International Journal of Production Research

ISSN: 0020-7543 (Print) 1366-588X (Online) Journal homepage: www.tandfonline.com/journals/tprs20

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

Bilal Messaoudi, Ammar Oulamara & Said Salhi

To cite this article: Bilal Messaoudi, Ammar Oulamara & Said Salhi (2023) A decomposition approach for the periodic consistent vehicle routeing problem with an application in the cleaning sector, International Journal of Production Research, 61:22, 7727-7748, DOI: 10.1080/00207543.2022.2162145

To link to this article: https://doi.org/10.1080/00207543.2022.2162145

Published online: 06 Jan 2023.

INTERNATIONAL JOURNAL OF PRODUCTION RESEARCH
2023, VOL. 61, NO. 22, 7727–7748
https://doi.org/10.1080/00207543.2022.2162145

Taylor & Francis
Taylor & Francis Group

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

Bilal Messaoudi[a,b], Ammar Oulamara[a,b,c] and Said Salhi[c]

[a]University of Lorraine, LORIA laboratory, Vandoeuvre-les-Nancy, France; [b]Antsway SA, ARTEM, Nancy, France; [c]Centre for Logistics & Heuristic Optimisation (CLHO), Kent Business School, University of Kent, Canterbury, 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 routeing 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 minimise the number of vehicles used and the total distance travelled. This problem is a generalisation of the periodic vehicle routeing problem. We decompose the problem into two sub-problems, namely, the planning and routeing optimisation sub-problems. We construct a mathematical model for the former and a large neighbourhood 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.

ARTICLE HISTORY
Received 4 October 2021
Accepted 29 November 2022

KEYWORDS
Routing; driver consistency; mixed-integer linear programming; large neighbourhood search; Decomposition methods

1. Introduction

This paper is motivated by a vehicle routeing problem observed in french company that provides sanitary products (e.g. cleaning gel, toilet paper), and soft cleaning service for business customers. Currently, the company serves about 6000 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. The driver, though he does not carry out cleaning jobs on a daily basis, but conducts small extra cleaning related tasks such as vacuuming the areas when replacing used mats with clean ones and also putting products in specific storage rooms in the companies or directly in the toilets. 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 a soft cleaning service time plus the product delivery service time. The objective is to meet customer demands with the smallest number of vehicles, while respecting operational constraints.

The company currently uses a basic vehicle routeing 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

CONTACT Ammar Oulamara ammar.oulamara@univ-lorraine.fr University of Lorraine, LORIA laboratory, Campus Scientifique – BP 239, Vandoeuvre-les-Nancy 54506, France; Antsway SA, 92, rue du sergent Blandan, ARTEM, Nancy 54000, France; Centre for Logistics & Heuristic Optimisation (CLHO), Kent Business School, University of Kent, Canterbury CT2 7FS, UK

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.

This real world vehicle routeing problem is known in the literature as a periodic vehicle touring problem (PVRP) with consistency. 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 of basic cyclic routes in each corresponding planning period (Holzapfel et al. 2016). Consistency in PVRP ensures that an agent visits the same set of clients (driver consistency) which allows to establish a personal relationship, and each customer.

This paper addresses the routeing problem with consistency constraints as described in the industrial case study, i.e. each customer is visited by the same agent on the same day of the week over the planning horizon. 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 An et al. (2012) where nurses always visit the same patients at regular periods. The objective is to minimise 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 optimisation. This interesting logistical practical problem was introduced by Messaoudi, Oulamara, and Rahmani (2019) 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 routeing 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 Messaoudi, Oulamara, and Rahmani (2019). 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 routeing 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 4.2.

(ii) We transformed our approach into a more flexible and powerful technique that also tackles related routeing 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 Section 2, to the best of our knowledge, this problem of periodic vehicle routeing with consistent constraints has not been considered in the literature, it is characterised 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.

The rest of the paper is organised as follows. Section 2 provides a brief review focussing on two categories, namely, the periodic VRP and consistency VRP problems. In Section 3, provides a formal description and the necessary notations and develop a mathematical model formulation of the problem. Our solution method is presented in Section 4. Computational results are provided and analysed in Section 5. Finally, some conclusions and research directions are outlined in Section 6.

2. Literature review

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

2.1. Periodic vehicle routeing problem

The periodic vehicle routeing problem – PVRP (Beltrami and Bodin 1974; R. Russell and Igo 1979), is a generalisation of the classical vehicle routeing 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 routeing decision while minimising the total cost over the planning horizon.

The PVRP has attracted a considerable amount of interest among researchers and practitioners (Toth and Vigo 2014). This is mainly due to the wide range of real-world applications that fit into this class of routeing. Among the applications, we can cite a few such as waste and garbage collection (Teemu et al. 2006; Matos and Oliveira 2004), animal waste (Coene, Arnout, and Spieksma 2010), home health care services (An et al. 2012), delivery of blood products to hospitals (Hemmelmayr, Doerner, and Hartl 2009), retail stock supply (Ronen and Goodhart 2008), maintenance service (Blakeley et al. 2003), perishable products (Ghasemkhani et al. 2021), and maritime surveillance (Fauske, Mannino, and Ventura 2020). Other real life applications are reported in Campbell and Wilson (2014).

The problem was introduced by Beltrami and Bodin (1974) where the authors studied the routeing problem related to municipal waste collection. They considered a one week planning horizon with the objective of minimising both the number of used vehicles and the total travel time. R. Russell and Igo (1979) proposed a formal definition of the problem and developed three cluster-based constructive methods with the aim to minimise the total distance travelled per week. Christofides and Beasley (1984) 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. Francis, Smilowitz, and Tzur (2006) studied a variant of the PVRP in which the service frequency is a decision variable with the objective of maximising service benefits and minimising routeing costs. Mourgaya and Vanderbeck (2007) put forward an exact method for the PVRP with the objective of balancing the workload across vehicles and spacial compactness of the routes. Baldacci et al. (2011) proposed a new formulation in generating strong lower bounds for the PVRP problem. Rothenbächer (2019) addresses the periodic vehicle routeing problem with time windows (PVRPTW) and an exact branch-and-price-and-cut algorithm is proposed. Some studies have developed exact methods for variants of the PVRP problem, such as flexible PVRP (Archetti, Fernandez, and Huerta-Munoz 2017, 2018). It seems that no contribution to the exact solution of the classical PVRP has been achieved after Baldacci et al. (2011). The main works in the literature has so far been focussed on the study of new variants inspired by applications (Mor and Speranza 2022) instead.

Meta-heuristics are the most adopted approaches for solving the PVRP and its variants. For instance, R. A. Russell and Gribbin (1991) presented a solution method that consists of an initial route design, followed by three different improvement phases aimed at escaping from the local optima. Cordeau, Gendreau, and Laporte (1997) presented a tabu search method for solving the periodic multi-depot vehicle routeing problem (MDVRP) in which two types of neighbourhood operators are proposed. Alegre, Laguna, and Pacheco (2007) 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. Hemmelmayr, Doerner, and Hartl (2009) put forward a variable neighbourhood search (VNS) algorithm with the solution acceptance being based on simulated annealing, Vidal et al. (2012) developed a hybrid genetic algorithm (GA) for the PVRP, and Zajac (2017) proposed an adaptive large neighbourhood 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, Nguyen, Crainic, and Toulouse (2014) proposed a hybrid genetic algorithm, and Wang et al. (2020) developed a heuristic algorithm based on improved ant colony optimisation (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, Cantu-Funes, Salazar-Aguilar, and Boyer (2018) developed a reactive greedy randomised adaptive search procedure, and Carotenuto et al. (2018) proposed a hybrid genetic algorithm. Interesting and informative surveys on periodic vehicle routeing problems can be found in Campbell and Wilson (2014) and Mor and Speranza (2022).

2.2. Consistent vehicle routeing problem

In recent years, Vehicle Routeing Problems (VRPs) with consistency features have received significant attention due to their practical importance. Three types of consistency are usually considered in the literature (Vidal, Laporte, and Matl 2020); (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 (D. Russell et al. 2011) where the operator (driver) knows the needs and preference of their patients (customers).

The consistency constraints (driver and time consistencies) were formally introduced in Groër, Golden and Wasil (2009). 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. Tarantilis, Stavropoulou, and Repoussis (2012) 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 Kovacs, Parragh, and Hartl (2014) a template-based Adaptive Large Neighbourhood Search algorithm (ALNS) is developed to solve the driver and time consistent periodic VRP problem. Xu and Cai (2018) and Kulachenko and Kononova (2020) also solve the same problem but using a variable neighbourhood search (VNS) algorithm instead. In Stavropoulou (2022) a hierarchical Tabu Search framework is proposed in which an upper-level Tabu Search method is combined with variable neighbourhood descent algorithm. Messaoudi, Oulamara, and Rahmani (2019) considered a PVRP problem with day visits and driver consistencies. A heuristic decomposition approach is proposed to solve the problem with the objective of minimising the number of vehicles and total cost. Feillet et al. (2014) considered the time consistency only, in which the day is discretised in time windows and imposes consistency by bounding the number of different time segments during which a customer is served. Here, a large neighbourhood 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, Campelo et al. (2019) 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 Goeke, Roberti, and Schneider (2019) who proposed an exact method based on column generation, and Rodríguez-Martín, Salazar-González, and Yaman (2019) 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, Smilowitz, Nowak, and Jiang (2013) focussed on maximising the number of times a unique driver visits each customer and in both Braekers and Kovacs (2016) and Luo et al. (2015) a limit of the number of different drivers serving any customer was imposed. In Mancini, Gansterer, and Hartl (2021) a time and collaborative consistency were considered where carriers can exchange customers who have to be serviced on a regular basis, and in Stavropoulou, Repoussis, and Tarantilis (2019) driver consistency PVRP with profit was considered. In Kovacs et al. (2014) a generalised consistency problem was considered where each customer is visited by a limited number of drivers around the same arrival time. Table 1 summarises some of the published work in this area.

In this study, we will be extending and enhancing the work of Messaoudi, Oulamara, and Rahmani (2019) mentioned in Section 1, and also comparing our results against those recently produced by Rodríguez-Martín, Salazar-González, and Yaman (2019) and Goeke, Roberti, and Schneider (2019) to assess the performance of the proposed algorithm.

3. Problem description, notations and formulation

3.1. 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 on the same day of the week (day consistency) by the same agent (driver consistency) and he/she must be visited no more than once a week during this planning 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 characterised by a time window $[e_i, l_i]$ of visits, where e_i and l_i are the earliest and latest times to serve customer i and a list P_i of requested products with $P_i \subseteq P$. For each product, $j \in P_i$,

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

Paper	Periods of the planning	Types of visits	Consistency constraints	Others constraints	Objective (Min)	Methods
Groër, Golden and Wasil (2009)	Days	Known and fixed	Driver consist. Arrival time consist.	Vehicle capacity limit, Travel time limit	Total travel time	Record-to-Record Tabu search (TTS)
Tarantilis, Stavropoulou, and Repoussis (2012)	Days	Known and fixed	Driver consist. Arrival time consist.	Vehicle capacity limit, Travel time limit	Total travel time	Template-based
Kovacs, Parragh, and Hartl (2014)	Days	Known and fixed	Driver consist. Arrival time consist.	Vehicle capacity limit, Travel time limit	Total travel time	Template-based ALNS
Kovacs et al. (2014)	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)
Feillet et al. (2014)	Days	Known and fixed	Visiting time consist.	Vehicle capacity limit	Total travel time	Large neighbourhood search (LNS)
Xu and Cai (2018)	Days	Known and fixed	Driver consist. Arrival time consist.	Vehicle capacity limit, Travel time limit	Total travel time	Variable neighbourhood search (VNS)
Goeke, Roberti, and Schneider (2019)	Days	Known and fixed	Driver consist.	Vehicle capacity limit, Travel time limit	Total travel time	Exact method (column generation)
Rodríguez-Martín, Salazar-González, and Yaman (2019)	Days	Decision variables from a set of patterns	Driver consist.	Limited number of customers per route	Total travel cost	Exact method (branch-and-cut)
Stavropoulou, Repoussis, and Tarantilis (2019)	Days	Known and fixed	Driver consist Arrival time consist.	Vehicle capacity limit, Travel time limit	Maximize the net profit	Adaptive Tabu Search
Kulachenko and Kononova (2020)	Days	Decision variables from a set of patterns	Driver consist.	Vehicle capacity limit, Travel time limit	Total travel distance	Variable neighbourhood search (VNS)
Stavropoulou (2022)	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)

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 a driver 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 minimising the number of vehicles used and minimising the total travel distance. An illustrative example is given in the following section.

3.2. Illustrative example

To illustrate the problem, let us consider a PVRP instance with a simplified planning horizon of 6 weeks ($W = 6$) where each week consists of two days ($D = 2$) delivery period. As shown in Figure 1, there are two available vehicles to handle the demands of sixteen customers for two products P_1 and P_2. For each product, visit patterns

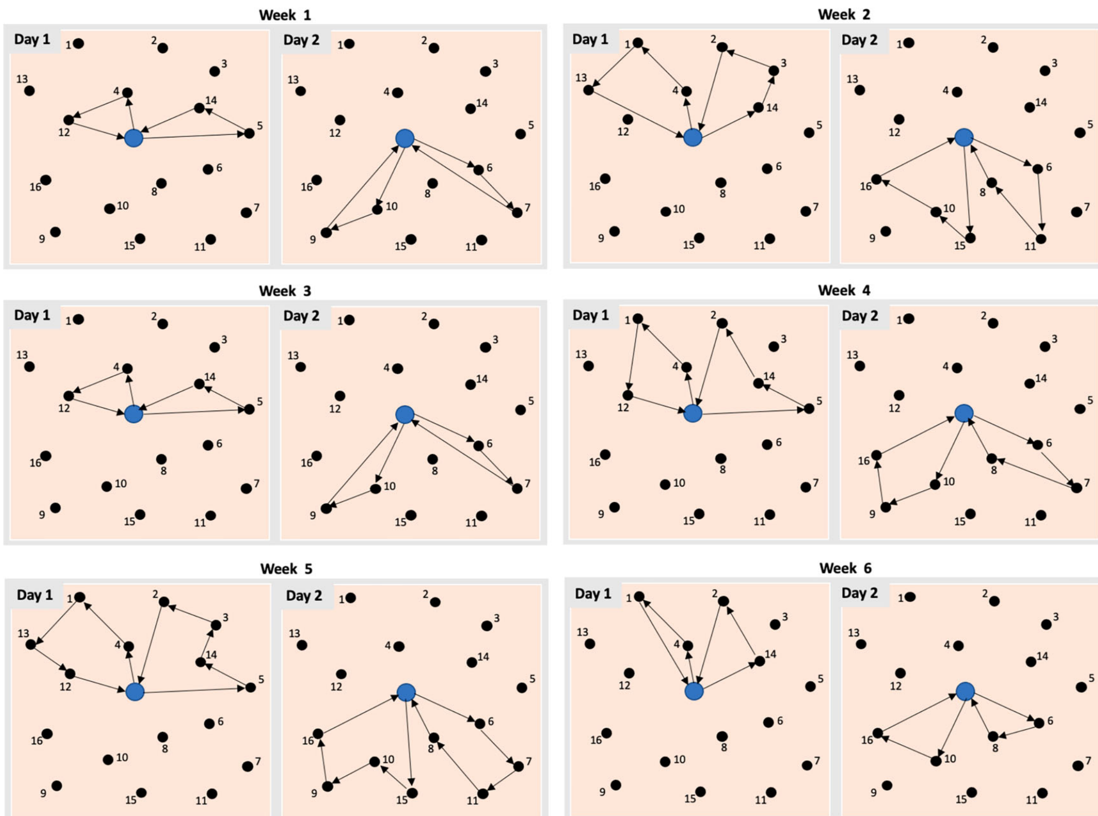


Figure 1. Illustrative example of transportation plan corresponding to patterns in bold in Table 2.

Table 2. Illustrative example of visiting patterns of customers.

Customers	Patterns of P_1	Patterns of P_2
1,2,8,16	{(1,4);**(2,5)**;(3,6)}	{(1,3,5); **(2,4,6)**}
3,11,13,15	{(1,4);**(2,5)**;(3,6)}	∅
5,7,9,12	{**(1,3,5)**;(2,4,6)}	{**(1,4)**;(2,5);(3,6)}
4,6,10,14	{**(1;2;3;4;5;6)**}	∅

for customers are given in Table 2. In this example, customers 1, 2, 3, 4, 5, 12, 13 and 14 are assigned to day one, while customers 6, 7, 8, 9, 10, 11, 15 and 16 are assigned to day two. Vehicle (agent) one serves sets $\{1, 4, 12, 13\}$ and $\{9, 10, 15, 16\}$ of customers on days one and two, respectively, while vehicle (agent) two serves sets $\{2, 3, 5, 12\}$ and $\{6, 7, 8, 11\}$ of customers on days one and two, respectively. 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.

3.3. 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 τ_{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}.

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

Subject to

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

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

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

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

$$h_{0kdw} = u_{kdw} \qquad \forall k \in \mathcal{M}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \quad (6)$$

$$h_{ikdw} \leq u_{kdw} \qquad \forall i \in V \setminus \{0\}, \quad \forall k \in \mathcal{M}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \qquad (7)$$

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

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

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

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

$$\sum_{i\in V}\sum_{j\in P_i} a^{ij}_{rw} q_{ij} h_{ikdw} \leq C_k \qquad \forall k \in \mathcal{M}, \quad \forall d \in \mathcal{D}, \quad \forall w \in \mathcal{W} \qquad (12)$$

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

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

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

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

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

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

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

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

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

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

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

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

The objective function (1) minimises the total travel distance and the number of vehicles used, where α and β are parameters set by the decision-maker according to the priority they give to each part of the objective function. The parameter α can be seen as the unit cost of the total travelled distance and β as the average fixed cost of using a vehicle over the planning horizon. 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) with constraints (9) ensure that each route starts and ends at 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 generalisation 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.

4. Solution methods

Since the considered routeing 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 6000 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 (Hu, Zhang, and Zhen 2017; Coene, Arnout, and Spieksma 2010; Liu 2016). Our decomposition method is similar to the one initiated in Messaoudi, Oulamara, and Rahmani (2019). The method consists of two phases, namely, the planning phase and the routeing 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 routeing phase consists of constructing daily routes over the planning horizon. The aim here is to minimise the number of vehicles used and the total distance travelled by the vehicles while respecting driver consistency constraints.

In Messaoudi, Oulamara, and Rahmani (2019) a heuristic method based on a day-pattern modelling 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 optimised 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 optimisation 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 routeing 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.

4.1. 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.

4.1.1. The weeks 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 3, the value f_{ij} defines the set R_{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 $v_k = v_1 + (k-1)\frac{W}{f_{ij}}$, $k = 2, \ldots, f_{ij}$. For example, if $f_{ij} = 3$ and the set of possible delivery patterns of product j is $R_{ij} = \{(1,5,9); (2,6,10); (3,7,11); (4,8,12)\}$ then choosing the first $v_1 = 1$ we obtain $v_2 = v_1 + 4 = 5$ and $v_3 = v_1 + 8 = 9$. 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. Finally, we introduce the notation $u = v[t]$ which means that $u = v \mod t$ that will be useful in the description of the mathematical model below.

The weeks planning problem is solved in two steps: the first step uses a mathematical model to assign customers to weeks in order to satisfy customer visit constraints and minimise the maximum workload over all weeks. This objective balances the workload over the weeks of the planning horizon. The second step of the solving approach takes into account the location of the customers. It starts by grouping the customers into clusters, then uses the same mathematical model of the first phase to assign the customers over the weeks, with the objective of reducing the number of clusters visited each week, while respecting the customer visit constraints and ensuring that the workload of each week does not exceed the maximum workload found in the first step.

– *First step* : In this step we use a mixed integer linear program (P) to model the week planning problem. The binary variable x_{ijw} specifies whether or not the delivery of product j of customer i is performed during week w. The objective is to minimise the maximum workload of weeks that will be used in the second step.

$$\text{(P)} \quad \min \quad L^* = L$$

$$\sum_{i \in V} \sum_{j \in P_i} s_{ij} x_{ijw} \leq L \quad \forall w \in \mathcal{W} \tag{25}$$

$$\sum_{i \in V} \sum_{j \in P_i} q_{ij} x_{i,j,w} \leq C \quad \forall w \in \mathcal{W} \tag{26}$$

$$\sum_{w=1}^{r_{ij}} x_{ijw} = 1 \quad \forall i \in V, \quad \forall j \in P_i \tag{27}$$

$$x_{ijw} = x_{ij(w+k.r_{ij})} \quad \forall i \in V, \quad \forall j \in P_i, \quad w = 1, \ldots, r_{ij}, \quad k = 1, \ldots, f_{ij} - 1 \tag{28}$$

$$x_{ig_i w} \leq \sum_{u \in H_w} x_{iju} \quad \forall i \in V, \quad \forall j \in P_i, \quad w = 1, \ldots, r_{ij},$$
$$H_w = \{u = 1, \ldots, r_{ij} : \text{ there exist } l \in \{1, \ldots, f_{ij}\} \text{ such that } u + (l-1) r_{ij} = k \text{ where } k = 1 \ldots, W \text{ and } k[r_{ig_i}] = w\} \tag{29}$$

$$x_{ijw} \in \{0, 1\} i \in V, \quad j \in P_i, \quad w = 1, \ldots W \tag{30}$$

Constraints (25) limit the workload of each week of the planning horizon. Constraints (26) ensure that vehicles' weekly capacity is not exceeded. Constraints (27) show that exactly one week is chosen for the first visit for each product and each customer. This is introduced so that the customer is visited according to the frequency of the product in the following weeks as stated in Constraints (28). Constraints (29) restrict the number of visits to each customer, depending on the requested products and their frequencies. An example of the application of constraint (29) is used is given below. Constraints (30) refer to the binary variables.

Example : Given a 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, according to constraints (29) 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 $\{1, 2, 3\}$ while that of p_2 takes its values in the set $\{1, 2, 3, 4\}$. If week 1 is chosen for p_1, then set of possible visits for p_2 remains $H_1 = \{1, 2, 3, 4\}$, see Figure 3.

– *Second step*: In order to consider the location of customers in the assignment of customers to weeks, we proceed to a partition of the customers into K clusters using a fast implementation of the k-medoids clustering algorithm (Schubert and Rousseeuw 2019) before assigning scenarios to customers. This clustering algorithm partitions the customers into K clusters with the objective of minimising the squared distances between the customers in a cluster and its centre 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, assigning the other

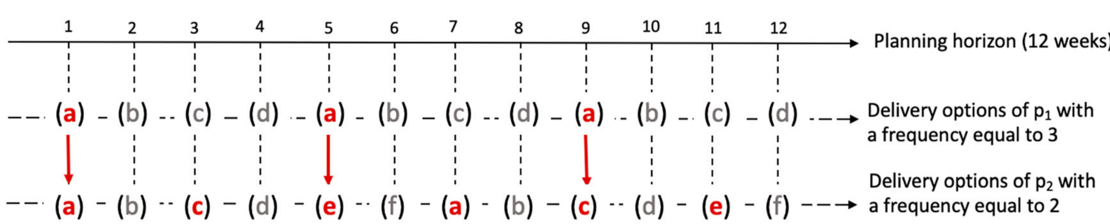


Figure 2. Example of delivery options for p_1 with a reduction of the possible delivery options for p_2.

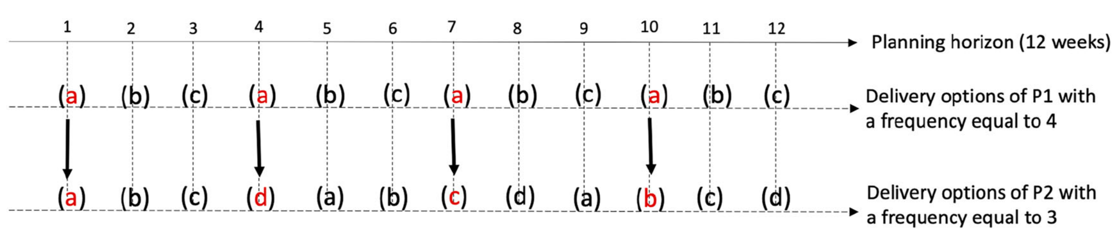


Figure 3. Example of delivery options for p_1 without a reduction of the possible delivery options for p_2.

customers to these centres, and computing the sum of the squared distances. The algorithm improves the clustering by considering all possible changes of medoids with non-medoids, which gives $K(n-K)$ candidates for swapping. The best change that reduces the sum of squared distances the most is then chosen. This process is repeated until no further improvement is 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 equals to 1 if customer i is in cluster l, $l = 1, \dots, K$.

After partitioning the customers into clusters, we use a mixed integer linear program (Q) to model the weeks planning problem which takes clusters into account when assigning customers to weeks. In addition to the binary variable x_{ijw} used in model (P), we define the binary variable y_{lw} which specifies whether or not the customer of cluster l is visited during week w. The objective here is to minimise the sum of clusters visited over the W weeks.

$$\text{(Q)} \quad \min \quad \sum_{l \in \mathcal{K}} \sum_{w \in \mathcal{W}} y_{lw}$$

$$\sum_{i \in V} \sum_{j \in P_i} u_{il} x_{ijw} \leq M y_{lw} \quad \forall l \in \mathcal{K}, \quad \forall w \in \mathcal{W} \tag{31}$$

$$\sum_{i \in V} \sum_{j \in P_i} s_{ij} x_{ijw} \leq L^* \quad \forall w \in \mathcal{W} \tag{32}$$

$$(26)\text{–}(30)$$

Constraints (31) ensure that no customer of a cluster is visited if this cluster is not selected. Constraints (32) limit the total workload of each week to L^*, where L^* is the optimal solution of model (P).

We solve both (P) and (Q) problems optimally using the optimisation 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 customers into K clusters.
2. Solve the (P) problem and obtain the value of L^*
3. Solve the (Q) problem and obtain the assignment of customers to weeks

4.1.2. 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 U 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 of working time between the days of the week.

$$\min \quad U$$

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

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

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

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

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

Constraints (33) restrict the total workload of each day. Constraints (34) guarantee that each vehicle's 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.

4.2. The routeing phase

After assigning all customers to the days of the planning horizon using the week and the days planning models (Sections 4.1.1 and 4.1.2) 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 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 routeing step is therefore to build optimised 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 optimised 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 routeing 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 routeing 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 $(C_k, C_k, \dots, C_k)$ 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 the 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 $\tilde{P}_{d,i} = (\tilde{P}^1_{d,i}, \dots, \tilde{P}^W_{d,i})$, 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 the 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

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

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 day d of each week of the planning horizon. This is then followed by a post-optimisation for each week, where customers with zero demands are then removed from each master-route, and the master-route is re-optimised again as a TSP problem. Our LNS method constructs master-routes of VRP(d) with the objective of minimising the number of used vehicles and the total distance travelled. 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 Salhi (2017).

Algorithm 2: LNS outline

Result: best feasible solution s^b
initialization: $s^b \leftarrow s$,
while *stop criterion is not met* **do**
  selectOperators(Ω^-, Ω^+);
  $s^t \leftarrow$ *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 at least q nodes from the current solution, with $q \in [\xi_{\min}, \xi_{\max}]$, where $\xi_{\min}$ 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 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 Ropke and Pisinger (2006b) 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 at least 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 minimise the number of vehicles (e.g. Ropke and Pisinger 2006a). 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 to this set of nodes as 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 insertions 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. In our experiment, we set k to 2, which corresponds to regret−2 in the literature (Ropke and Pisinger 2006a).

At each iteration of the LNS method, the quality of the solution is evaluated using three metrics: (i) the total distance travelled 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, where the first two being discussed earlier in mathematical model (Section 3.3).

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-optimising these routes again. Here, the intra-route re-optimisation problem reduces to the TSP problem (Bellmore and Nemhauser 1968). In this step, for simplicity, we use the Lin-Kernighan heuristic (Lin and Kernighan 1973) though other more powerful heuristics could also be adopted instead.

5. 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.

5.1. 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.

5.1.1. 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 correspond on average to 11.54 requests per customer. The service times of customers are in the range of [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 to 1600 kg. The maximum driving and servicing time is limited to 7 h and 30 min per day for each driver. Note that in the

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

initial experiment conducted in Messaoudi, Oulamara, and Rahmani (2019), the maximum routeing time was set to 7 h for each vehicle (agent). However, the reorganisation of the company's internal logistics allowed 30 additional minutes of driver maximum service duration to be used in the routeing phase. Therefore, for consistency, we will also compare the additional gain obtained by the extension of the maximum route duration.

5.1.2. Week planning results

The mixed integer linear programs of the week planning model defined in Algorithm 1 (Section 4.1.1) are optimally solved in a very short time, less than 2 min. (Q) 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 Messaoudi, Oulamara, and Rahmani (2019). 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 kilometers. 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 min for CL-60, 15 min for CL-120, 18 min for CL-120 and 26 min for Messaoudi, Oulamara, and Rahmani (2019). 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 Messaoudi, Oulamara, and Rahmani (2019) 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 6. Statistical data on the distribution of customers 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, Oulamara, and Rahmani (2019)	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

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, Oulamara, and Rahmani (2019)												
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

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

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, Oulamara, and Rahmani (2019)												
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

5.1.3. Days planning results

The MILP of the days planning model is solved with a gap of 0.2% in 3 min. 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 Messaoudi, Oulamara, and Rahmani (2019) 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 Messaoudi, Oulamara, and Rahmani (2019). Table 8 summarises the distribution of the number of customers 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 Messaoudi, Oulamara, and Rahmani (2019) which does not consider the effect of clustering. Thus, for the routeing phase, we will use the CL-60 clustering days planning results only.

Table 8. Statistical parameters on the distribution of customers 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

5.1.4. Routeing results

In this phase, we use the LNS algorithm developed in Section 4.2 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 (Kirkpatrick, Gelatt, and Vecchi 1983). As stated in Ropke and Pisinger (2006a), 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 related to LNS as defined in Section 4.2 are set to $(\alpha, \beta, \gamma) = (1, 3000, 120, 000)$ which are based on preliminary experimentation. Time and distance travel matrices are computed using Google Distance Matrix API (Google 2018). In the following, we present the results of LNS metaheuristic, and we compare that results with results of Messaoudi, Oulamara, and Rahmani (2019).

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 min (7 h and 30 min) per driver as explained in Section 5.1.1. A computing time limit of 2 h was also set. First, in Table 9 we evaluate the effect of increasing the maximum duration of the routes from 420 min (7 h) to

450 min (7 h and 30 min) in the resolution of the problem with the pattern method of Messaoudi, Oulamara, and Rahmani (2019). 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 km) due to the extra flexibility in allowing a small increase of 30 min. This demonstrates that time flexibility in routeing can be paramount and ought not to be overlooked.

In the following section, we compare the pattern method (Messaoudi, Oulamara, and Rahmani 2019) with the vector method with a maximum route duration of 450 min.

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 (Messaoudi, Oulamara, and Rahmani 2019). The results empirically demonstrate that this innovative idea of modelling 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 Messaoudi, Oulamara, and Rahmani (2019). 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 routeing duration obtained for each day of each week, and a comparison with Messaoudi, Oulamara, and Rahmani (2019) is also provided. We can see that the average service duration of the agents is 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 Messaoudi, Oulamara, and Rahmani (2019) 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.

5.2. Comparison with related work

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 Goeke, Roberti, and Schneider (2019) and the other by Rodríguez-Martín et al. Rodríguez-Martín, Salazar-González, and Yaman (2019) where the best-known results are also reported.

5.2.1. Comparison with the study of Goeke, Roberti, and Schneider (2019)

In Goeke, Roberti, and Schneider (2019), the periodic vehicle routeing problem assumes that the visit schedule of each customer is given and the objective is to minimise the total vehicle operating time over the time horizon while satisfying the driver and the arrival-time consistency constraints. Arrival-time consistency refers

Table 9. Total distances obtained for real instance over 12 weeks with pattern method (Messaoudi, Oulamara, and Rahmani 2019).

	Weeks												
	1	2	3	4	5	6	7	8	9	10	11	12	Total
Km (420 min)	11,235.9	10,722	10,391.7	10,333.5	11,166.8	10,734.5	10,477.8	9965.3	11,129.2	10,994	10,393.6	10,026.3	127,570.6
Km (450 min)	11,051.6	10,371.1	10,253	10,086.7	10,988.3	10,371.4	10,320.5	9774.1	10,990.3	10,547.6	10,255.1	9796	114,434.3

Table 10. Total distances obtained for real instances over 12 weeks (in kilometers).

	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

Note: Vector: The proposed LNS method, Pattern: Method of Messaoudi, Oulamara, and Rahmani (2019).

Table 11. Average routeing 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

Note: Vector: The proposed LNS method, Pattern: Method of Messaoudi, Oulamara, and Rahmani (2019).

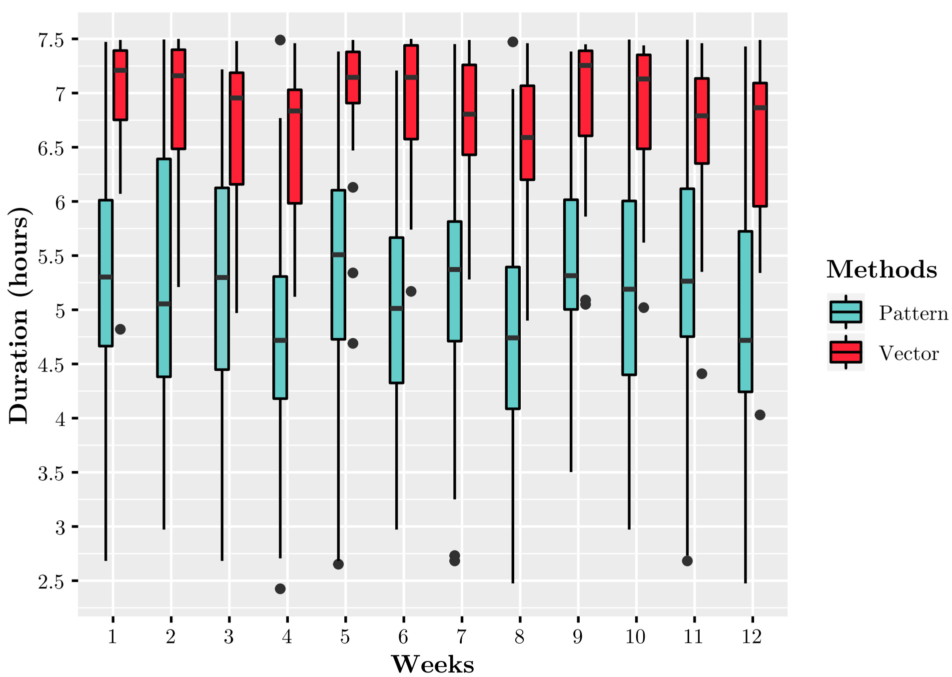


Figure 4. Distribution of route durations with the Vector and Pattern methods for Monday.

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 $L = +\infty$ refers 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 on presenting the results on instances of the set C with value $L = +\infty$.

To be consistent with Goeke, Roberti, and Schneider (2019), we have also adapted our LNS method to the same objective function to be minimised, 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 Goeke, Roberti, and Schneider (2019), the average solution value of the 10 runs (Avg), the best solution value of the 10 runs (Best), average computing time (t)

Table 12. Consistent VRP results: comparison with Goeke, Roberti, and Schneider (2019).

Service frequency	Instance	|C|	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

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 Goeke, Roberti, and Schneider (2019), 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.

5.2.2. Comparison with the study of Rodríguez-Martín, Salazar-González, and Yaman (2019)

In Rodríguez-Martín, Salazar-González, and Yaman (2019), authors 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 have 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 Rodríguez-Martín, Salazar-González, and Yaman (2019). 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 customer 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 1 when pattern p of customer i is selected. Let p_d be a parameter equal to 1 if day d is included in pattern p. The resulting modified

MILP model is as follows.

$$\min \quad K$$

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

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

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

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

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 minimised in the objective function. Constraints (42) refer to the binary decision variables.

The instances in Rodríguez-Martín, Salazar-González, and Yaman (2019) 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 min in computing time for all instances. This time remains relatively much lower than the time limit of two hours fixed in Rodríguez-Martín, Salazar-González, and Yaman (2019). 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

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

$\lvert C\rvert$	$\lvert D\rvert$	$\lvert V\rvert$	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

Note: 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.

$\lvert C\rvert$	$\lvert D\rvert$	$\lvert V\rvert$	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

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

($\lvert C\rvert$), the number of days in the planning horizon ($\lvert D\rvert$), the number of vehicles ($\lvert V\rvert$), the type of instances (Inst), the best-known solution (BKS) from Rodríguez-Martín, Salazar-González, and Yaman (2019) 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 h 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 Rodríguez-Martín, Salazar-González, and Yaman (2019). 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.

6. Conclusion and suggestions

In this paper, we have investigated the design of tactical plans for a challenging periodic routeing problem encountered in a company in the cleaning service sector. We modelled 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 fixed costs which can provide a significant competitive advantage to the company. In addition, we tested our method on the benchmarks of the literature on PVRP. Our method found three new best solutions and identified 8 new upper bounds which were not found previously in the literature.

The following future research avenues could be worth considering:

(i) Sophisticated optimisation techniques for the weeks and days planning models that consider distances between customers could be explored. This can be achieved by adapting column generation techniques.
(ii) A backtracking mechanism between the three optimisation phases could be implemented. This needs to improve the route construction by the reassignment of customers to weeks and days.
(iii) It is also worth highlighting that in many real-life applications, parameter' uncertainty is usually common and therefore ought not to be ignored. Concepts inspired from stochastic optimisation or/and fuzzy logic could be one way forward.

Acknowledgments

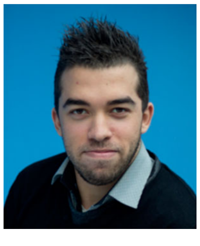
This paper is an extension of the preliminary study conducted by Messaoudi, Oulamara, and Rahmani (2019), we are grateful to the authors of Messaoudi, Oulamara, and Rahmani (2019), for providing us with the material (instances and results) of their study and allowing the successful completion of this publication. We are also grateful to the authors of Rodríguez-Martín, Salazar-González, and Yaman (2019) for providing us with detailed results of all instances and to the referees for their interesting comments that improved both the content as well as the presentation of the paper.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Notes on contributors

Ammar Oulamara is a professor in Computer Science at the University of Lorraine. He received M.Sc. in Computer Science from National Polytechnic Institute of Grenoble, a Ph.D. in Computer Science from Joseph Fourier University, Grenoble, in 2001, and Habilitation in Computer Science from the National Polytechnic Institute of Lorraine, Nancy, in 2009. His research interests lie mainly in production scheduling and logistics, combinatorial optimisation, methods applied to manufacturing systems. He has published papers in many Operational Research and Industrial Engineering journals.



Bilal Messaoudi is an optimisation scientist at Antsway with over six years of experience in Operations Research. He received a Ph.D. degree in Computer Science from Universitè de Lorraine, in the year of 2020. His main focus is on design and implementation of algorithms applied to real-world problems. He currently heads up optimisation engine development in Antsroute, a web-based vehicle routeing software.

Said Salhi is Emeritus Professor in Operational Research/Management Science at the Kent Business School (UK) where he was full professor for 16 years until January 2021. He is currently Editor in Chief for the Journal of the Operational Research Society (JORS). He was Head of the Management Science Group for over 8 years (2008-May 2017), founder and designer of the MSc in Logistics & Supply Chain Management. He was also the Director of the Centre on Logistics and Heuristic Optimisation (CLHO), which he founded in 2008, till January 2021. Prior to his appointment to Kent in 2005, Said was at the University of Birmingham in the School of Mathematics for 15 years and in his later years he acted as the Head of the Management Mathematics Group. Pr Salhi has edited 11 special issues including the recent OR handbook 'The Palgrave Handbook of Operations Research' published in Summer 2022 and authored a book on 'Heuristic Search: The Emerging Science of Problem Solving" (2017). He published over 130 papers in academic journals (excluding edited proceedings and book chapters). Pr. Salhi is a fellow of the OR Society (FORS), the IMA (FIMA), and CILT (FCILT), and an Executive member of NATCOR and a member of COPIOR. He has also been an associate editor of the IMA Journal of Management Mathematics since 2004 and sits in several editorial boards.

Data availability statement

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

References

Alegre, J., M. Laguna, and J. Pacheco. 2007. "Optimizing the Periodic Pick-Up of Raw Materials for a Manufacturer of Auto Parts." *European Journal of Operational Research* 179 (3): 736–746. doi:10.1016/j.ejor.2005.03.063.

An, Y.-J., Y.-D. Kim, B. Jeong, and S.-D. Kim. 2012. "Scheduling Healthcare Services in a Home Healthcare System." *Journal of the Operational Research Society* 63 (11): 1589–1599. doi:10.1057/jors.2011.153.

Archetti, C., E. Fernandez, and D. L. Huerta-Munoz. 2017. "The Flexible Periodic Vehicle Routing Problem." *Computers & Operations Research* 85: 58–70. doi:10.1016/j.cor.2017.03.008.

Archetti, C., E. Fernandez, and D. L. Huerta-Munoz. 2018. "A Two-Phase Solution Algorithm for the Flexible Periodic Vehicle Routing Problem." *Computers & Operations Research* 99: 27–37. doi:10.1016/j.cor.2018.05.021.

Baldacci, R., E. Bartolini, A. Mingozzi, and A. Valletta. 2011. "An Exact Algorithm for the Period Routing Problem." *Operations Research* 59 (1): 228–241. doi:10.1287/opre.1100.0875.

Bellmore, M., and G. L. Nemhauser. 1968. "The Traveling Salesman Problem: A Survey." *Operations Research* 16 (3): 538–558. doi:10.1287/opre.16.3.538.

Beltrami, E. J., and L. D. Bodin. 1974. "Networks and Vehicle Routing for Municipal Waste Collection." *Networks* 4 (1): 65–94. doi:10.1002/(ISSN)1097-0037.

Blakeley, F., B. Arguello, B. Cao, W. Hall, and J. Knolmajer. 2003. "Optimizing Periodic Maintenance Operations for Schindler Elevator Corporation." *Interfaces* 33 (1): 67–79. doi:10.1287/inte.33.1.67.12722.

Braekers, K., and A. A. Kovacs. 2016. "A Multi-Period Dial-a-Ride Problem with Driver Consistency." *Transportation Research Part B: Methodological* 94: 355–377. doi:10.1016/j.trb.2016.09.010.

Campbell, A. M., and J. H. Wilson. 2014. "Forty Years of Periodic Vehicle Routing." *Networks* 63 (3): 276–276. doi:10.1002/net.21544.

Campelo, P., F. Neves-Moreira, P. Amorim, and B. Almada-Lobo. 2019. "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. doi:10.1016/j.ejor.2018.07.030.

Cantu-Funes, R., M. A. Salazar-Aguilar, and V. Boyer. 2018. "Multi-Depot Periodic Vehicle Routing Problem with Due Dates and Time Windows." *Journal of the Operational Research Society* 69 (2): 296–306. doi:10.1057/s41274-017-0206-7.

Carotenuto, P., S. Giordani, S. Massari, and F. Vagaggini. 2018. "A Multi-Depot Periodic Vehicle Routing Model for Petrol Station Replenishment." In *Advanced Concepts, Methodologies and Technologies for Transportation and Logistics*, edited by Żak J., Y. Hadas, and R. Rossi, 421–437.

Christofides, N., and J. E. Beasley. 1984. "The Period Routing Problem." *Networks* 14 (2): 237–256. doi:10.1002/(ISSN)1097-0037.

Coene, S., A. Arnout, and F. C. R. Spieksma. 2010. "On a Periodic Vehicle Routing Problem." *Journal of the Operational Research Society* 61 (12): 1719–1728. doi:10.1057/jors.2009.154.

Cordeau, J.-F., M. Gendreau, and G. Laporte. 1997. "A Tabu Search Heuristic for Periodic and Multi-Depot Vehicle Routing Problems." *Networks: An International Journal* 30 (2): 105–119. doi:10.1002/(ISSN)1097-0037.

Fauske, M. F., C. Mannino, and P. Ventura. 2020. "Generalized Periodic Vehicle Routing and Maritime Surveillance." *Transportation Science* 54 (1): 164–183. doi:10.1287/trsc.2019.0899.

Feillet, D., T. Garaix, F. Lehuédé, O. Péton, and D. Quadri. 2014. "A New Consistent Vehicle Routing Problem for the Transportation of People with Disabilities." *Networks* 63 (3): 211–224. doi:10.1002/net.v63.3.

Francis, P., K. Smilowitz, and M. Tzur. 2006. "The Period Vehicle Routing Problem with Service Choice." *Transportation Science* 40 (4): 439–454. doi:10.1287/trsc.1050.0140.

Ghasemkhani, A., R. Tavakkoli-Moghaddam, Y. Rahimi, S. Shahnejat-Bushehri, and H. Tavakkoli-Moghaddam. 2021. "Integrated Production-Inventory-Routing Problem for Multi-Perishable Products under Uncertainty by Meta-Heuristic Algorithms." *International Journal of Production Research* 0 (0): 1–21.

Goeke, D., R. Roberti, and M. Schneider. 2019. "Exact and Heuristic Solution of the Consistent Vehicle-Routing Problem." *Transportation Science* 53 (4): 1023–1042. doi:10.1287/trsc.2018.0864.

Google. 2018. "Distance Matrix API." Accessed October 30, 2018. https://developers.google.com/maps/documentation/distance-matrix/start.

Groër, C., B. Golden, and E. Wasil. 2009. "The Consistent Vehicle Routing Problem." *Manufacturing & Service Operations Management* 11 (4): 630–643. doi:10.1287/msom.1080.0243.

Hemmelmayr, V. C., K. F. Doerner, and R. F. Hartl. 2009. "A Variable Neighborhood Search Heuristic for Periodic Routing Problems." *European Journal of Operational Research* 195 (3): 791–802. doi:10.1016/j.ejor.2007.08.048.

Holzapfel, A., A. Hubner, H. Kuhn, and M. G. Sternbeck. 2016. "Delivery Pattern and Transportation Planning in Grocery Retailing." *European Journal of Operational Research* 252 (1): 54–68. doi:10.1016/j.ejor.2015.12.036.

Hu, H., Y. Zhang, and L. Zhen. 2017. "A Two-Stage Decomposition Method on Fresh Product Distribution Problem." *International Journal of Production Research* 55 (16): 4729–4752. doi:10.1080/00207543.2017.1292062.

Kirkpatrick, S., C. Gelatt, and M. Vecchi. 1983. "Optimization by Simulated Annealing." *Science (New York, N.Y.)* 220 (4598): 671–680. doi:10.1126/science.220.4598.671.

Kovacs, A. A., B. L. Golden, R. F. Hartl, and S. N. Parragh. 2014. "The Generalized Consistent Vehicle Routing Problem." *Transportation Science* 49 (4): 796–816. doi:10.1287/trsc.2014.0529.

Kovacs, A. A., S. N. Parragh, and R. F. Hartl. 2014. "A Template-Based Adaptive Large Neighborhood Search for the Consistent Vehicle Routing Problem." *Networks* 63 (1): 60–81. doi:10.1002/net.21522.

Kulachenko, I. N., and P. A. Kononova. 2020. "A Hybrid Local Search Algorithm for the Consistent Periodic Vehicle Routing Problem." *Journal of Applied and Industrial Mathematics* 14 (2): 340–352. doi:10.1134/S199047892002012X.

Lin, S., and B. W. Kernighan. 1973. "An Effective Heuristic Algorithm for the Traveling-Salesman Problem." *Operations Research* 21 (2): 498–516. doi:10.1287/opre.21.2.498.

Liu, C.-H. 2016. "Solving the Bi-Objective Optimisation Problem with Periodic Delivery Operations Using a Lexicographic Method." *International Journal of Production Research* 54 (8): 2275–2283. doi:10.1080/00207543.2015.1070969.

Luo, Z., H. Qin, C. Che, and A. Lim. 2015. "On Service Consistency in Multi-Period Vehicle Routing." *European Journal of Operational Research* 243 (3): 731–744. doi:10.1016/j.ejor.2014.12.019.

Mancini, S., M. Gansterer, and R. F. Hartl. 2021. "The Collaborative Consistent Vehicle Routing Problem with Workload Balance." *European Journal of Operational Research* 293 (3): 955–965. doi:10.1016/j.ejor.2020.12.064.

Matos, A. C., and R. C. Oliveira. 2004. "An Experimental Study of the Ant Colony System for the Period Vehicle Routing Problem." In *Ant Colony Optimization and Swarm Intelligence*, 286–293. Berlin, Heidelberg: Springer.

Messaoudi, B., A. Oulamara, and N. Rahmani. 2019. "Multiple Periods Vehicle Routing Problems: A Case Study." In *Evolutionary Computation in Combinatorial Optimization*, edited by A. Liefooghe and L. Paquete, 83–98. Cham: Springer International Publishing – LNCS (11452).

Mor, A., and M. G. Speranza. 2022. "Vehicle Routing Problems Over Time: A Survey." *Annals of Operations Research* 314: 255–275.

Mourgaya, M., and F. Vanderbeck. 2007. "Column Generation Based Heuristic for Tactical Planning in Multi-Period Vehicle Routing." *European Journal of Operational Research* 183 (3): 1028–1041. doi:10.1016/j.ejor.2006.02.030.

Nguyen, P. K., T. G. Crainic, and M. Toulouse. 2014. "A Hybrid Generational Genetic Algorithm for the Periodic Vehicle Routing Problem with Time Windows." *Journal of Heuristics* 20 (4): 383–416. doi:10.1007/s10732-014-9244-3.

Rodríguez-Martín, I., J.-J. Salazar-González, and H. Yaman. 2019. "The Periodic Vehicle Routing Problem with Driver Consistency." *European Journal of Operational Research* 273 (2): 575–584. doi:10.1016/j.ejor.2018.08.032.

Ronen, D., and C. A. Goodhart. 2008. "Tactical Store Delivery Planning." *Journal of the Operational Research Society* 59 (8): 1047–1054. doi:10.1057/palgrave.jors.2602449.

Ropke, S., and D. Pisinger. 2006a. "An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows." *Transportation Science* 40 (4): 455–472. doi:10.1287/trsc.1050.0135.

Ropke, S., and D. Pisinger. 2006b. "A Unified Heuristic for a Large Class of Vehicle Routing Problems with Backhauls." *European Journal of Operational Research* 171 (3): 750–775. doi:10.1016/j.ejor.2004.09.004.

Rothenbächer, A.-K. 2019. "Branch-and-Price-and-Cut for the Periodic Vehicle Routing Problem with Flexible Schedule Structures." *Transportation Science* 53 (3): 850–866. doi:10.1287/trsc.2018.0855.

Russell, R. A., and D. Gribbin. 1991. "A Multiphase Approach to the Period Routing Problem." *Networks*21 (7): 747–765. doi:10.1002/(ISSN)1097-0037.

Russell, R., and W. Igo. 1979. "An Assignment Routing Problem." *Networks* 9: 1–17. doi:10.1002/(ISSN)1097-0037.

Russell, D., R. J. Rosati, P. Rosenfeld, and J. M. Marren. 2011. "Continuity in Home Health Care: Is Consistency in Nursing Personnel Associated with Better Patient Outcomes?." *Journal for Healthcare Quality* 33 (6): 33–39. doi:10.1111/j.1945-1474.2011.00131.x.

Salhi, S. 2017. *Heuristic Search: The Emerging Science of Problem Solving*. Springer.

Schubert, E., and P. J. Rousseeuw. 2019. "Faster k-Medoids Clustering: Improving the PAM, CLARA, and CLARANS Algorithms." In *International Conference on Similarity Search and Applications*, 171–187. Springer.

Smilowitz, K., M. Nowak, and T. Jiang. 2013. "Workforce Management in Periodic Delivery Operations." *Transportation Science* 47 (2): 214–230. doi:10.1287/trsc.1120.0407.

Stavropoulou, F. 2022. "The Consistent Vehicle Routing Problem with Heterogeneous Fleet." *Computers & Operations Research* 140: 105644. doi:10.1016/j.cor.2021.105644.

Stavropoulou, F., P. Repoussis, and C. Tarantilis. 2019. "The Vehicle Routing Problem with Profits and Consistency Constraints." *European Journal of Operational Research* 274 (1): 340–356. doi:10.1016/j.ejor.2018.09.046.

Tarantilis, C. D., F. Stavropoulou, and P. P. Repoussis. 2012. "A Template-Based Tabu Search Algorithm for the Consistent Vehicle Routing Problem." *Expert Systems with Applications* 39 (4): 4233–4239. doi:10.1016/j.eswa.2011.09.111.

Teemu, N., K. Jari, N. Harri, and B. Olli. 2006. "Improved Route Planning and Scheduling of Waste Collection and Transport." *Expert Systems with Applications* 30 (2): 223–232. doi:10.1016/j.eswa.2005.07.009.

Toth, P., and D. Vigo. 2014. *Vehicle Routing: Problems, Methods, and Applications*. MOS-SIAM Series on Optimization.

Vidal, T., T. G. Crainic, M. Gendreau, N. Lahrichi, and W. Rei. 2012. "A Hybrid Genetic Algorithm for Multidepot and Periodic Vehicle Routing Problems." *Operations Research* 60 (3): 611–624. doi:10.1287/opre.1120.1048.

Vidal, T., G. Laporte, and P. Matl. 2020. "A Concise Guide to Existing and Emerging Vehicle Routing Problem Variants." *European Journal of Operational Research* 286 (2): 401–416. doi:10.1016/j.ejor.2019.10.010.

Wang, Y., L. Wang, G. Chen, Z. Cai, Y. Zhou, and L. Xing. 2020. "An Improved Ant Colony Optimization Algorithm to the Periodic Vehicle Routing Problem with Time Window and Service Choice." *Swarm and Evolutionary Computation* 55: 100675. doi:10.1016/j.swevo.2020.100675.

Xu, Z., and Y. Cai. 2018. "Variable Neighborhood Search for Consistent Vehicle Routing Problem." *Expert Systems with Applications* 113: 66–76. doi:10.1016/j.eswa.2018.07.007.

Zajac, S. 2017. "An Adaptive Large Neighborhood Search for the Periodic Vehicle Routing Problem." In *Computational Logistics*, edited by T. Bektaş, S. Coniglio, A. Martinez-Sykora, and S. Voß, 34–48. Springer International Publishing.