arrival time consist.	Vehicle capacity limit, Travel time limit	Maximize the net profit	Adaptive Tabu Search
[31]	Days	Decision variables from a set of patterns	Driver consist.	Vehicle capacity limit, Travel time limit	Total travel distance	Variable neighborhood search (VNS)
[53]	Days	Known and fixed	Driver consist.	Heterogeneous vehicles, Compatibility vehicle-customer, Travel time limit	Total travel cost	Hierarchical Tabu Search (HTS) Search (HTS)
This paper	Weeks and Days	Decision variables from a set of patterns	Driver consist. Day consist.	Heterogeneous vehicles, Vehicle capacity limit, Travel time limit, Time windows, Multi-products demand per customer	Total travel distance and number of vehicles	Decomposition method (MIP and LNS)

4 Problem description, notations and formulation

4.1 Problem description and notations

The problem is defined as follows. Let $G = (V, A)$ be a network with $V = \{0, \ldots, n\}$ a set of $n+1$ nodes and A is a set of arcs. The customers to be visited are represented by nodes $1, \ldots, n$, and node 0 is the depot. For each arc $(u, v) \in A$, we define a travel time t_{uv} and a travel distance d_{uv}. The planning horizon consists of the set $\mathcal{W} = \{1, \ldots, W\}$ weeks and each week contains $\mathcal{D} = \{1, \ldots, D\}$ of days. A customer is visited during this time horizon according to frequencies that depend on the type of requested products. We denote by P the set of types of products available at the depot. A visit to a customer involves delivering a set of products as well as an on-site activity such as toilet cleaning. Each customer i is characterized by a time window $[e_i, l_i]$ of visits and a list P_i of requested products with $P_i \subseteq P$. For each product $j \in P_i$, the frequency f_{ij} of visits, the service time s_{ij} and the quantity q_{ij} requested at each visit during the planning period are known. The frequency f_{ij} represents the number of times customer i receives product j over the planning horizon of W weeks. Furthermore, a customer is always visited on the same day of the week defining the day consistency, and by the same agent representing the driver consistency. In our industrial case, the possible frequencies for each product are taken from the set $\{1, 2, 3, 6, 12\}$ and the planning horizon $W = 12$. For example, if $f_{ij} = 3$, customer i receives product j three times during the planning horizon and the visits must be spread and well-balanced over the W weeks. Thus, for each frequency f_{ij}, we define a set R_{ij} of all possible combinations of weeks in the planning period, we call them patterns and only one pattern is selected from R_{ij}. For example, if $f_{ij} = 3$ then the set of possible delivery patterns of product j is $R_{ij} = \{(1, 5, 9); (2, 6, 10); (3, 7, 11); (4, 8, 12)\}$ and for instance choosing the pattern $(1, 5, 9)$ means that product j is delivered to customer i on weeks 1, 5 and 9. A set $\mathcal{M} = \{1, \ldots, m\}$ of m heterogeneous vehicles are available, and each vehicle k has a capacity C_k, $k \in \mathcal{M}$, and the same maximum service duration T, where the service duration includes the total service times and the total travel times. The aim is to provide a visiting schedule for each customer and a set of routes for each day of the planning horizon. This is achieved by minimizing the number of vehicles used and minimizing the total travel distance. An illustrative example is given in the following section.

4.2 Illustrative example

To illustrate the problem, let consider a PVRP with a simplified planing horizon of 6 weeks ($W = 6$) where each week consists of one day delivery period. As shown in Figure 1, there are two available vehicles to handle demands of six customers for two products P_1 and P_2. For each product, visit patterns for clients are given in Table 2. In this example, vehicle one is responsible for delivering demands of customers 1, 2 and 3, while vehicle two, is in charge of customers 4, 5 and 6. Figure 1 illustrates a schematic of a transportation plan when choosing patterns are represented in bold in Table 2. Figure 1 shows that agent consistency is respected as each customer is visited by the same agent when a visit is scheduled.

Table 2 – Illustrative example of visiting patterns of customers

customers	Patterns of P_1	Patterns of P_2
1	{(1,4);**(2,5)**;(3,6)}	∅
2	{**(1,3,5)**;(2,4,6)}	{**(1,4)**;(2,5);(3,6)}
3	∅	{**(1;2;3;4;5;6)**}
4	{**(1;2;3;4;5;6)**}	∅
5	{(1,4);**(2,5)**;(3,6)}	{(1,3,5); **(2,4,6)**}
6	∅	{**(1,3,5)**;(2,4,6)}

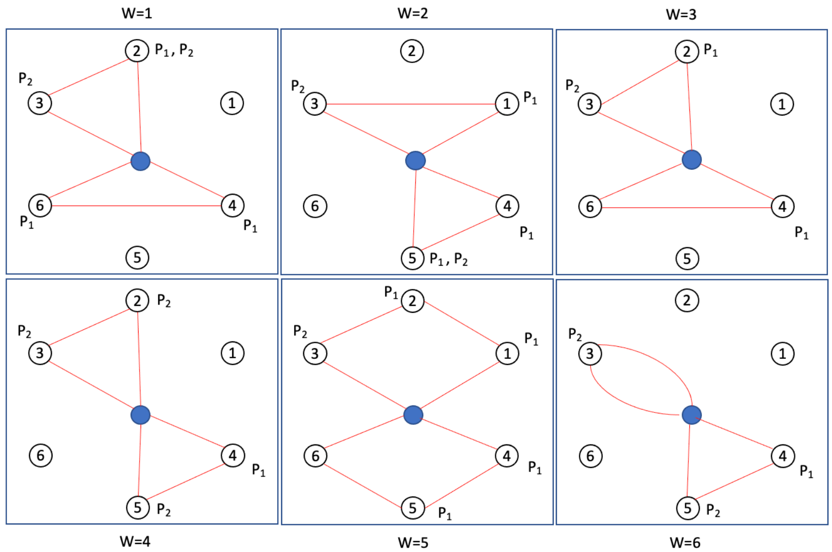

Figure 1 – Illustrative example of transportation plan corresponding to patterns in bold in Table 2.

4.3 Mathematical formulation

This section formulates the problem using 5 sets of variables as follows: the binary variable z_{ijr} which is equal to 1 if pattern r of set R_{ij} is selected, 0 otherwise; the binary variable v_{iw} specifies whether or not customer i is visited during week w of the planning horizon; the binary variable y_{ikw} specifies whether or not the customer i is visited by vehicle (agent) k during week w; the binary variable h_{ikdw} specifies whether or not the customer i is visited by vehicle (agent) k during day d of week w; the binary variable x_{ijkdw} specifies whether or not the customer j is visited after customer i by vehicle k during day d of week w, the binary variable u_{kdw} specifies whether or not the vehicle k is used during the day d of week w. Continuous variables l_{idw} denote the arrival time at customer i on day d of week w. Finally, we use the parameter a^{ij}_{wr} which is equal to 1 if week w is within pattern r of set R_{ij}.

$$\text{Minimize} \quad \sum_{i\in V}\sum_{j\in V}\sum_{k\in\mathcal{K}}\sum_{d\in\mathcal{D}}\sum_{w\in\mathcal{W}} t_{ij}x_{ijkdw} + \sum_{k\in\mathcal{K}}\sum_{d\in\mathcal{D}}\sum_{w\in\mathcal{W}} u_{kdw} \tag{1}$$

Subject to

$$\sum_{r\in R_{ij}} z_{ijr} = 1 \qquad \forall i \in V\setminus\{0\}, \quad \forall j \in P_i \tag{2}$$

$$a^{ij}_{wr} z_{ijr} \leq v_{iw} \qquad \forall i \in V\setminus\{0\}, \quad \forall j \in P_i, \quad \forall w \in \mathcal{W} \tag{3}$$

$$\sum_{j\in P_i} a^{ij}_{wr} z_{ijr} \geq v_{iw} \qquad \forall i \in V\setminus\{0\}, \quad \forall w \in \mathcal{W} \tag{4}$$

$$\sum_{k\in\mathcal{M}} y_{ikw} = v_{iw} \qquad \forall i \in V\setminus\{0\}, \quad \forall w \in \mathcal{W} \tag{5}$$

$$y_{0kw} = 1 \qquad \forall k \in \mathcal{M}, \quad \forall w \in \mathcal{W} \tag{6}$$

$$h_{ikdw} \leq u_{kdw} \qquad \forall i \in V \quad \forall k \in \mathcal{M}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \tag{7}$$

$$\sum_{d \in \mathcal{M}} h_{ikdw} = y_{ikw} \qquad \forall i \in V \setminus \{0\}, \quad \forall k \in \mathcal{M}, \quad \forall w \in \mathcal{W} \tag{8}$$

$$\sum_{j \in V} x_{ijkdw} = \sum_{j \in V} x_{jikdw} = h_{ikdw} \qquad \forall i \in V \setminus \{0\}, \quad \forall k \in \mathcal{M}, \quad \forall d \in \mathcal{D}, \quad \forall w \mathcal{W} \tag{9}$$

$$\sum_{i \in V} \sum_{j \in V} t_{ij} x_{ijkdw} + \sum_{i \in V} s_i h_{ikdw} \leq T \qquad \forall k \in \mathcal{M}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \tag{10}$$

$$\sum_{i \in S} \sum_{j \in S} x_{ijkdw} \leq |S| - 1 \qquad \forall k \in \mathcal{M}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \quad \forall S \subseteq N, \ |S| \geq 2 \tag{11}$$

$$\sum_{i \in V} \sum_{j \in P_i} a_{rw}^{ij} q_{ij} h_{ikdw} \leq Q_k \qquad \forall k \in \mathcal{M}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \tag{12}$$

$$M(h_{ikdw} - 1) + h_{ikdw} + \sum_{u=1:u \neq w}^{W} \sum_{l=1:l \neq d}^{D} h_{iklu} \leq y_{ikw} \qquad \forall i \in V \setminus \{0\}, \quad \forall k \in \mathcal{M}, \quad \forall w \in \mathcal{W} \tag{13}$$

$$l_{idw} + (s_i + t_{ij}) x_{ijkdw} - (1 - x_{ijkdw}) T \leq l_{jdw} \qquad \forall i, j \in V \setminus \{0\}, \quad \forall k \in \mathcal{M}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \tag{14}$$

$$l_{idw} + (s_i + t_{ij}) x_{ijkdw} + (1 - x_{ijkdw}) T \geq l_{jdw} \qquad \forall i, j \in V \setminus \{0\}, \quad \forall k \in \mathcal{M}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \tag{15}$$

$$l_{idw} + s_i + t_{i0} \leq T \qquad \forall i \in V \setminus \{0\}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \tag{16}$$

$$e_i \leq l_{idw} \leq l_i \qquad \forall i \in V \setminus \{0\}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \tag{17}$$

$$x_{ijkdw} \in \{0, 1\} \qquad \forall i, j \in V \quad \forall k \in \mathcal{K}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \tag{18}$$

$$h_{ikdw} \in \{0, 1\} \qquad \forall i \in V \quad \forall k \in \mathcal{K}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \tag{19}$$

$$y_{ikw} \in \{0, 1\} \qquad \forall i \in V \quad \forall k \in \mathcal{K}, \quad \forall w \in \mathcal{W} \tag{20}$$

$$u_{kdw} \in \{0, 1\} \qquad \forall k \in \mathcal{K}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \tag{21}$$

$$v_{iw} \in \{0, 1\} \qquad \forall i \in V \quad \forall w \in \mathcal{W} \tag{22}$$

$$z_{ijr} \in \{0, 1\} \qquad \forall i \in V \quad \forall j \in P_i \quad \forall r \in R_{ij} \tag{23}$$

$$l_{idw} \geq 0 \qquad \forall i \in V \setminus \{0\}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \tag{24}$$

The objective function (1) minimizes the total travel distance and the number of vehicles used. Constraints (2) make sure that exactly one pattern is selected for every product of customers. Constraints (3) ensure that a customer is visited on weeks of the selected pattern. Constraints (4) ensure that a customer is not visited on weeks that are not in the selected pattern. Constraints (5) guarantee that each customer is serviced by exactly one vehicle on each week he/she requires service. Constraints (6) ensure that each route starts from the depot. Constraints (7) ensure that each customer is serviced by a used vehicle. Constraints (8) guarantee that each customer is serviced only on one day of a week he/she requires service. Constraints (9) are flow conservation constraints. Constraints (10) keep the number of driving hours for every vehicle the daily restrictions. Constraints (11) are subtour elimination constraints. Constraints (12) limit the vehicle capacity. Constraints (13) ensure the driver and the day consistency. Constraints (14) and (15) set the arrival times at the customers. Constraints (16) enforce that vehicles return to the depot in time. Constraints (17) ensure the Time window feasibility. Constraints (18) - (24) define the domains of the decision variables.

The problem is NP hard as it is a generalization of the classical VRP problem. One way forward is to tackle this challenging problem using a decomposition based approach that combines both mathematical models as well as metaheuristics as will be shown in the rest of the paper.

5 Solution methods

Since the considered routing problem is NP-hard as it is an extension of the VRP, its resolution is made even more complex given the large size of the real application, which often exceeds 6,000 customers. In other words, it is not suitable to solve the problem optimally with an exact method. In this study, we adopt a decomposition method which is a practical method for the PVRP problem, widely used in the literature [27], [15], [33]. Our decomposition method is similar to the one initiated in [38]. The method consists of two phases, namely, the planning phase and the routing phase. In the planning phase, customers are assigned to weeks based on their requested products and their frequencies. Then, customers are assigned to the days of the weeks. The primary goal in the planning phase is to balance the workload of the week's days. The routing phase consists in constructing daily routes over the planning horizon. The aim here is to minimize the number of vehicles used and the total distance travelled by the vehicles while respecting driver consistency constraints.

In [38] a heuristic method based on a day-pattern modeling was developed. This method proceeds in three steps: (i) construct a set of day-patterns where each day-pattern corresponds to the same day of the week over the 12 weeks of the planning horizon. For example, the first day-pattern corresponds to the 12 Mondays of the planning horizon. A list of customers is associated to each day-pattern, and for each customer a set of requested products is provided. This set contains all products requested over the 12 corresponding days of the planning horizon, and each product is requested with an estimated demand corresponding to the average value of the quantities requested over the 12 corresponding days of the planning horizon; (ii) generate optimized routes for each day-pattern using a local search, and finally (iii) schedule and adjust the obtained routes over the days of the planning horizon by removing from each route those customers for which visits are not planned.

This section uses the planning optimization phase for the assignment of customers to weeks and days while introducing the following three aspects: (a) the customer assignment over the weeks and days of the planning horizon takes into account the geographical position of the customers, i.e. the clustering of the customers is integrated in the assignment phase, (b) a novel approach for the routing phase based on the LNS methodology is developed and (c), instead of estimating the customer demand with the average quantity, which can be too simplistic and in some cases even misleading, an effective data structure is designed instead. This powerful scheme uses a requests vector which considers, in an intelligent way, the quantities needed by customers for each week in the planning horizon.

5.1 The planning phase

The objective of the planning phase is to assign customers to weeks based on their respective geographical locations, requested products and their frequencies which is then followed by the assignment of customers to the days of the weeks. The goal of this phase is to balance the drivers workload during the week's days and thus reduce the number of vehicles used.

5.1.1 The week planning model

The week planning model focuses on the problem of assigning customers to weeks. The objective is to balance the workload over the W weeks of the planning horizon while satisfying the requests of customers and reducing the number of visits for each customer over the planning horizon.

Given a set of customers V, and a set P of products, each customer i requests a subset of products P_i. Also, for each product $j \in P_i$ there is an associated quantity q_{ij}, a service time s_{ij} and a delivery frequency f_{ij} over the W weeks of the planning horizon. As described in Section 4, the value f_{ij} defines the set S_{ij} of possible delivery scenarios. It is worth noting that choosing a pattern for a product j is equivalent to selecting the week v_1 of the first visit, and the k^{th} visit occurs in week $v_k = v_1 + (k-1)\frac{W}{f_{ij}}$, $k = 1, \ldots, f_{ij}$. Since each customer requests several products and each product has its own frequency of visits, it is therefore convenient to choose appropriate patterns that reduce the number of visits. In other words, we choose the patterns for the products so that the weeks of visits for the different products coincide as much as possible.

Let define $r_{ij} = \frac{W}{f_{ij}}$, $\forall i \in V,\ j \in P_i$, then a solution of the week planning problem consists in selecting, for each product j with frequency f_{ij}, the first visit in the set $\{1, 2, \ldots, r_{ij}\}$. Furthermore, we define the frequency f_i of customer i as the maximum of the frequencies of the products requested by customer i, and the index of the product requested by i with the frequency f_i as g_i. Let $h^i_{jg_i}$ be the greatest common divisor between r_{ig_i} and r_{ij}, $\forall j \in P_i$ and $\forall i \in V$. Finally, we introduce the notation $u[s] \equiv v[t]$ which means that $(u \mod s) = (v \mod t)$ that will be useful in the description of the mathematical model below.

In order to consider the location of customers in the assignment of customers to weeks, we proceed to a partition of the customers into a K clusters set using a fast implementation of k-medoids clustering algorithm ([51]) before assigning scenario's to customers. This clustering algorithm partitions the customers into K clusters with the objective of minimizing the squared distances between the customers in a cluster and its center which is defined as the medoid of that cluster which for simplicity is one of the customer sites. The algorithm starts by choosing K random customer locations, assign the other customers to these centers and compute the sum of the squared distances. The algorithm improves the clustering by considering all possible changes of medoids with non-medoids, which gives $K(n - K)$ candidate for swapping. The best change that reduces the sum of squared distances the most is then chosen. This process is repeated until no further improvements are found. Note that this simple scheme is also well known in the location analysis literature particularly in the p-median problem. The number of clusters K used in our partition will be discussed in the experimentation section. Let u_{il} be a parameter equal to 1 if customer i is in cluster l, $l = 1, \ldots, K$.

In the following, we use a mixed integer linear program (MIP1) to model the weeks planning problem which takes clusters into account when assigning customers to weeks. The binary variable x_{ijw} specifies whether or not the delivery of product j of customer i is performed during week w and the binary variable y_{lw} specifies whether or not the customer of cluster l is visited during week w. The objective is to minimize the sum of clusters visited over the the W weeks.

$$
\begin{aligned}
&\text{(MIP1)} \quad \min && \sum_{l\in\mathcal{K}}\sum_{w\in\mathcal{W}} y_{lw} && \\
&\sum_{i\in V}\sum_{j\in P_i} u_{il}x_{ijw} && \le My_{lw} && \forall l\in\mathcal{K}, \quad \forall w\in\mathcal{W} && (25)\\
&\sum_{i\in V}\sum_{j\in P_i} s_{ij}x_{ijw} && \le L^* && \forall w\in\mathcal{W} && (26)\\
&\sum_{i\in V}\sum_{j\in P_i} q_{ij}x_{i,j,w} && \le C && \forall w\in\mathcal{W} && (27)\\
&\sum_{w=1}^{r_{ij}} x_{ijw} && = 1 && \forall i\in V, \quad \forall j\in P_i && (28)\\
&x_{ijw} && = x_{ij(w+k.r_{ij})} && \forall i\in V, \quad \forall j\in P_i, \quad w=1,\ldots,r_{ij}, && \\
& && && k=1,\ldots,f_{ij}-1 && (29)\\
&x_{ig_iw} && \le \sum_{u\in H_w} x_{iju} && \forall i\in V, \quad \forall j\in P_i, \quad w=1,\ldots,r_{ij}, && \\
& && && H_w=\{u=1,\ldots,r_{ij} : u[h^i_{jg_i}] \equiv w[h^i_{jg_i}]\} && (30)\\
&x_{ijw} && \in\{0,1\} && i\in V, \quad j\in P_i, \quad w=1,\ldots W && (31)
\end{aligned}
$$

Constraints (25) ensure that no customer of a cluster is visited if this cluster is not selected. Constraints (26) limits the total workload of each week to L^*, where the value of L^* is obtained by the mixed integer linear program (MIP2) described below. Constraints (27) ensure that vehicles weekly capacity are not exceeded. Constraints (28) show that exactly one week is chosen for the first visit for each product and each customer and so that the customer is visited according to the frequency of the product in the following weeks as stated in Constraints (29). Constraints (30) restrict the number of visits to each customer, depending on the requested products and their frequencies. Constraints (31) refer to the binary variables. For instance, for a given customer who requests two products, say p_1 and p_2 with frequencies 3 and 2, respectively, over the 12 weeks planning period, the first visit of the customer for product p_1 takes its value in the set $\{1,2,3,4\}$ while the first visit of the customer for product p_2 takes its value in the set $\{1,2,3,4,5,6\}$. Thus, if the MIP chooses week 1 for p_1 then, the first visit for the product p_2 is restricted to the set $H_1=\{1,3,5\}$ which allows to restrict the number of visits to the customer for the products p_1 and p_2, see Figure 2. Note that some combinations of frequencies mean that the number of visits cannot be reduced. For example, if p_1 and p_2 have frequencies 4 and 3, respectively, then the first visit for product p_1 takes its values in the set $\{1,2,3,4\}$ while that of p_2 takes its values in the set $\{1,2,3\}$. If week 1 is chosen for p_1, then set of possible visits for p_2 remains $\{1,2,3\}$, see Figure 3.

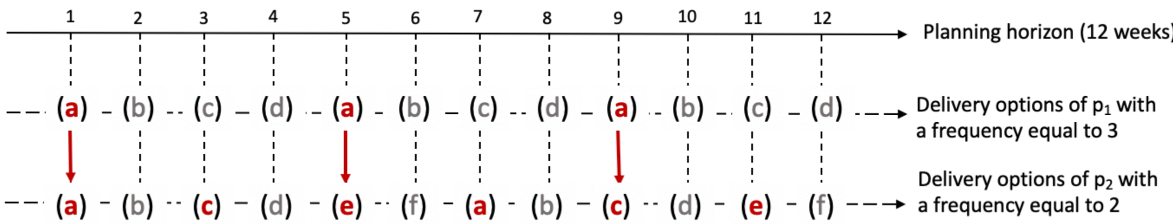

Figure 2 – Example of delivery options for p_1 with a reduction of the possible delivery options for p_2

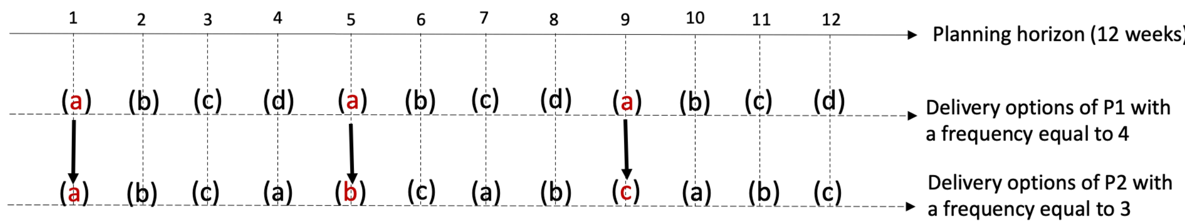

Figure 3 – Example of delivery options for p_1 without a reduction of the possible delivery options for p_2

In order to fix the value of L^* in constraints (26) we solve the same week planning problem described above without considering the locations of customers. More precisely, we solve the following mixed integer linear problem (MIP2).

$$\begin{aligned}
(\text{MIP2}) \quad L^* = \quad & \min \quad L \\
s.t. \quad & \sum_{i \in V} \sum_{j \in P_i} s_{ij} x_{ijw} \leq L \quad \forall w \in \mathcal{W} \qquad (32) \\
& (27) - (31)
\end{aligned}$$

We solve both (MIP1) and (MIP2) problems optimally using the optimization solver. Thus the complete schema of the planning model is given by the Algorithm 1.

Algorithm 1: Week planning approach

Result: assignment of customers to weeks

1 Apply the fast k-medoids clustering algorithm to partition clients into K clusters.
2 Solve the (MIP2) problem and obtain the value of L^*
3 Solve the (MIP1) problem and obtain the assignement of customers to weeks

5.1.2 The days planning model

The second step of the planning phase focuses on the problem of assigning the customers to days of the week. Since the day consistency constraint must be satisfied, this second step ensures that each customer is assigned to the same day of the week over all weeks of the planning horizon once the customer's visits are scheduled. The objective of the days planning model is to balance the workload between days of the week.

Given a set of customer N_w assigned to week w, let $s_{i,w}$ and $Q_{i,w}$ be the service time and the demand quantity of customer i during week w, respectively, obtained by solving the week planning model. Let $N_{w,f}$ be the set of customers assigned to week w having a frequency equals to f. In the following we use a mixed integer linear program to model the days planning problem. The decision variable T represents the maximum service time of all customers during each day of the planning horizon and the binary variable x_{id} specifies whether or not customer i is visited on day d. The objective of the daily planning model is to balance the workload measured in terms working time between the days of the week.

$$\min \quad T$$

$$T \geq \sum_{i \in N_w} s_{iw} x_{id} \qquad \forall w \in \mathcal{W}, \quad \forall d \in \mathcal{D} \tag{33}$$

$$C \geq \sum_{i \in N_w} Q_{iw} x_{id} \qquad \forall w \in \mathcal{W}, \quad \forall d \in \mathcal{D} \tag{34}$$

$$\sum_{d \in D} x_{id} = 1 \qquad \forall i \in C \tag{35}$$

$$\sum_{i \in N_{wf}} x_{id} \leq \left\lceil \frac{|N_{wf}|}{D} \right\rceil \qquad \forall w \in \mathcal{D}, \quad \forall d \in \mathcal{D}, \quad \forall f \tag{36}$$

$$x_{id} \in \{0,1\} \qquad \forall i \in C, \quad \forall d \in \mathcal{D} \tag{37}$$

Constraints (33) restrict the total workload of each day. Constraints (34) guarantee that each vehicle daily capacity C is not exceeded. Constraints (35) ensure that exactly one day is selected for every customer. Constraints (36) balance different frequencies of the customers between days of the week and constraints (37) refer to the binary variables. Note that constraints (36) are not imposed by the industrial case, but they allow balancing the distribution of customers with the same frequency on the days of the week. This is useful for balancing the routes in the second phase of the solution approach.

5.2 The routing phase

After assigning all customers to the days of the planning horizon using the week and the days planning models (Sections 5.1.1 and 5.1.2) respectively, we now solve for each day of the planning horizon a variant of the VRP. More specifically, in addition to the classical VRP constraints such as vehicle capacities, time windows of visits and the limited driving hours per vehicle per day, there is the requirement that a customer visits need to be performed by the same driver (driver consistency). This is an important constraint which imposes that solving the VRP for each day of the planning horizon cannot be performed independently. The objective of the routing step is therefore to build optimized routes for each day of the planning horizon while guaranteeing that customers are visited by the same vehicle. In other words, this problem is equivalent to selecting a subset of customers to be visited by each vehicle throughout the planning horizon, and for each subset, the route must be optimized in terms of distance. The main challenge is to balance the routes. This is achieved by avoiding routes such that for some weeks a given route is overloaded where the number of visits is much higher while in the other weeks the same route is under-loaded requiring a relatively much lower number of visits.

In this section we develop an LNS meta-heuristic to solve the routing problem considering the effective quantities requested by customers for each week in the planning horizon. In fact, as we have the same set of customers for each day of the week over all weeks of the planning horizon, we will focus here on solving D vehicle routing problems where each one is related to the same day of the week over the W weeks of the planning horizon. In the following, we present our method on a given day d of the week where VRP(d) refers to solving the VRP on day d over W weeks of the planning horizon. Note that the method remains valid on the other days, since each customer is visited at most once a week.

For each individual problem such as the VRP(d), a heterogeneous fleet of vehicles is available and each vehicle k has a vector capacity $(C_k, C_k, \ldots, C_k)$ of size W where the l^{th} component of the vector corresponds the l^{th} week of the planning horizon. Let S_d be a set of customers to be served on day d over W weeks of the planning horizon. For each customer i, $i \in S_d$, we define a vector of requested products-quantities $\tilde{P}_{d,i}$ with $\tilde{P}_{d,i} = (\tilde{P}^1_{d,i}, \ldots, \tilde{P}^W_{d,i})$, where $\tilde{P}^l_{d,i}$ is a list of couples $(p_{d,i}; q_{d,i})$ of all products and quantities requested by customer i during week l. When a product is not required during a given week l we set its requested quantity to zero and when a customer is not visited in that week we also set its quantities of its requested products to zero. Thus in the VRP(d) problem, a route is feasible if the time window of each customer is satisfied and the quantities requested by customers do not exceed the capacity of the vehicles. In other words, for a given route R we have

$$\sum_{i \in R} \sum_{p_{di} \in \tilde{P}^l_{di}} q_{di} \leq C_k, \qquad l = 1, \ldots, W. \tag{38}$$

The VRP(d) problem can be seen as a classic VRP problem with vectors of customer demands. To solve this problem we develop a LNS method where each customer is represented by a vector of demands. Such a method provides us with master-routes that are feasible for redday d of each week of the planning horizon. This is then followed by a post-optimization for each week, where customers with zero demands are then removed from each master-route, and the master-route is re-optimized again as a TSP problem. Our LNS method constructs master-routes of VRP(d) with the objective of minimizing the number of used vehicles and the total distance traveled. The classical LNS algorithm is an iterative process where, at each iteration, part of the current solution is destroyed and then reconstructed in order to find a better solution. The destruction step consists of removing some nodes from the existing routes using removal operators to make up the *unassigned set*. Then the construction step, also known as the building or the repair step, inserts the nodes from the *unassigned set* into the routes of the partial solution using the repair operators. From a set of destruction and repair operators, a random selection of these operators is applied in each phase based on a wheel selection mechanism. This succession of destroy and repair steps is carried out within a simulated annealing (SA) framework to manage the acceptance of the new solutions. An overview of the proposed LNS approach is described in Algorithm 2. For more information on heuristics and metaheuristics in general and for LNS in particular see [50].

Algorithm 2: LNS outline

Result: best feasible solution s^b

initialization: $s^b \leftarrow s$,

while stop criterion is not met **do**

  $selectOperators(\Omega^-, \Omega^+)$;

  $s^t \leftarrow repair(destroy(s))$;

  if $accept(s^t, s)$ **then**

    $s^b \leftarrow s^t$;

  end

  if $obj(s^t) < obj(s^b)$ **then**

    $s^b \leftarrow s^t$

  end

end

In the following we describe the construction and destruction operators used in our method.

a. Destruction operators. These operators aim to disconnect q nodes from the current solution, with

$q \in [n \cdot \xi_{\min}, n \cdot \xi_{\max}]$, where n is the number of nodes in the current solution, and $\xi_{\min}$ and $\xi_{\max}$ are parameters. We adapt a removal strategy that combines randomness and some form of guidance. In our study, a Random-Removal, a Cluster-Removal, and a Route-Removal operators are used.

- *Random-Removal.* Select q nodes randomly and remove them from the current solution.

- *Cluster-Removal.* Here, a large set of the closest nodes in terms of distance is disconnected instead. The cluster removal process which was introduced in [45] begins with a random selection of a route, followed by a clustering step. The latter step consists in partitioning the nodes of the current route into two clusters using a modified Kruskal's algorithm for the minimum spanning tree problem. One of the two clusters is randomly selected, its nodes disconnected and then added to the unassigned set. The process is repeated until q nodes are removed.

- *Routes-Removal.* Aims to eliminate one or more complete routes. This avoids the multi-step setting of upper bounds for the number of vehicles used as adopted in some LNS implementations that minimize the number of vehicles (e.g. [44]). This operator first selects a route randomly, then removes all nodes contained in the route and adds them to the list of nodes to be inserted. The procedure is repeated until at least q nodes are deleted.

b. Construction operators. These operators aim to insert into the current solution the nodes that either have been deleted and placed in an unassigned set or could not be inserted in the previous solution. We refer this set of nodes to R. In this study, we use both the Best-Insertion and the Regret-Insertion operators.

- *Best-Insertion.* From the set R of nodes, the one with the lowest insertion cost considering all possible insertion is performed. The process is repeated till R is empty or none nodes of R can be inserted.

- *Regret-Insertion.* At each iteration, a node i of R with the highest value $\sum_{l=1}^{k}(f_{i,l} - f_{i,1})$ is selected and inserted in its best position, where $f_{i,l}$ is the insertion cost of node i in its l^{th} best position over all routes.

At each iteration of the LNS method, the quality of the solution is evaluated using three metrics: (i) the total distance traveled by all vehicles in the current solution, (ii) the number of vehicles required and (iii) the number of unassigned customers. These three metrics are weighted by coefficients α, β and γ respectively.

The master-routes constructed by the LNS method allow the same customers to be served by the same vehicle. However, when scheduling the master-routes on the days of the planning horizon, some customers in these routes may not necessarily be scheduled for a visit in certain weeks. In this case, those unplanned customers are removed from these master-routes resulting in re-optimizing these routes again. Here, the intra-route re-optimization problem reduces to the TSP problem [6]. In this step, for simplicity, we use the Lin-Kernighan heuristic ([32]) though other more powerful heuristics could also be adopted instead.

6 Experimental results

The proposed algorithms are implemented in Java and the mathematical models of the first and the second stages of our approach are solved using CPLEX 12.8. In the following, we present our experimental results on (i) the industrial instances, then followed by (ii) benchmarks from the literature on instances that are closely related to the current logistical problem.

6.1 Industrial instance

In the following, detailed characteristics of the industrial instance are provided and experimental results are analysed. The results of our computational experiments are discussed in terms of the objective set at each step of our approach, while the final result is compared against the solution already used by the company. Given that the industrial application does not impose limitations on the calculation time used, for consistency the computations time are therefore not reported in this experiment. However, the running times of the different phases of our approach are given for information only.

6.1.1 Instance description

The company covers 6062 customers in the Ile-de-France region (France). The planning horizon is 12 weeks, with 5 working days in each week. The company provides 14 types of products, and these products are requested with frequencies 1, 2, 3, 6 and 12. Each customer requests a subset of products and the frequency of delivery for each product over the planning horizon. In total, there are 69951 product-customers-frequencies requests which corresponds on average to 11.54 requests per customer. The service times of customers is in the range $[2, 328]$ minutes. A summary of the distribution of customers by the number of requested products and by the maximum frequency of visits are given in Tables 3 and 4, respectively.

The company uses 40 vehicles of 12 different capacity types (payloads), ranging from 500 kg to 1600 kg. The maximum driving and servicing time is limited to 7 hours and 30 minutes per day for each driver. Note that in the initial experiment conducted in Messaoudi et al. [38], the maximum routing time was set to 7 hours for each vehicle (agent). However, the reorganization of the company's internal logistics allowed 30 additional minutes of working shift to be used in the routing phase. Therefore, for consistency we will also compare the additional gain obtained by the extension of the maximum route duration.

Table 3 – Distribution of customers by the number of requested products

Products	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Customers	1400	545	1235	1103	1109	328	182	79	43	25	9	1	3	0

Table 4 – Distribution of customers by their maximum requested frequencies

Frequency	1	2	3	6	12
Customers	247	369	3403	1259	784

6.1.2 Week planning results

The mixed integer linear programs of the week planning model defined in Algorithm 1 (Section 5.1.1) are optimally solved in a very short time, less than 2 minutes. MIP1 is solved with the values of K in the set $A = \{60, 84, 120\}$. These values are chosen such that on average there are between 1 and 2 clusters per day during the 12 weeks. The results of the three values of K are presented in Table 5. For completeness, we also provide at the end of this table the results of the week planning without clustering which are originally produced by [38]. Besides, we also report in Table 5, for each week of the time horizon, the total service time of customers in hours (TST), the number of customers assigned to that week (Nb-C) and the maximum travel distance between two customers (MD) in kilometres. The results clearly show a well balanced total service time between the weeks for all values of K, with a slight advantage for $K = 60$ followed by $K = 120$. For each value of K, the maximum difference between the busiest week and the lightest week are found to be 10 minutes for CL-60, 15 minutes for CL-120, 18 minutes for CL-120 and 26 minutes for Messaoudi et al. [38]. However, we can also observe that the number of customers is less balanced between weeks. Table 6 provides statistical metrics on the distribution of customers over the weeks. It can also be noted that the standard deviation of the average number of customers per week is smallest for the case CL-60 with std.dev = 48.32, which shows a balanced distribution of customers over all weeks. This is followed by CL-120 with std.dev

= 69.40. In addition, it is worth mentioning that the effect of the customer clustering approach is significant as it clearly improves the distribution of customers over the weeks compared to the results of [38] where the standard deviation value reaches 445.27. Besides, the average value of the maximum distance between customers (DM) is also found to be smallest in the CL-60 case.

Table 5 – Results of the weeks planning model

Week	1	2	3	4	5	6	7	8	9	10	11	12
Results obtained with K = 60 clusters (CL-60)												
TST	547.93	548.03	547.98	547.92	548.02	548.00	547.90	547.87	548.05	547.92	547.90	548.03
Nb-C.	2283	2341	2393	2415	2278	2325	2367	2417	2295	2331	2373	2404
MD	81.17	76.17	81.04	76.17	72.76	76.17	71.17	76.17	82.76	72.17	81.04	71.17
Results obtained with K = 84 clusters (CL-84)												
TST	547.95	548.07	547.98	548.00	548.03	548.05	548.00	547.95	547.80	548.03	548.10	547.98
Nb-C.	2471	2221	2495	2213	2474	2211	2511	2239	2457	2181	2515	2237
MD	83.64	87.08	82.61	76.15	84.75	87.08	83.30	86.86	83.64	87.08	84.75	76.15
Results obtained with K = 120 clusters (CL-120)												
TST	548	547.87	548.12	547.95	548.05	547.93	548.07	547.95	547.93	547.97	548.05	548.07
Nb-C.	2389	2242	2415	2367	2399	2232	2423	2365	2365	2246	2435	2347
MD	86.49	75.62	87.21	76.00	86.49	75.62	86.79	76.00	87.21	75.62	86.79	76.00
Results obtained in Messaoudi et al. [38]												
TST	547.72	548.13	547.98	548.15	547.83	547.92	547.97	547.98	548.08	548.12	548.08	548.08
Nb-C.	3120	2308	2137	1987	3043	2240	2171	1839	3051	2386	2123	1823
MD	86.76	86.90	86.10	86.91	86.76	86.90	86.10	82.59	86.76	87.04	87.76	82.69

TST. total service time (hours), Nb-C. number of customers, MD. Maximum distance between clients (km)

Table 6 – Statistical data on the distribution of clients in the week planning model

		min	max	average	std. deviation
CL-60	TST	547.87	548.05	547.96	0.06
	Nb-C	2278	2417	2351.83	48.32
	DM	71.17	82.76	76.50	4
CL-84	TST	547.80	548.10	548	0.07
	Nb-C	2181	2515	2352.08	136.62
	DM	76.15	87.08	83.59	3.67
CL-120	TST	547.87	548.12	548	0.07
	Nb-C	2232	2435	2352.08	69.40
	DM	75.62	87.21	81.32	5.52
Messaoudi et al. [38]	TST	547.72	548.15	548	0.13
	Nb-C	1823	3120	2352.33	445.27
	DM	82.59	87.76	86.11	1.60

6.1.3 Days planning results

The MILP of the days planning model is solved with a gap of 0.2% in 3 minutes. Table 7 reports, for each value of K and each week of the planing horizon, the minimum (min-ST) and the maximum (max-ST) service time in hours over the five days, with the results obtained in [38] also given at the bottom of this table for completeness. We observe a perfect balance of service time between all days of the week, regardless of the value of K, and these results are similar to those of [38]. Table 8 summarizes the distribution of the number of clients over the days of the week. We observe that the clustering of the customers, regardless of the value of K, gives a balanced distribution of the customers on the days of the weeks. We can see a clear benefit of the clustering with $K = 60$ (CL-60) with the lowest standard deviation, followed by $K = 120$ (CL-120). The clustering of the customers largely dominates the approach of [38] which does not consider the effect of clustering. Thus, for the routing phase we will use the CL-60 clustering days planning results only.

Table 7 – Service time distribution in days planning model

Weeks	1	2	3	4	5	6	7	8	9	10	11	12
Results obtained with 60 clusters												
min-ST	109.40	109.50	109.32	109.40	109.47	109.37	109.42	109.17	109.35	109.35	109.47	109.47
max-ST	109.75	109.70	109.75	109.77	109.75	109.80	109.78	109.77	109.75	109.73	109.73	109.75
Results obtained with 84 clusters												
min-ST	109.33	109.68	109.08	109.23	109.47	109.75	109.10	109.45	109.27	109.38	109.40	109.67
max-ST	109.77	109.98	109.88	109.87	109.72	109.97	109.85	109.77	109.80	109.77	109.78	109.95
Results obtained with 120 clusters												
min-ST	109.43	109.40	109.57	109.18	109.43	109.42	109.43	109.47	109.47	109.35	109.52	109.42
max-ST	109.70	109.67	109.67	109.73	109.73	109.73	109.73	109.72	109.75	109.73	109.73	109.75
Results obtained in Messaoudi et al. [38]												
min-ST	109.27	109.35	109.13	109.38	109.23	109.18	109.53	109.08	109.42	109.45	109.48	109.35
max-ST	109.72	109.77	109.78	109.85	109.82	109.78	109.57	109.87	109.87	109.77	109.75	109.75

Table 8 – Statistical parameters on the distribution of clients in the days planning model

parameters	days	min nb of cust.	max nb of cust.	average	std deviation
CL-60	Monday	456	482	469.83	9.25
	Tuesday	453	482	469.50	9.68
	Wednesday	456	484	470.08	9.47
	Thursday	456	485	471.42	9.91
	Friday	454	486	471.00	10.61
CL-84	Monday	435	504	469.58	27.75
	Tuesday	431	503	469.00	28.59
	Wednesday	439	504	471.42	26.58
	Thursday	438	503	470.67	26.27
	Friday	438	505	471.42	27.60
CL-120	Monday	444	486	470.58	14.26
	Tuesday	446	488	471.42	14.57
	Wednesday	447	488	470.75	13.67
	Thursday	447	487	469.92	13.50
	Friday	444	487	469.42	13.72
Messaoudi et.al.	Monday	367	625	471.50	89.01
	Tuesday	365	621	469.92	88.50
	Wednesday	362	623	470.00	88.75
	Thursday	365	626	470.25	89.22
	Friday	363	625	470.67	89.83

6.1.4 Routing results

In this phase we use the LNS algorithm developed in Section 5.2 to solve the VRP problem for each day of the week. Some preliminary experiments are conducted to calibrate the values of the parameters of our LNS. The number of nodes to be removed in each iteration is randomly chosen in the interval $[\xi_{\min}, \xi_{\max}]$, with $\xi_{\min} = min(7, 0.1 \times n)$ and $\xi_{\max} = min(40, 0.25 \times n)$, where n refers to the number of nodes contained in the current solution. The parameters of our simulated annealing include the initial temperature and the cooling schedule [28]. As stated in [44], the initial temperature T_{init} depends strongly on the instance of the problem. In this study, the initial temperature is set to a value such that the new solution is accepted with a probability of 0.5 if the value of its objective function is at most $w\%$ far away from the objective value of the current solution, where w is fixed to 40. The temperature decrease is given as follows: $T_{iter+1} = T_{iter} - c$, where c is fixed to 0.88. Finally, the coefficients of the objective function are set to $(\alpha, \beta, \gamma) = (1, 3000, 120000)$. Time and distance travel matrices are computed using Google Distance Matrix API ([23]). In the following, we present the results of LNS metaheuristic, and we compare that results with results of [38].

Each customer's demand is represented as a product-quantity vector for the 12 weeks of the planning horizon. For each day, we restricted the route duration limit to 450 minutes (7 hours and 30 minutes) per driver as explained in Section 6.1.1. A computing time limit of 2 hours was also set. First, in Table 9 we evaluate the effect of increasing the maximum duration of the routes from 420 minutes (7 hours) to 450 minutes (7 hours and 30 minutes) in the resolution of the problem with the pattern method of [38]. It is good to note that the number of vehicles used has not changed, there are still 35 vehicles used in both cases. However, in terms of the number of kilometers travelled, according to Table 9, there is a massive decrease of 10.3% (i.e., 13,136.3 kms) due to the extra flexibility in allowing a small increase of 30 minutes. This demonstrates that time flexibility in routing can be paramount and ought not to be overlooked.

Table 9 – Total distances obtained for real instance over 12 weeks with pattern method [38]

	Weeks												
	1	2	3	4	5	6	7	8	9	10	11	12	total
Km (420 min)	11235.9	10722	10391.7	10333.5	11166.8	10734.5	10477.8	9965.3	11129.2	10994	10393.6	10026.3	127,570.6
Km (450 min)	11051.6	10371.1	10253	10086.7	10988.3	10371.4	10320.5	9774.1	10990.3	10547.6	10255.1	9796	114,434.3

In the following section we compare the pattern method [38] with the vector method with a maximum route duration of 450 minutes.

Comparison of Vector vs Pattern - Using the vector method, the number of vehicles used is 24, 23, 25, 24 and 25 for Monday, Tuesday, Wednesday, Thursday and Friday, respectively, while the pattern method uses 35 vehicles for each day of the week [38]. The results empirically demonstrate that this innovative idea of modeling customer demand as a vector can drastically reduce the number of vehicles used where the reduction varies between 10 and 12 vehicles.

Tables 10 reports the total distance obtained for each day of each week alongside the results provided by the pattern method of [38]. It was found that the average distances travelled, shown in bold, are reduced by nearly 16% ((2181.9 − 1834.5)/2181.9) for Monday and up to over 18% for Wednesday. Table 11 shows the average routing duration obtained for each day of each week, and comparison with [38] is also provided. We can see that the average service duration of the agents are very close to the total service limit per agent. This shows that resources are better exploited while respecting operational constraints. Figure 4 provides an illustrative comparison of the distribution of the duration of the routes between the results of [38] and those found by the proposed LNS method using box plots for the first day of the week. A similar pattern to the one shown in Figure 4 is observed for the rest four days of the week.

Table 10 – Total distances obtained for real instances over 12 weeks (in kilometres)

	Monday		Tuesday		Wednesday		Thursday		Friday	
	Vector	Pattern	Vector	Pattern	Vector	Pattern	Vector	Pattern	Vector	Pattern
Week 1	1834.5	2181.9	1731.7	2087.0	1893.3	2319.9	1848.4	2258.1	1921.8	2204.7
Week 2	1813.5	2090.0	1626.4	2031.0	1837.9	2143.8	1881.1	2127.5	1680.0	1978.8
Week 3	1738.9	2058.7	1610.0	1990.4	1661.9	2136.9	1555.9	1984.3	1684.4	2082.7
Week 4	1846.6	2069.4	1695.8	2041.7	1670.7	2011.6	1678.1	2025.6	1585.1	1938.4
Week 5	1824.3	2182.5	1725.3	2083.1	1810.7	2295.3	1792.1	2227.9	1844.5	2199.5
Week 6	1794.9	2057.3	1620.6	2013.7	1829.5	2209.5	1863.2	2113.7	1683.7	1977.2
Week 7	1807.3	2101.7	1612.5	1992.5	1735.1	2141.7	1564.4	1990.8	1767.0	2093.8
Week 8	1546.8	1941.8	1508.2	1991.1	1611.2	1964.6	1476.4	1911.7	1647.5	1964.9
Week 9	1832.1	2181.0	1731.5	2084.5	1809.1	2296.5	1795.6	2229.2	1844.2	2199.1
Week 10	1895.5	2149.7	1858.9	2090.5	1831.3	2153.4	1894.4	2145.3	1703.9	2008.7
Week 11	1794.1	2097.8	1616.1	1993.1	1654.3	2109.7	1551.7	1962.0	1685.6	2092.5
Week 12	1557.0	1940.0	1506.7	1993.0	1622.8	2033.3	1525.7	1932.3	1540.1	1897.4
Average	**1773.8**	2087.7	**1653.6**	2032.6	**1747.3**	2151.3	**1702.2**	2075.7	**1715.6**	2053.2

Vector: The proposed LNS method, Pattern: Method of [38]

Table 11 – Average routing durations obtained for real instances over 12 weeks (in hours)

	Monday		Tuesday		Wednesday		Thursday		Friday	
	Vector	Pattern	Vector	Pattern	Vector	Pattern	Vector	Pattern	Vector	Pattern
Week 1	6.95	5.37	7.02	5.72	6.80	5.58	6.82	5.43	6.91	5.67
Week 2	6.87	5.28	6.86	5.37	6.70	5.25	6.85	5.30	6.81	5.27
Week 3	6.74	5.36	6.65	5.12	6.38	4.98	6.55	4.91	6.55	5.38
Week 4	6.56	4.82	6.70	4.99	6.45	5.01	6.78	5.10	6.40	5.04
Week 5	6.93	5.37	7.00	5.64	6.77	5.45	6.62	5.29	6.76	5.59
Week 6	6.88	5.06	6.83	5.33	6.64	5.40	6.58	5.08	6.86	5.20
Week 7	6.75	5.31	6.80	5.17	6.49	5.12	6.56	4.88	6.51	5.30
Week 8	6.53	4.82	6.72	4.95	6.30	4.96	6.70	4.97	6.41	5.07
Week 9	6.89	5.45	6.98	5.52	6.77	5.65	6.87	5.44	6.77	5.56
Week 10	6.88	5.25	7.09	5.43	6.75	5.35	6.83	5.34	6.86	5.31
Week 11	6.63	5.31	6.80	5.21	6.40	5.08	6.47	4.75	6.51	5.24
Week 12	6.54	4.85	6.48	4.99	6.28	4.74	6.68	4.97	6.35	5.07
Average	6.76	5.19	6.83	5.29	6.56	5.21	6.69	5.12	6.64	5.31
Average % of agent service time	**90.13**	69.20	**87.46**	70.53	**87.46**	69.46	**89.20**	68.26	**88.53**	70.80

Vector: The proposed LNS method, Pattern: Method of [38]

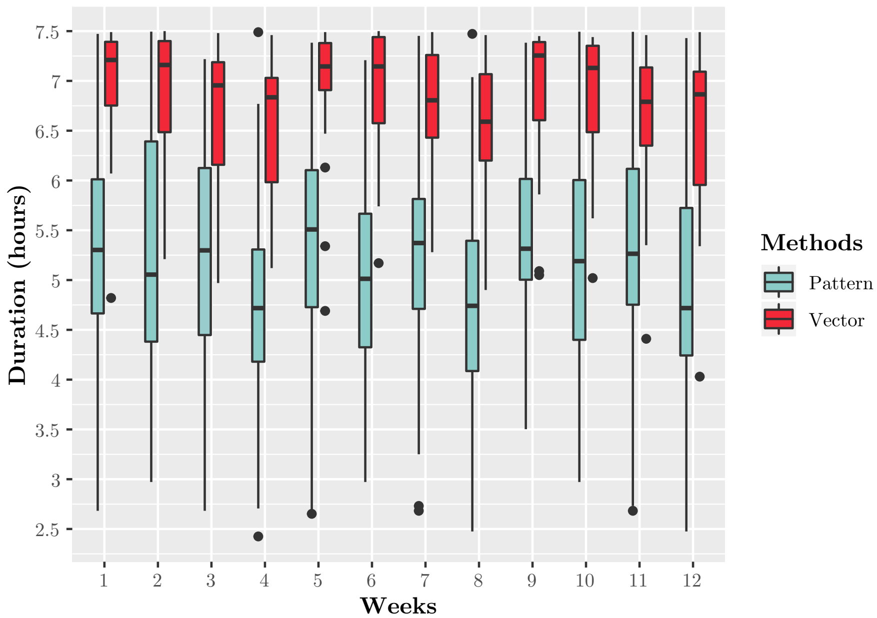

Figure 4 – Distribution of route durations with the Vector and Pattern methods for Monday

6.2 State-of-the-art results on related problems

In this section, we assess the performance of our approach by benchmarking our results against the best known results from the literature on those similar problems (instances) to ours. We have chosen the two recent works, one by [22] and the other by Rodríguez-Martín et al. [42] where the best known results are also reported.

6.2.1 Comparison with [22]

In [22], the periodic vehicle routing problem assumes that the visit schedule of each customer is given and the objective is to minimize the total vehicle operating time over the time horizon while satisfying the driver and the arrival-time consistency constraints. Arrival-time consistency refers to customers being served at roughly the same time on every day of the planning horizon on which the service is required. This is measured as a maximum allowed time difference L between the latest and the earliest arrival time at each customer over the planning horizon. In the experiments below, we compare our results when $L = +\infty$ referring to the arrival-time consistency being relaxed. Furthermore, only the routes construction phase of our approach is used for comparison since the subset of days on which each customer must be served is given in their paper. Three sets A, B and C of instances are used. The set A contains five instances with 10 customers each and five instances with 12 customers each. Here, a planning horizon of three days is used. The set B contains 12 instances with 50 to 199 customers and a planning horizon of five days. The set C is an extension of the set B in which different values of L and frequency of visits are generated to obtain a total of 144 instances. Since the set A contains only small instances, and the set C is an extension of the set B, we focus in presenting the results on instances of the set C with value $L = +\infty$. To be consistent with [22], we have also adapted our LNS method to the same objective function to be minimized, and we have set the number of iterations to 25000. We kept the other parameters as in their work. Table 12 reports the instance name (Instance), the best known solution (BKS) from [22], the average solution value of the 10 runs (Avg), the best solution value of the 10 runs (Best), average computing time (t) in seconds, and the gap of the best solution value to the BKS.

Although the routes construction of our approach has not been built specifically to cater for the problem in [22], our results are found to be significant. Indeed, the average deviation from the BKS is between 0.32% and 0.66% only, and our worst result is recorded as 1.88%. In addition, our approach discovers new best results on three instances out of the 36 (this refers to 12 for each of the 3 frequencies). One of the solutions is related to a frequency of 0.5 and the other two are found when the frequency is 0.9. These best results are shown in bold.

6.2.2 Comparison with [42]

In [42], Rodríguez-Martín et al. consider the periodic VRP with driver consistency constraint and additional bounds on the minimal and the maximal number of customers in each route. Also, each customer has an associated set of allowable visit schedules. The objective is to design a set of minimum cost routes that service all customers while respecting their visit requirements, the driver consistency, and the bounds on the number of customers in each route. The authors use a branch-and-cut algorithm to solve the problem. They generate 240 new instances with 10 to 70 customers. They also the number of vehicles to be between 2 and 4, and a planning horizon of 2 to 5 days.

In order to test our approach against theirs, we have adapted the day planning phase to take into account the specific constraints of [42]. As they do not consider the service time and the customers demand, we set the service time to 0 and the demand for all customers to 1. The aim is to balance the total number of customers visits over the days. In the MILP of the days planning model we retain the binary variable $x_{i,d}$ which is equal to 1 when customer i is visited on day d. We also add new decision variables where $y_{i,p}$ equals to 1 when pattern p of customer i is selected. Let p_d be a parameter equal

Table 12 – Consistent VRP results: comparison with [22]

Service frequency	Instance	$\lvert C \rvert$	BKS	Avg.	Best	t (seconds)	Gap (%)
0.5	1_5_0.5	50	1616.37	1657.86	1646.77	48.44	1.88
	2_5_0.5	75	2554.83	2610.63	2584.19	64.37	1.15
	3_5_0.5	100	2632.43	2736.43	2651.82	220.90	0.74
	4_5_0.5	150	3317.49	3385.18	3332.86	450.57	0.46
	5_5_0.5	199	3986.56	4095.72	4017.07	548.54	0.77
	6_5_0.5	50	2863.55	2872.46	2868.98	27.81	0.19
	7_5_0.5	75	4632.31	4668.79	4642.51	44.40	0.22
	8_5_0.5	100	5332.55	5361.56	5336.30	126.78	0.07
	9_5_0.5	150	7347.4	7398.62	7356.00	252.32	0.12
	10_5_0.5	199	9267.06	9347.53	**9262.88**	344.34	**-0.05**
	11_5_0.5	120	3245.08	3378.13	3269.51	517.97	0.75
	12_5_0.5	100	2835.65	2929.18	2881.57	159.96	1.62
Average						233.86	0.66
0.7	1_5_0.7	50	2105.39	2129.74	2117.01	47.67	0.55
	2_5_0.7	75	3481.82	3540.07	3486.68	63.53	0.14
	3_5_0.7	100	3266.77	3322.87	3301.90	217.71	1.08
	4_5_0.7	150	4346.38	4511.06	4408.95	479.03	1.44
	5_5_0.7	199	5464.52	5614.85	5539.88	613.35	1.38
	6_5_0.7	50	4048.96	4064.34	4064.34	24.01	0.38
	7_5_0.7	75	6645.05	6701.97	6652.18	41.90	0.11
	8_5_0.7	100	7092.22	7174.47	7114.69	110.83	0.32
	9_5_0.7	150	10316.71	10399.34	10355.96	259.06	0.38
	10_5_0.7	199	12827.08	12976.35	12920.60	362.16	0.73
	11_5_0.7	120	4443.76	4537.43	4462.58	352.23	0.42
	12_5_0.7	100	3408.55	3471.76	3422.27	178.94	0.40
Average						229.20	0.61
0.9	1_5_0.9	50	2478.84	2507.08	2492.51	39.69	0.55
	2_5_0.9	75	4001.08	4040.55	4005.57	68.31	0.11
	3_5_0.9	100	3974.74	4056.31	4011.96	239.28	0.94
	4_5_0.9	150	4942.23	5058.15	4952.25	465.22	0.20
	5_5_0.9	199	6376.09	6418.95	**6340.84**	668.03	**-0.55**
	6_5_0.9	50	4751.79	4784.07	4767.65	25.45	0.33
	7_5_0.9	75	7705.73	7751.22	7710.09	43.28	0.06
	8_5_0.9	100	8733.72	8784.01	8747.48	127.37	0.16
	9_5_0.9	150	12377.6	12464.26	12428.56	275.15	0.41
	10_5_0.9	199	15820.63	15950.97	15897.26	387.22	0.48
	11_5_0.9	120	4986.96	5012.73	**4980.10**	391.32	**-0.14**
	12_5_0.9	100	4011.73	4080.31	4061.19	179.63	1.23
Average						242.49	0.32

to 1 if day d is included in pattern p. The resulting modified MILP model is as follows.

$$\min \quad K$$

$$\sum_{p \in P_i} y_{i,p} = 1 \qquad \forall i \in C \tag{39}$$

$$x_{i,d} = \sum_{p \in P_i} p_d \cdot y_{i,p} \qquad \forall i \in C; d = 1, \ldots, D \tag{40}$$

$$K \geq \sum_{i \in C} x_{i,d} \qquad d = 1, \ldots, D \tag{41}$$

$$x_{i,d} \in \{0, 1\} \qquad i \in C; d = 1, \ldots, D \tag{42}$$

Constraints (39) ensure that only one pattern is selected for each customer. Constraints (40) define variables x_{id}. Constraints (41) restrict the total number of visited customers for each day to C_d which is minimized in the objective function. Constraints (42) refer to the binary decision variables.

The instances in [42] contain a maximum of 70 customers that are spread over a planning horizon of no more than 5 days. This leads to a non-dense distribution of customers over the days. Thus, balancing the number of customers per day is not entirely efficient in this case. In this situation, it could be more efficient to cluster the customers prior to the MILP to take into account the sparsity of the customers and thus balance the clusters served each day of the planning horizon.

In our LNS, we have integrated the bound on the maximum number of customers per route. However, the bound on the minimum number of customers per route is not added to our LNS since it completely modifies the structure of our model. Here, we have considered the solution to be infeasible when the minimum bound on the number of customers is not satisfied. We set the number of iterations in the LNS algorithm to 50,000 which takes at most 15 minutes in computing time for all instances. This time remains relatively much lower than the time limit of two hours fixed in [42]. The computing time of the MILP model using CPLEX is found to be negligible and hence not recorded here.

This study presents the results for the instances with 60 and 70 customers and are shown in Tables 13 and 14, respectively. These tables report the number of customers ($|C|$), the number of days in the planning horizon ($|D|$), the number of vehicles ($|V|$), the type of instances (Inst), the best known solution (BKS) from [42] and the type of the selected solution (OPT(BKS)) by noting whether the BKS solution is either optimal (O), feasible (F) or there is no feasible solution found (U) in 2 hours of computing time. We also report the average computing time to find BKS (t(BKS)), the average solution value over 10 runs of LNS (Avg), the best value of the 10 runs (Best), and the gap (in %) of the best solution value of LNS to the BKS defined as $gap(\%) = \frac{Best-BKS}{BKS}.100$.

The results obtained are very encouraging. This is particularly important for large instances with

60 and 70 customers, where we discovered feasible solutions for 10 instances for which no feasible solution was initially reported in [42]. In addition, we also obtained better results for 2 instances with 60 and 70 customers respectively. Furthermore, over all instances, our solutions are on average between 3.68% and 6.3% worse only. It is worth noting that the obtained results, especially the worst ones, are strongly related to the results of the MILP of customer assignment to days. This is important as we do not take into account the distances between customers in our MILP, but the balancing of the number of customers between days only.

7 Conclusion and Suggestions

In this paper, we have investigated the design of tactical plans for a challenging periodic routing problem encountered in a company in the cleaning service sector. We modeled the problem as a multiple periods VRP and proposed a decomposition approach based on a hybridisation of integer programming models and the LNS method with the addition of a new feature to respect the consistency of the routes. Our optimisation tool improves on the current implementation of the company by requiring up to 17 fewer vehicles compared to the 40 vehicles used by the company over the planning horizon. This saving represents a massive 40% reduction in the fixed costs which can provide a significant competitive advantage to the company.

From an academic contribution view point, we have developed a new and innovative approach in the LNS. We introduce a vector of demands of customers that takes into account the consistency constraint making the search relatively much more efficient. We have also incorporated the effect of customers clustering into our search. This is performed by using a simple but powerful clustering scheme to reflect the effect of customer locations in the decision. For completeness, we have also provided an additional assessment by modifying our approach accordingly so to suit the two most recent and related studies to ours that treat PVRPs. To our surprise, we discovered three new best results in the study of [24] while identifying 8 new upper bounds which were not found previously in [42]. This is an important and encouraging result given our approach was not originally designed for these PVRP related problems. This demonstrates empirically that our approach is robust and hence practically useful to be relied upon in real life.

The following future research avenues could be worth considering:

(i) Sophisticated optimization techniques for the weeks and days planning models that consider distances between customers could be explored. This can be achieved by adapting column

Table 13 – PVRP with driver consistency results for instances with 60 customers

$\|C\|$	$\|D\|$	$\|V\|$	Inst.	BKS	opt (BKS)	t (BKS)	Avg.	Best	t (sec.)	Gap (%)
60	2	2	a	1183.54	O	152.77	1242.99	1218.36	143.2	2.94
			b	1151.32	O	70.33	1180.76	1177.31	141.64	2.26
			c	1141.72	O	75.6	1196.51	1165.5	145.35	2.08
		3	b	1224.35	O	419.13	1259.41	1252.1	69.9	2.27
			c	1240.17	O	2970.31	1293.99	1265.62	70.31	2.05
		4	a	1348.38	F	7200	1400.94	1396.66	54.87	3.58
			b	1332.15	O	3365.16	1377.03	1374.13	55.49	3.15
			c	1332.19	O	1564.16	1367.19	1364.46	55.07	2.42
	3	2	a	1756.59	O	440.41	1870.66	1866.29	194.59	6.25
			b	1710.43	O	665.05	1801.11	1799.82	200.12	5.23
			c	1606.92	O	440.94	1691.87	1684.41	195.77	4.82
		3	a	1846.13	O	1789.75	1943.76	1931.94	91.27	4.65
			b	1856.51	O	5737.73	1980.97	1967.44	92.92	5.98
			c	1748.09	O	5201.29	1852.4	1839.3	94.63	5.22
		4	a	2080.98	F	7200	2085.29	2085.29	67.45	0.21
			b	1987.7	F	7200	2053.12	2052.55	71.44	3.26
			c	1903.19	F	7200	1988.2	1984.04	73.07	4.25
	4	2	a	2217.47	O	1003.59	2326.62	2323.39	273.93	4.78
			b	1991.91	O	773.91	2178.9	2162.81	310.96	8.58
			c	2007.89	O	974.76	2144.06	2132.19	331.88	6.19
		3	a	2381.97	O	6978.88	2517.53	2487.37	123	4.42
			b	2177.16	F	7200	2350.03	2337.74	131.05	7.38
			c	2195.61	F	7200	2304.99	2300.1	151.07	4.76
		4	a	2728.02	F	7200	2611.51	**2608.19**	105.03	**-4.39**
			b	2316.39	F	7200	2501.39	2452.24	100.64	5.86
			c	2238.48	F	7200	2441.38	2382.51	99.04	6.43
	5	2	a	2482.63	O	364.01	2698.08	2682.27	398.58	8.04
			b	2430.1	O	505.12	2598.63	2598.58	446.31	6.93
			c	2617.31	F	7200	2794.72	2779.23	345.84	6.19
		3	a	2722.17	F	7200	2917.27	2908.63	184.43	6.85
			b	2648.87	F	7200	2884.59	2871.54	191.61	8.41
			c	2847.49	F	7200	3003.76	2978.56	156.2	4.6
		4	a	2891.74	F	7200	3114.82	3035.03	127.83	4.96
			b	2757.69	F	7200	2990.42	2966.2	129.87	7.56
			c	NA	U	7200	3260.25	**3249.54**	121.54	NA
Average						4248.37			158.45	4.65

F: feasible solution, O: optimal solution, U: no feasible solution found, NA: not available.

Table 14 – PVRP with driver consistency results for large instances with 70 customers

$\|C\|$	$\|D\|$	$\|V\|$	Inst.	BKS	opt (BKS)	t (BKS)	Avg.	Best	t (sec.)	Gap (%)
70	2	2	a	1266.34	O	143.63	1337.62	1289.67	243.88	1.84
			b	1219.47	O	329.47	1260.45	1255.81	248.16	2.98
			c	1199.9	O	89.15	1251.24	1245.48	251.05	3.80
		3	a	1371.05	O	1295.82	1413.68	1377.57	117.61	0.48
			b	1302.41	O	4894.69	1326.73	1322.86	115.51	1.57
			c	1278.2	O	343.31	1316.57	1316.57	115.73	3.00
		4	a	1457.6	O	7201.60	1493.62	1474.98	86.75	1.19
			b	1406.92	F	7200.00	1426.64	1424.22	91.27	1.23
			c	1351.38	O	1839.77	1388.95	1387.62	90.49	2.68
	3	2	a	1666.67	O	596.59	1804.86	1783.81	389.8	7.03
			b	1650.93	O	480.56	1773.65	1771.15	400.36	7.28
			c	1602.37	O	315.57	1685.38	1671.06	436	4.29
		3	a	1799.03	O	5929.94	1937.01	1916.68	171.13	6.54
			b	1761.31	O	3150.72	1884.82	1874.7	189.2	6.44
			c	1720.3	O	709.02	1835.82	1816.73	191.11	5.61
		4	a	1964.17	F	7200.00	2085.11	2075.99	134.25	5.69
			b	1945.41	F	7200.00	2028.39	2022.15	132.22	3.94
			c	1882.63	F	7200.00	1979.14	1969.21	131.38	4.60
	4	2	a	2186.3	O	1678.45	2361.28	2329	556.76	6.53
			b	2201.65	F	7200.00	2379.49	2331.33	526.82	5.89
			c	2189.53	O	284.23	2370.45	2323.57	494.44	6.12
		3	a	2465.64	F	7200.00	2508.74	2493.86	243.12	1.14
			b	2525.04	F	7200.00	2500.74	**2485.57**	221.33	**-1.56**
			c	2381.88	F	7200.00	2528.07	2507.46	226.7	5.27
		4	a	NA	U	7200.00	2707.43	**2694.38**	158.82	NA
			b	2589.37	F	7200.00	2635.52	2622.95	162.73	1.30
	5	2	a	2885.63	O	7108.68	3153.64	3059.35	592.48	6.02
			b	2621.04	F	7200.00	2798.67	2779.76	635.1	6.06
			c	2580.35	O	5430.72	2816.88	2792.76	729.3	8.23
		3	a	NA	U	7200.00	3329.47	**3318.06**	276.52	NA
			b	NA	U	7200.00	3015.39	**3005.93**	279.41	NA
			c	2773.14	F	7200.00	2949.77	2916.22	328.39	5.16
		4	a	NA	U	7200.00	3530.81	**3530.81**	197.94	NA
			b	NA	U	7200.00	3224.13	**3223.16**	199.41	NA
			c	2887.95	F	7200.00	3158.75	3131.24	190.56	8.42
Average						4274.06			273.02	3.68

F: feasible solution, O: optimal solution, U: no feasible solution found, NA: not available.

generation techniques.

(ii) A backtracking mechanism between the three optimization phases could be implemented. This needs to improve the route construction by the reassignment of customers to weeks and days.

(iii) Other powerful meta-heuristics that do not incorporate MILP models could also be considered. This latter issue is rather important as some companies may like to be self sufficient by relying on in-house developments and prefer not to have commercial optimisation solvers as a part of their IT systems.

(iv) It is also worth highlighting that in many real life applications, parameter' uncertainty is usually common and therefore ought not be ignored. Concepts inspired from stochastic optimisation or/and fuzzy logic could be one way forward.

Acknowledgement

We are grateful to the authors of [42] for providing us with detailed results of all instances. The authors are also grateful to the referees for their interesting comments that improved both the content as well as the presentation of the paper.

Data Availability Statement

The data used in the industrial instance is the property of the company, and only available on request due to privacy restrictions.

References

[1] J. Alegre, M. Laguna, and J. Pacheco. Optimizing the periodic pick-up of raw materials for a manufacturer of auto parts. European Journal of Operational Research, 179(3):736–746, jun 2007.

[2] Y-J. An, Y-D. Kim, B.J. Jeong, and S-D. Kim. Scheduling healthcare services in a home healthcare system. Journal of the Operational Research Society, 63(11):1589–1599, nov 2012.

[3] Claudia Archetti, Elena Fernandez, and Diana L. Huerta-Munoz. The flexible periodic vehicle routing problem. Computers Operations Research, 85:58–70, 2017.

[4] Claudia Archetti, Elena Fernandez, and Diana L. Huerta-Munoz. A two-phase solution algorithm for the flexible periodic vehicle routing problem. Computers Operations Research, 99:27–37, 2018.

[5] R. Baldacci, E. Bartolini, A. Mingozzi, and A. Valletta. An exact algorithm for the period routing problem. Operations research, 59(1):228–241, 2011.

[6] M. Bellmore and G. L. Nemhauser. The traveling salesman problem: A survey. Operations Research, 16(3):538–558, jun 1968.

[7] E. J. Beltrami and L. D. Bodin. Networks and vehicle routing for municipal waste collection. Networks, 4(1):65–94, 1974.

[8] F. Blakeley, B. Arguello, B. Cao, W. Hall, and J. Knolmajer. Optimizing periodic maintenance operations for schindler elevator corporation. Interfaces, 33(1):67–79, feb 2003.

[9] Kris Braekers and Attila A Kovacs. A multi-period dial-a-ride problem with driver consistency. Transportation Research Part B: Methodological, 94:355–377, 2016.

[10] A. M. Campbell and J. H. Wilson. Forty years of periodic vehicle routing. Networks, 63(3):276–276, mar 2014.

[11] P. Campelo, F. Neves-Moreira, P. Amorim, and B. Almada-Lobo. Consistent vehicle routing problem with service level agreements: A case study in the pharmaceutical distribution sector. European Journal of Operational Research, 273(1):131–145, 2019.

[12] Roberto Cantu-Funes, M. Angelica Salazar-Aguilar, and Vincent Boyer. Multi-depot periodic vehicle routing problem with due dates and time windows. Journal of the Operational Research Society, 69(2):296–306, 2018.

[13] Pasquale Carotenuto, Stefano Giordani, Simone Massari, and Fabrizio Vagaggini. A multi-depot periodic vehicle routing model for petrol station replenishment. In Jacek Żak, Yuval Hadas, and Riccardo Rossi, editors, Advanced Concepts, Methodologies and Technologies for Transportation and Logistics, pages 421–437, 2018.

[14] N. Christofides and J. E. Beasley. The period routing problem. Networks, 14(2):237–256, 1984.

[15] S. Coene, A. Arnout, and F. C. R. Spieksma. On a periodic vehicle routing problem. Journal of the Operational Research Society, 61(12):1719–1728, dec 2010.

[16] J-F. Cordeau, M. Gendreau, and G. Laporte. A tabu search heuristic for periodic and multi-depot vehicle routing problems. Networks: An International Journal, 30(2):105–119, 1997.

[17] G. Dantzig and J. Ramser. The truck dispatching problem. Management Science, 6(1):80–91, 1959.

[18] Maria Fleischer Fauske, Carlo Mannino, and Paolo Ventura. Generalized periodic vehicle routing and maritime surveillance. Transportation Science, 54(1):164–183, 2020.

[19] Dominique Feillet, Thierry Garaix, Fabien Lehuédé, Olivier Péton, and Dominique Quadri. A new consistent vehicle routing problem for the transportation of people with disabilities. Networks, 63(3):211–224, 2014.

[20] P. Francis, K. Smilowitz, and M. Tzur. The period vehicle routing problem with service choice. Transportation Science, 40(4):439–454, nov 2006.

[21] Ahmad Ghasemkhani, Reza Tavakkoli-Moghaddam, Yaser Rahimi, Sina Shahnejat-Bushehri, and Haed Tavakkoli-Moghaddam. Integrated production-inventory-routing problem for multi-perishable products under uncertainty by meta-heuristic algorithms. International Journal of Production Research, 0(0):1–21, 2021.

[22] D. Goeke, R. Roberti, and M. Schneider. Exact and heuristic solution of the consistent vehicle-routing problem. Transportation Science, 53(4):1023–1042, 2019.

[23] Google. Distance Matrix API. https://developers.google.com/maps/documentation/distance-matrix/start. Accessed: 2018-10-30.

[24] C. Groër, B. Golden, and E. Wasil. The consistent vehicle routing problem. Manufacturing & service operations management, 11(4):630–643, 2009.

[25] V. C. Hemmelmayr, K. F. Doerner, and R. F. Hartl. A variable neighborhood search heuristic for periodic routing problems. European Journal of Operational Research, 195(3):791–802, jun 2009.

[26] Andreas Holzapfel, Alexander Hubner, Heinrich Kuhn, and Michael G. Sternbeck. Delivery pattern and transportation planning in grocery retailing. European Journal of Operational Research, 252(1):54–68, 2016.

[27] Hongtao Hu, Ye Zhang, and Lu Zhen. A two-stage decomposition method on fresh product distribution problem. International Journal of Production Research, 55(16):4729–4752, 2017.

[28] S. Kirkpatrick, C.D. Gelatt, and M.P. Vecchi. Optimization by simulated annealing. Science, 220(4598):671–680, 1983.

[29] A. A. Kovacs, B. L. Golden, R. F. Hartl, and S. N. Parragh. The generalized consistent vehicle routing problem. Transportation Science, 49(4):796–816, 2014.

[30] Attila A Kovacs, Sophie N Parragh, and Richard F Hartl. A template-based adaptive large neighborhood search for the consistent vehicle routing problem. Networks, 63(1):60–81, 2014.

[31] Igor Nikolaevich Kulachenko and Polina Aleksandrovna Kononova. A hybrid local search algorithm for the consistent periodic vehicle routing problem. Journal of Applied and Industrial Mathematics, 14(2):340–352, 2020.

[32] S. Lin and B. W. Kernighan. An effective heuristic algorithm for the traveling-salesman problem. Operations Research, 21(2):498–516, apr 1973.

[33] Cheng-Hsiang Liu. Solving the bi-objective optimisation problem with periodic delivery operations using a lexicographic method. International Journal of Production Research, 54(8):2275–2283, 2016.

[34] Z. Luo, H. Qin, C. Che, and A. Lim. On service consistency in multi-period vehicle routing. European Journal of Operational Research, 243(3):731 – 744, 2015.

[35] Simona Mancini, Margaretha Gansterer, and Richard F. Hartl. The collaborative consistent vehicle routing problem with workload balance. European Journal of Operational Research, 293(3):955–965, 2021.

[36] B. Martin, W. H. Paul, and Y. Leslie-Ann. The two-period travelling salesman problem applied to milk collection in ireland. Computational Optimization and Applications, 7(3):291–306, May 1997.

[37] A. C. Matos and R. C. Oliveira. An experimental study of the ant colony system for the period vehicle routing problem. In Ant Colony Optimization and Swarm Intelligence, pages 286–293. Springer Berlin Heidelberg, 2004.

[38] Bilal Messaoudi, Ammar Oulamara, and Nastaran Rahmani. Multiple periods vehicle routing problems: A case study. In Arnaud Liefooghe and Luís Paquete, editors, Evolutionary Computation in Combinatorial Optimization, pages 83–98, Cham, 2019. Springer International Publishing - LNCS (11452).

[39] Andrea Mor and Maria Grazia Speranza. Vehicle routing problems over time: a survey. Annals of Operations Research, pages 1–21, 2022.

[40] M. Mourgaya and F. Vanderbeck. Column generation based heuristic for tactical planning in multi-period vehicle routing. European Journal of Operational Research, 183(3):1028–1041, dec 2007.

[41] Phuong Khanh Nguyen, Teodor Gabriel Crainic, and Michel Toulouse. A hybrid generational genetic algorithm for the periodic vehicle routing problem with time windows. Journal of Heuristics, 20(4):383–416, 2014.

[42] I. Rodríguez-Martín, J-J. Salazar-González, and H. Yaman. The periodic vehicle routing problem with driver consistency. European Journal of Operational Research, 273(2):575–584, 2019.

[43] D. Ronen and C. A. Goodhart. Tactical store delivery planning. Journal of the Operational Research Society, 59(8):1047–1054, aug 2008.

[44] S. Ropke and D. Pisinger. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. Transportation Science, 40(4):455–472, nov 2006.

[45] S. Ropke and D. Pisinger. A unified heuristic for a large class of vehicle routing problems with backhauls. European Journal of Operational Research, 171(3):750–775, jun 2006.

[46] Ann-Kathrin Rothenbächer. Branch-and-price-and-cut for the periodic vehicle routing problem with flexible schedule structures. Transportation Science, 53(3):850–866, 2019.

[47] David Russell, Robert J Rosati, Peri Rosenfeld, and Joan M Marren. Continuity in home health care: is consistency in nursing personnel associated with better patient outcomes? Journal for healthcare quality, 33(6):33–39, 2011.

[48] R. Russell and W Igo. An assignment routing problem. Networks, 9:1 – 17, 03 1979.

[49] R. A. Russell and D. Gribbin. A multiphase approach to the period routing problem. Networks, 21(7):747–765, dec 1991.

[50] S. Salhi. Heuristic search: The emerging science of problem solving. Springer, 2017.

[51] Erich Schubert and Peter J Rousseeuw. Faster k-medoids clustering: improving the pam, clara, and clarans algorithms. In International Conference on Similarity Search and Applications, pages 171–187. Springer, 2019.

[52] Karen Smilowitz, Maciek Nowak, and Tingting Jiang. Workforce management in periodic delivery operations. Transportation Science, 47(2):214–230, 2013.

[53] F. Stavropoulou. The consistent vehicle routing problem with heterogeneous fleet. Computers Operations Research, 140:105644, 2022.

[54] F. Stavropoulou, P.P. Repoussis, and C.D. Tarantilis. The vehicle routing problem with profits and consistency constraints. European Journal of Operational Research, 274(1):340–356, 2019.

[55] Christos D Tarantilis, Foteini Stavropoulou, and Panagiotis P Repoussis. A template-based tabu search algorithm for the consistent vehicle routing problem. Expert Systems with Applications, 39(4):4233–4239, 2012.

[56] N. Teemu, K. Jari, N. Harri, and B. Olli. Improved route planning and scheduling of waste collection and transport. Expert Systems with Applications, 30(2):223–232, feb 2006.

[57] Paolo Toth and Daniele Vigo. Vehicle routing: problems, methods, and applications. SIAM, 2014.

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

[59] Thibaut Vidal, Gilbert Laporte, and Piotr Matl. A concise guide to existing and emerging vehicle routing problem variants. European Journal of Operational Research, 286(2):401–416, 2020.

[60] Yuan Wang, Ling Wang, Guangcai Chen, Zhaoquan Cai, Yongquan Zhou, and Lining Xing. An improved ant colony optimization algorithm to the periodic vehicle routing problem with time window and service choice. Swarm and Evolutionary Computation, 55:100675, 2020.

[61] Zefeng Xu and Yanguang Cai. Variable neighborhood search for consistent vehicle routing problem. Expert Systems with Applications, 113:66–76, 2018.

[62] Sandra Zajac. An adaptive large neighborhood search for the periodic vehicle routing problem. In Tolga Bektaş, Stefano Coniglio, Antonio Martinez-Sykora, and Stefan Voß, editors, Computational Logistics, pages 34–48. Springer International Publishing, 2017.