A set partitioning heuristic for the home health care routing and scheduling problem

Florian Grenouilleau*, Antoine Legrain, Nadia Lahrichi, Louis-Martin Rousseau

[a] *École Polytechnique de Montréal, 2900 Boulevard Edouard-Montpetit, H3T 1J4, Montréal, QC, Canada*

Abstract

The home health care routing and scheduling problem comprises the assignment and routing of a set of home care visits over the duration of a week. These services allow patients to remain in their own homes, thereby reducing governmental costs by decentralizing the care. In this work, we present a set partitioning heuristic which takes into account most of the industry's practical constraints. The developed method is based on a set partitioning formulation and a large neighborhood search (LNS) framework. The algorithm solves a linear relaxation of a set partitioning model using the columns generated by the large neighborhood search. A constructive heuristic is then called to build an integer solution. This project is joint work with Alayacare, a start-up sited in Montral (Canada) developing an operations management platform for home health care agencies. They provide their clients with a flexible optimization module that solves real-life instances in no more than 10 minutes. Based on their real instances, the proposed method is able to

*Corresponding author

Email addresses: florian.grenouilleau@cirrelt.net (Florian Grenouilleau), antoine.legrain@polymtl.ca (Antoine Legrain), nadia.lahrichi@polymtl.ca (Nadia Lahrichi), louis-martin.rousseau@polymtl.ca (Louis-Martin Rousseau)

provide a reduction in travel time by 37% and an increase by more than 16% in continuity of care. We also provide a public benchmark for this problem.
Keywords: OR in health services, Routing, Scheduling, Set partitioning, Large Neighborhood Search

1. Introduction

Home health care services improve patients' quality of life by helping them remain independent and in their own homes, often surrounded by family and friends, while maintaining their regular habits. From a governmental point of view, home care services decrease hospital congestion by freeing up hospital beds, which also results in reducing costs for these institutions (Macintyre et al., 2002). In 2012, in Canada, more than 2.2 million people received home care services (Maire & Amanda, 2014). These services are various: from personal support (bathing, dressing, housekeeping) to more specific tasks such as insulin injection or wound care. Due to the variety of tasks required, different medical specialties and skills are needed (e.g., personal social worker or nurse).

In this paper, we investigate the home health care routing and scheduling problem (HHCRSP) within a practical context. This problem is interested in determining the assignment of a set of home visits to a set of caregivers over the course of a week and the routing of these caregivers' workdays. The HHCRSP can be described as a multi-depot vehicle routing problem (MDVRP) with time windows and time-dependent travel issues. Moreover, the home care context adds constraints focusing on the caregivers' skills and the patients' requirements (both mandatory and optional), as well as the management of the caregivers' work time contracts. Finally, the HHCRSP has a major concern which is the continuity of care, corresponding to the upkeep of a strong patient-caregiver relationship.

In this work, we present a set partitioning heuristic (SPH) to address the weekly version of the HHCRSP. This method is based on the heuristic concentration principle (Rosing & ReVelle, 1997). The goal of our SPH is to solve a set partitioning formulation of the HHCRSP using the columns (feasible routes) generated by a Large Neighborhood Search (LNS) (Shaw, 1998). Due to the necessity to produce high quality solutions in a small computational time, the SPH solves a linear relaxation of the set partitioning formulation and a constructive heuristic is then applied to build an integer solution based on the solution found. This paper presents three major contributions. First, the proposed method takes into account a larger set of practical constraints and solves instances covering up to 430 visits, over the course of a week, in less than 10 minutes. Second, we propose a relaxed heuristic concentration approach that combines the global perspective of a mathematical program with the efficiency of a heuristic approach. Finally, we propose new LNS' operators, specifically designed for the HHCRSP, which permit the extension of the search space to find new and improved solutions.

To assess the quality of the proposed method, we have evaluated it performance against a classic LNS approach. Furthermore, as our research's context differs from existing benchmarks, and that reproducibility is of major importance in research, we provide and make public a set of realistic generated instances. We hope that this benchmark will help to homogenize the research about the HHCRSP and help the future authors to compare their methods.

The paper is organized as follows. Section 2 presents the literature review on the HHCRSP. Section 3 details the problem and its formulation. Section 4 describes our approach and Section 5 shows the computational results on generated and real instances. Finally, conclusions are drawn in Section 6.

2. Literature review

From our knowledge, the routing and scheduling optimization in the home health care context is a 20 years old problem (Begur et al., 1997; Cheng & Rich, 1998). According to the existing literature, we observe that no standard version of the problem exists. Authors use different constraints and objectives. This plurality, usually due to the authors's country's home care management, makes it difficult to compare the existing methods.

The HHCRSP was originally solved on a daily planning horizon. Then, it has evolved to integrate more practical constraints such as the maximization of the patients and caregivers' preferences (Braekers et al., 2016), the balance of the workload (Bertels & Fahle, 2006), visits incompatibilities (Manerba & Mansini, 2016), shared visits (Frifita et al., 2017), multiple modes of transportation (Hiermann et al., 2015) or even the time-dependent travel time (Rest & Hirsch, 2016). Thereafter, the HHCRSP has been extended to a weekly horizon that allows for better coping with the reality of some constraints, such as the patients' care plan and/or the continuity of care. Some methods using branch-and-price (Gamst & Jensen, 2012), branch-and-price-and-cut (Trautsamwieser & Hirsch, 2014), cardinality constraints (Cappanera et al., 2017) integer linear based method (Borsani et al., 2006; Torres-Ramos et al., 2014) have been proposed, but the complexity of the problem leads to scalability issues.

To deal with these issues, methods based on heuristics or meta-heuristics have been developed using frameworks such as large neighborhood search (Di Gaspero & Urli, 2014), memetic algorithm (Decerle et al., 2018), ant colony optimization (Zhang et al., 2018), two-phase algorithm (Yalçındağ et al., 2016), or harmony search (Lin et al., 2018). In Nickel et al. (2012), the problem is split in two: the *master problem*, which uses a constructive heuristic and an ALNS to build a feasible assignment of the visits, and the *operational problem*, which integrates the last minute changes (e.g., visit cancellation or sick caregiver) into the current schedule with an insertion heuristic and a tabu-search. Finally, Duque et al. (2015) propose a two-phase method based on a set partitioning formulation. The first phase produces pools of visit patterns and solves the patterns' assignment using Cplex. Then, the second phase improves the best patterns' assignment with a local search procedure that swaps patients' visits to reduce travel time and maximize patients and nurses' preferences.

For more references, we refer the reader to two comprehensive surveys published recently (Cissé et al., 2017; Fikar & Hirsch, 2017). For an overview on the multi-day HHCRSP (features and classic constraints), we refer the reader to Table 1. We highlight for each paper the main characteristics of the problem, and constraints are classified either as hard (H) or soft (S). Last column indicates the instances' availability. According to this table, we observe that we tackle a richer version of the problem than most of the existing literature. In the similar work of (Duque et al., 2015), instances are not available. We fill this gap by providing new benchmark to the community (http://dx.doi.org/10.17632/cbgt59hnhk.1).

	Features						Constraints				
Article	Travel cost	Workload Balance	Time-dependent	Patient's preferences	Caregiver's preferences	Continuity of care	Work time contracts	Time windows	Qualifications	Unscheduled visits	Instances available
(Gamst & Jensen, 2012)	S				S			S	H	H	
(Trautsamwieser & Hirsch, 2014)							H	H	H	H	✓
(Cappanera et al., 2017)	S	S				H	H		H	H	
(Borsani et al., 2006)				S		S			H	S	
(Torres-Ramos et al., 2014)	S						H	H	H	H	
(Di Gaspero & Urli, 2014)	S						H	H	H	S	
(Zhang et al., 2018)	S						H	H	H		
(Yalçındağ et al., 2016)	S	S				H			H	H	
(Lin et al., 2018)	S				H		H		H	H	✓
(Nickel et al., 2012)	S					S	S	H	H	S	
(Duque et al., 2015)				S		H	H	S	S	H	
Our approach	S		H	S	S	S	S	H	H	S	✓

Table 1: Characteristics of the multi-day HHCRSP literature

3. Problem definition

The home health care routing and scheduling problem can be described as a "multi-attribute" vehicle routing problem as it considers many features (e.g., patients' requirements, caregivers' skills, time-dependent travel times, and contracted working hours). We define the sets P of patients and C of caregivers. The objective is to determine patient-to-caregiver assignments and build the caregivers' routes over the horizon of H days ($H = 7$ in our context) according to the required number of visits of the patients. The caregivers' assignments must take into account patients' mandatory (e.g., specialty) and optional (e.g., language) requirements and caregivers' skills and characteristics (e.g., gender). (Note that in the remainder of the paper, caregivers characteristics will be included in the set of skills for the sake of simplicity). The routing part of the problem must cope with patients' availability (days and time windows) and caregivers' work shifts. Caregivers' work contracts (i.e., minimum and maximum amount of working time per day and week) have to be managed as well. Finally, the impact of traffic delays on travel time are taken into account, through a time-dependent distance matrix.

For each patient $p \in P$, we define a number n_p of required visits of duration dur_p, a subset $D_p \subseteq [1, ..., H]$ of available days and a hard time-window $[e_p^d, l_p^d]$ for each available day $d \in D_p$. Moreover, we also define two lists M_p and O_p that respectively contain the mandatory and optional requirements. The optional requirements could be described as patient's preferences about, for example, the gender or the language spoken by the assigned caregiver. For each caregiver $c \in C$, we similarly define a list E_c of skills, a soft minimum $\underline{w}_c^w$ and maximum $\overline{w}_c^w$, working hours over the week, and a subset $D_c \subseteq [1, ..., H]$ of workdays. Each of these workdays d also has a time-window $[a_c^d, b_c^d]$ and a soft minimum $\underline{w}_c^d$ and maximum $\overline{w}_c^d$ of working hours. Every patient and caregiver have their home location (respectively l_p and l_c) that belongs to a set L of possible zip codes. Finally, the continuity of care measures the strength of a patient-caregiver relationship with a score $CC_{p,c}$. This is based on the number of times that the caregiver c has been assigned to the past visits of patient p.

We propose to formulate the HHCRSP as a set partitioning problem (SPP)

that aims at selecting the best routes for each caregiver c among a set Ω of daily feasible caregivers' routes. We also define the subsets $\Omega_d \subset \Omega$ and $\Omega_c \subset \Omega$ that correspond to the routes associated respectively to day d and caregiver c. Each route $\omega \in \Omega$ is defined by a set of visited patients. Implicitly, each route ensures the respect of the patients' mandatory requirements, the caregivers' skills, the time-windows and the time-dependent travel times for all the visited patients. To each route, we compute a cost based on: 1) the number of missing optional requirements (defined as the number of optional requirements of patient (O_p) minus the intersection between sets O_p and E_c, where c is the caregiver assigned to the route ω); 2) a travel time tt_ω; 3) a score for continuity of care, and 4) a length len_ω penalty (this captures the minimum or maximum daily working hours for each caregiver).

The score for continuity of care f_1 is given by :

$$f_1(CC_{p,c}) = \begin{cases} 1 & \text{if } \ CC_{p,c} = 0 \\ \frac{2}{3} & \text{if } \ 1 \leq CC_{p,c} \leq 2 \\ \frac{1}{3} & \text{otherwise} \end{cases}$$

and, the working hours penalty function f_2 is described as follows:

$$f_2(len_\omega) = \max(0,\ \underline{w}_d - len_\omega,\ len_\omega - \overline{w}_d)$$

The cost c_ω of each route $\omega \in \Omega$ is therefore defined as a weighted sum :

$$c_\omega = \gamma_1 . \sum_{p \in P} a_{\omega,p}(|O_p| - |O_p \cap E_c|) + \gamma_2 . tt_\omega + \gamma_3 . \sum_{p \in P} a_{\omega,p} . f_1(CC_{p,c}) + \gamma_4 . f_2(len_\omega),$$

where $a_{\omega,p}$ equals to 1 if the route ω visits patient p and $\gamma_1, ... \gamma_4$ correspond to the weight of each soft constraint and capture the significance of each objective function's component.

The decision variables of the problem are given by :

- x_ω which equals 1 if the route ω is selected, 0 otherwise;

- o_c which measures the weekly overtime for caregiver c;

- u_c which also measures the weekly idle time for caregiver c. This corresponds to the number of working hours not used in the caregiver's week, i.e, the difference between the minimal amount of working hours and the actual amount of scheduled hours;

- z_p which counts the number of unscheduled visits for patient p.

The corresponding SPP formulation is defined as follows:

$$(SPP) : \min \quad \sum_{\omega \in \Omega} c_\omega x_\omega + \beta_1 . \sum_{c \in C} (o_c + u_c) + \beta_2 . \sum_{p \in P} z_p \tag{1}$$

$$\text{subject to:} \sum_{\omega \in \Omega_d} a_{\omega,p} x_\omega \leq 1 \qquad \forall p \in P, d \in D_p \tag{2}$$

$$\sum_{\omega \in \Omega} a_{\omega,p} x_\omega + z_p = n_p \qquad \forall p \in P \tag{3}$$

$$\sum_{\omega \in \Omega_d \cap \Omega_c} x_\omega \leq 1 \qquad \forall c \in C, d \in D_c \tag{4}$$

$$\sum_{\omega \in \Omega} len_\omega x_\omega + u_c \geq \underline{w}_c^w \qquad \forall c \in C \tag{5}$$

$$\sum_{\omega \in \Omega} len_\omega x_\omega - o_c \leq \overline{w}_c^w \qquad \forall c \in C \tag{6}$$

$$x_\omega \in \{0, 1\} \qquad \forall \omega \in \Omega \tag{7}$$

$$z_p \geq 0 \qquad \forall p \in P \tag{8}$$

$$o_c, u_c \geq 0 \qquad \forall c \in C \tag{9}$$

The objective function (1) corresponds to a weighted sum of costs associated, respectively, to the routes, the weekly caregivers' overtimes and idle time and the unscheduled visits. Weights β_1 and β_2 capture the significance of the second and third objective function's components. Constraints (2) ensure that patient p is visited a maximum of once per day, Constraints (3) count the number of unscheduled visits per patient. Then, Constraints (4) ensure that no more than one route per day is assigned to each caregiver. Finally, Constraints (5) – (6) measure, respectively, the weekly idle time and overtime. The domains of the variables are defined by Constraints (7) – (9).

One should note that required number of visits for each patient is a soft constraint. In practice, agencies may outsource the visits they can not provide. Finally, we consider single visits per day. Our model is easy to extend to multiple visits.

4. Solution Method

In this section, we present the set partitioning heuristic (SPH). The proposed SPH is a matheuristic based on the resolution of the SPP presented in the section 3. This method is based on the heuristic concentration principle (Rosing & ReVelle, 1997). The aim of the heuristic concentration is to keep the best solutions found by a heuristic procedure and then use a set partitioning that combines parts of these solutions to create a better one. This combination of heuristic and exact approaches have already been used for the VRPTW (Muter et al., 2010; Mendoza et al., 2016). In our method, the possible SPP's routes are found using a Large Neighborhood Search (LNS).

4.1. Set Partitioning Resolution

The resolution of a set partitioning model is difficult and can be computationally expensive ((Gamst & Jensen, 2012), (Trautsamwieser & Hirsch, 2014)). The industrial context of this project however required relatively short solution time, basically less than 10 minutes. To address this challenge, we solve a relaxation of the proposed SPP model ($Relaxed_{SPP}$) and reconstruct the integer solution using a constructive heuristic ($Heur_{SPP}$). The $Relaxed_{SPP}$ corresponds to the resolution of the proposed SPP while relaxing the integrity of the decision variables x_ω, generating the fractional variables $\bar{x}_\omega$. After the resolution of the $Relaxed_{SPP}$, the $Heur_{SPP}$ procedure is called to build an integer solution according to the resultant relaxed values $\bar{x}_\omega$. An overview of the method is given by the Algorithm 1.

Algorithm 1: $Heur_{SPP}$

Create the list $L^{\bar{\Omega}}$, copy of the routes in Ω, sorted in decreasing order of the values $\bar{x}_\omega$ from the last $Relaxed_{SPP}$
Empty solution s
forall *route* ω *in* $L^{\bar{\Omega}}$ **do**
    forall *patient visit* v *in* ω*'s visit list* **do**
        if *The patient of the visit* v *has all his/her visits scheduled in* s **then**
            remove v from ω's visit list
    end forall
    if ω*'s visit list is not empty* **then**
        Reschedule ω with the remaining visits
        Insert the route ω in s
end forall
if *The solution* s *is better than the best found solution* **then**
    Update the best found solution with s
end if

As presented in Algorithm 1, $Heur_{SPP}$ creates an integer solution based on the fractional one found by the previous $Relaxed_{SPP}$. First, we sort the existing routes $\overline{\Omega}$ by their value $\bar{x}_\omega$ and store them in the list $L^{\bar{\Omega}}$. Then a constructive method is applied starting from an empty solution s. Iteratively, we select the next route r in the list and remove from r the patients for which, all the visits are already scheduled in s. If the route is not empty, we determine the visit times for the remaining patients in the route r and insert r in s. At the end of the algorithm, if the built solution s is better than the best one found during the previous SPH's iterations, we update the best solution with s.

4.2. LNS-based route generation

In order to quickly generate a set of high-quality routes, a Large Neighborhood Search (LNS) method is developed. On top of the continuous generation of feasible caregivers' routes, the LNS also allows to gradually improve the best found solution by using a set of classic and problem-specific operators. The found routes are then used to solve the proposed $Relaxed_{SPP}$.

The LNS (Shaw, 1998) is a meta-heuristic using the $ruin-and-recreate$ principle (Schrimpf et al., 2000). This method, starting from an initial solution, iteratively destroys a part of the current solution, then repairs it in order to improve its quality. The current and best solutions are then updated if necessary. A full description of the LNS can be found in Gendreau & Potvin (2010). In the following subsections, we present the main components of our LNS. In particular, we present the specific operators that we have developed to cope with some HHCRSP's difficulties.

4.2.1. Initial solution

In order to create the initial feasible solution, a lowest-cost insertion method is used. We first sort the patients in decreasing order of their visits' durations, then, following this order, we try to insert the patient's visits at the lowest cost. The patients with unscheduled visits are stored in a list until the first LNS iteration. In our context, the possibility to have unscheduled visits ensures to always have a feasible solution (in the worst case, all the visits are unscheduled and all the caregivers' routes are empty).

4.2.2. Classic LNS operators

In our LNS' implementation, we use a mix of both classic and new operators for the destruction/repair operations. For the destruction procedure, the classic operators are *WorstRemoval*, *RandomRemoval* from Ropke & Pisinger (2006) and the *RelatedRemoval* from Shaw (1998). For the repair procedure, the classic operators are the *Greedy Heuristic*, *regret*-2 and *regret*-3 from Ropke & Pisinger (2006).

4.2.3. New LNS operators

In order to focus the search on some difficult aspects of the problem, some problem-specific destroy and repair operators have been implemented in the LNS.

New destroy operators. Let us recall that q, the number of destroyed visits, is randomly selected at each LNS' iteration. The developed destroy operators are as follows :

I The *ServiceRemoval* operator randomly selects a patient and removes all his/her scheduled visits. This process is repeated until at least q visits are removed. This new operator permits a reset of the assigned visit days of the patient and potentially creates a new pattern of visits during the repair part.

II The *FlexibleAvailRemoval* operator deletes from the current schedule the patients with the highest flexibility (i.e., highest value of $\frac{|D_p|}{n_p}$). Iteratively, the most flexible patient is selected and all its scheduled visits are removed from the current schedule. The patients list is scanned this way until q visits are removed.

III The *DualRemoval* operator uses the dual values from the last $Relaxed_{SPP}$ resolution. Based on constraints (3), this operator sorts the patients in decreasing order of their dual values, then iteratively selects the patient at the top of the list (lowest dual value), and removes his/her visits. The process is repeated until q visits are removed like the other destroy operators.

New repair operators. For the proposed LNS, two new repair operators have been created :

I The *RandomService* operator randomly chooses one of the patients for which some visits are not scheduled. A lowest-cost insertion logic is used to schedule his/her visits over the horizon. This process is repeated until every patient with missing visits has been tested.

II The *DualRepair* operator focuses on the patients with the highest dual values. It sorts the patient in decreasing order of their dual values, based on constraints (3) of the last $Relaxed_{SPP}$'s resolution. Then, the operator follows this ordered list and tries to schedule as many visits as possible for each patient using again a lowest-cost insertion logic.

Due to the fact that the dual values come from the $Relaxed_{SPP}$, the dual operators (*DualRemoval*, *DualRepair*) can't be used in the first LNS's segment (first 1,000 iterations). They are introduced in the operators lists at the end of the first $Relaxed_{SPP}$. Note that the dual values remain unchanged until the following $Relaxed_{SPP}$ is solved.

4.2.4. Range of destruction

The number q of visits destroyed at each iteration is randomly drawn in a range $[min\_percent, max\_percent] * Sched_s$ where $Sched_s$ is the number of visits scheduled in the impacted solution s.

4.2.5. Solution Analysis

After the destroy and repair procedures, the created solution is analyzed to decide if its quality is good enough to be kept as a best or current solution. Three cases may occur in this context :

1. the new solution is better than the best found, the LNS updates the best and current solutions with the new one;
2. the new solution is better than the current solution, only the current solution is updated;
3. the new solution is worse than the current solution, a simulated annealing accept criterion is then used to either accept or refuse it.

This simulated annealing accept criterion (Kirkpatrick et al., 1983) accepts the new solution with a probability $e^{-\frac{f(s_{new})-f(s_{cur})}{T}}$ where $f(s_{new})$ and $f(s_{cur})$ are respectively the value of the new and current solutions. The value T is the current temperature of the problem which decreases at each simulated annealing call, according to the relation subscript $T_{n+1} = T_n \times c$ where $0 < c < 1$ is the decrease coefficient. According to Ropke & Pisinger (2006), the decrease coefficient c and the initial temperature T_0 are set to 0.99975 and $1.05 \times f(s_0)$, respectively, where s_0 is the initial solution.

4.2.6. Termination criterion

The SPH ends when reaching either a maximum number of LNS' iterations or a maximum computational time.

4.2.7. Management of the time-dependent travel time

In order to adapt to practical settings, we include the time-dependent travel times between the patients' locations. In our implementation, we use a dynamic computation of the time-dependent travel times based on the algorithm described by Ichoua et al. (2003). This algorithm computes the traffic-dependent travel times between two locations according to the departure time. It is based on a stepwise speed functions; the adjustment of speed between two periods of time ensures the respect of the FIFO principle.

4.3. Methods overview

Now that we have presented our method's two main components (SPP's resolution and LNS-based route generation), we can now give the overview of our set partitioning heuristic (SPH). A SPH's description is given by the Algorithm 2. The first part of the algorithm is based on the LNS' procedure described earlier (initial solution, destruction, repair, analysis). Then, at the end of each segment (i.e., a block of 1,000 iterations in our case), a sub-procedure is called. This procedure solves the $Relaxed_{SPP}$ and applies the $Heur_{SPP}$ presented in the subsection 4.1.

Algorithm 2: SPH

Find an initial feasible solution ;
while *No termination criteria met* **do**
  for *A segment of 1,000 iterations* **do**
    $s \leftarrow currentSolution$;
    Select and apply a destroy operator on s ;
    Select and apply a repair operator on s ;
    Analyze the solution s ;
  end for
  Solve $Relaxed_{SPP}$;
  Apply $Heur_{SPP}$;
end while
Return the best found solution ;

5. Computational Results

This section presenting some computational experiments is divided into two parts. The first set of experiments assesses the suitability of the overall SPH, by studying the effectiveness of the new operators with respect to the classical ones and examining the impact of the proposed set partitioning resolution ($Relaxed_{SPP}$ then $Heur_{SPP}$). In the second part, we analyze the performance of SPH on real instances provided by our industrial partner. The proposed algorithm has been implemented in C++ and all the tests are run on an Intel(R) Xeon(R) (duo core) X5675 3.07GHz, with 96 GB RAM and running on Linux operating system. We use the solver CPLEX, version 12.6.1. to solve $Relaxed_{SPP}$. All the experiments are run on a single thread. The termination criterion are set to 10 minutes and 10^5 LNS' iterations. The *min_percent* and *max_percent* have been respectively set to 2% and 5%. Finally, the weights (γ_1,γ_2,γ_3,γ_4,β_1,β_2,) have been fixed after preliminary evaluations in collaboration with Alayacare (refer to Appendix, Table 9 for the values.

5.1. Experiments on generated-instances

In order to test the proposed SPH, we have based our analysis on a benchmark of 60 instances : three sets (Small, Medium, Large) of 20 instances corresponding to the different problem's sizes that must be solved by the algorithm. An overview of the instances' characteristics is given in the table 2. Instances can be downloaded from: http://dx.doi.org/10.17632/cbgt59hnhk.1

Instance	Patient	Visits	Caregiver	Workdays
Small	40	120	5	25
Medium	80	225	10	45
Large	150	430	20	90

Table 2: Characteristics of the generated instances

These sets have been randomly generated based on real instances' characteristics provided by our industrial partner and each value has a predefined range. The instances' generation is based on 5 different requirements/skills, 141 possible locations, and several parameters described in Tables 3.

Parameter	Name	Minimum	Maximum
n_p	Number of visits	1	7
dur_p	Duration of visits (in min)	40	60
$\|M_p\|$	Mandatory requirements	1	2
$\|O_p\|$	Optional requirements	0	2
$\frac{l_p^d - e_p^d}{dur_p}$	Time-window's size	2	4
$\underline{w}_c^w$	Minimum week working hours (in min)	0	600
$\overline{w}_c^w$	Maximum week working hours (in min)	1200	2400
$b_c^d - a_c^d$	time-window's size (in min)	420	720
$\underline{w}_c^d$	Minimum day working hours (in min)	0	$30\% \cdot (b_c^d - a_c^d)$
$\overline{w}_c^d$	Maximum day working hours (in min)	$80\% \cdot (b_c^d - a_c^d)$	$100\% \cdot (b_c^d - a_c^d)$
$\|E_c\|$	Skills list	2	3

Table 3: Employees' parameters for the generated instances

In order to observe the impact of the proposed operators, we define 2 groups of operators :

- CL : The classic operators with $WorstRemoval$, $RandomRemoval$, $RelatedRemoval$ for the destroy part and *Greedy Heuristic*, *regret-2* and *regret-3* for the repair ones.
- NW : The new operators : $ServiceRemoval$, $FlexibleAvailRemoval$ and $DualRemoval$ for the destroy operators, $RandomService$ and $DualRepair$ for the repair ones. These operators necessitate the resolution of the $Relaxed_{SPP}$.

Moreover, to test the impact of the $Heur_{SPP}$, we distinguish the use or not of this algorithm.

For this analysis, 10 runs of each instance have been computed for three different scenarios (CL, CL + NW and CL + NW + $Heur_{SPP}$). The presented results are based on the average of the best found solutions' costs over the 10 runs. Figures 1 and 2 present the comparison of the three scenarios. The values correspond to the gap between each scenario's value and the value of the CL one. According to these results, we can observe that, on average, the new operators ($CL + NW$ scenario), by extending the search space, find better solutions and reduce the solutions' cost for the small, medium and large instances by respectively 7.63%, 10.06% and 2.34% (see tables 5, 6 and 7 in Appendix). The reduced improvements produced by the new operators on the large instances could be due to the reduced number of iterations done (see table 8 in Appendix). This reduction of the number of iteration (32211 for CL to 24101 for $CL + NW$) is caused by the time spent in the resolution of $Relaxed_{SPP}$ at each end of segment. Furthermore, we can observe that the $Heur_{SPP}$ ($CL + NW + Heur_{SPP}$ scenario) is able to find the best solutions for all instances : the improvements for the three instances' sets are respectively 13.76%, 20.82% and 14.39%. According to these observations, we'll keep the $CL + NW + Heur_{SPP}$ scenario for the real instances' resolution.

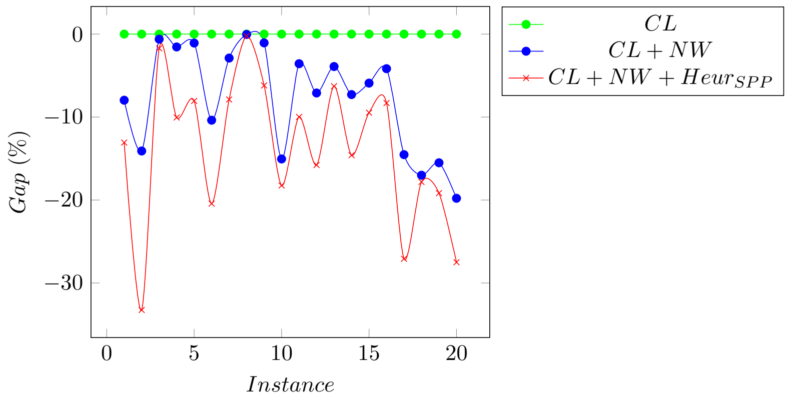

Figure 1: Comparison of the cost for the small instances

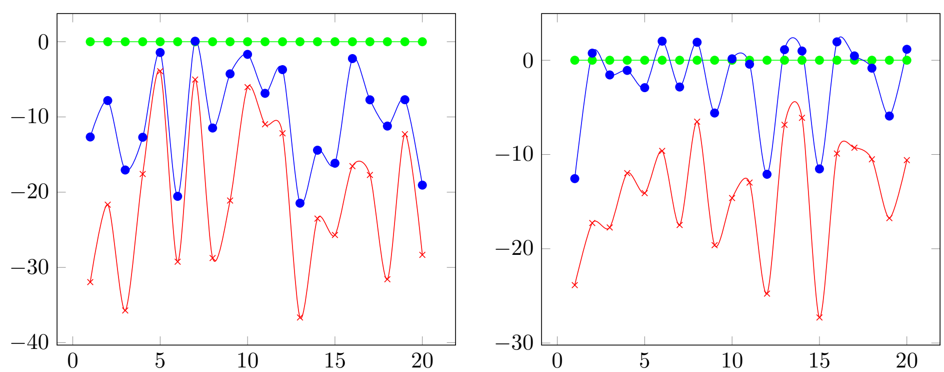

Figure 2: Comparison of the cost for the medium (left) and large (right) instances

5.2. Real-World Instances

In this section, we describe the tests performed on instances from one of Alayacare's clients. For the studied client, the objective was to analyze the improvements both in terms of travel time and continuity of care provided by the proposed method. In these experiments, 4 instances representing 4 different weeks have been used. These instances are described as $P\_V\_C\_R$ where P is the number of patients, V the number of visits, C the number of caregivers and R the number of routes (number of workdays). For these instances, the chosen patients were homogeneous, so the same requirements were needed. The available days correspond to actual patients' visits' days (i.e. $|D_p| = n_p$ for each patient). The patients' time windows were designed around their actual visit times. For the employees, their workdays, work time contracts and time windows were given by the client.The figure 3 presents the distribution of the number of visits per patient for the real instances. According to this figure, the majority of patients only need 1 or 2 visits per week.

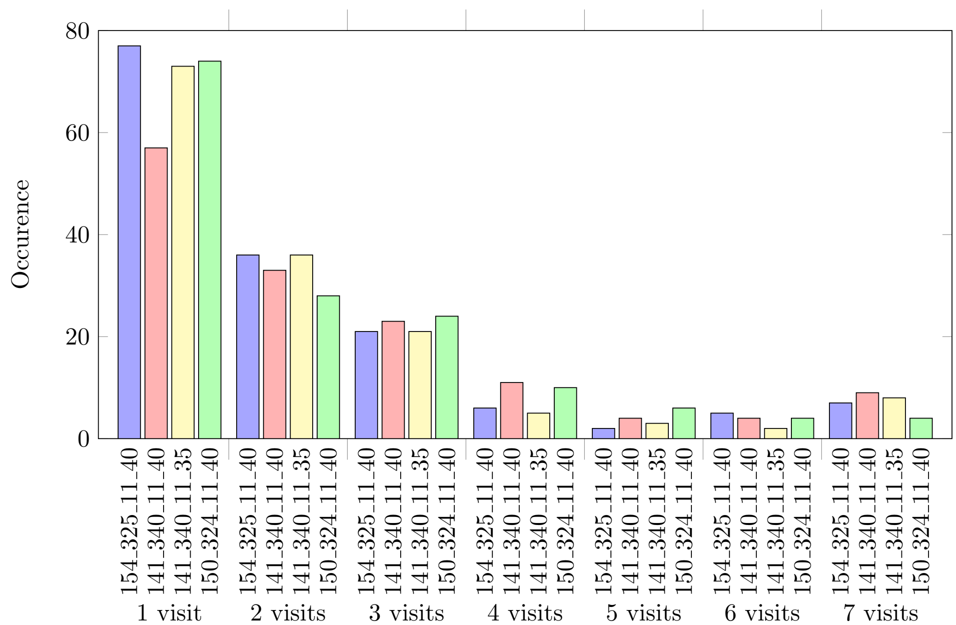

Figure 3: Distribution of the number of visit per patient

A comparison of Alayacare's current client solutions and our SPH's solutions on these 4 instances is presented in Table 4. According to the client's will, we focus here on two major indicators, the total travel time (TT) and the continuity of care (CC, i.e., the percentage of scheduled visits for which the patient p and the caregiver c have $CC_{p,c} \neq 0$).

	Current solution		***SPH*'s solution**		Δ	
Instance	*TT*	*CC*	*TT*	*CC*	*TT*	*CC*
154_325_11_40	4361.16	60%	**2431.62**	**75.94%**	-44.24%	+15.94%
141_340_11_40	4549.03	62.33%	**2833.18**	**79.05%**	-37.72%	+16.72%
148_311_11_35	3832.94	71.69%	**2571.29**	**85.98 %**	-32.92%	+14.29%
150_324_11_40	3686.57	64.43%	**2464.22**	**82.10%**	-33.16%	+17.67%
Mean	4107.43	64.61%	2575.08	80.77%	**-37.01%**	**+16.16%**

Table 4: Comparison of the actual solutions with those produced by our approach

According to the Table 4, our approach improves the solutions both in terms of travel time and continuity of care. On average, the proposed algorithm reduces the total travel time by 37.01% and increases the continuity of care by 16.16%. These results show that the use of such method by Alayacare's clients could lead to large improvement in term of costs reduction and quality of service.

6. Conclusions

The HHCRSP is a complex problem due to the simultaneous management of the assignment (requirements, skills, continuity of care) and routing (travel time, work time contracts, impact of the traffic) constraints. Nevertheless, we have proposed a set partitioning heuristic able to cope with all these requirements. The presented method is firstly based on a set partitioning formulation of the problem. The resolution of this set partitioning is done in two phases : the resolution of the relaxation ($Relaxed_{SPP}$) followed by a constructive heuristic ($Heur_{SPP}$). To populate the SPP's columns, we developed a LNS procedure. This LNS has three benefits, it allows us : to generate possible routes for the SPP, to always have a feasible primal solution and, during the segments, to continuously improve the best found solution. To extend the LNS' search space, five new operators have also been proposed.

According to the results, we observed that the new operators and the constructive heuristic permit a dramatic reduction in term of solutions' costs for the generated instances (respectively 13.76%, 20.82% and 14.39% for the small, medium and large sets). On the real instances, the algorithm permitted, on average, a 37% reduction in travel time and a 16% increase in the continuity of care. The developed method has been approved by our industrial partner, integrated in their software, and used by Alayacare's clients around the world (Canada, USA, Australia, Singapore) since November 2017.

Acknowledgment

We thank our industrial partner Alayacare for their precious collaboration, confidence and financial support together with the grant agencies MITACS and MEDTEQ. This project has received the Canadian Operational Research Society's practice prize in June 2018.

References

Begur, S. V., Miller, D. M., & Weaver, J. R. (1997). An integrated spatial dss for scheduling and routing home-health-care nurses. *Interfaces*, *27*, 35–48.

Bertels, S., & Fahle, T. (2006). A hybrid setup for a hybrid scenario: combining heuristics for the home health care problem. *Computers & Operations Research*, *33*, 2866–2890.

Borsani, V., Matta, A., Beschi, G., & Sommaruga, F. (2006). A home care scheduling model for human resources. *IEEE Service systems and service management, 2006 international conference on*, *1*, 449–454.

Braekers, K., Hartl, R. F., Parragh, S. N., & Tricoire, F. (2016). A bi-objective home care scheduling problem: Analyzing the trade-off between costs and client inconvenience. *European Journal of Operational Research*, *248*, 428–443.

Cappanera, P., Scutellà, M. G., Nervi, F., & Galli, L. (2017). Demand uncertainty in robust home care optimization. *Omega*, *80*, 95–110.

Cheng, E., & Rich, J. L. (1998). A home health care routing and scheduling problem. *Working paper*, *NA*, NA.

Cissé, M., Yalçındağ, S., Kergosien, Y., Şahin, E., Lenté, C., & Matta, A. (2017). Or problems related to home health care: A review of relevant routing and scheduling problems. *Operations Research for Health Care*, *13*, 1–22.

Decerle, J., Grunder, O., El Hassani, A. H., & Barakat, O. (2018). A memetic algorithm for a home health care routing and scheduling problem. *Operations research for health care*, *16*, 59–71.

Di Gaspero, L., & Urli, T. (2014). A cp/lns approach for multi-day homecare scheduling problems. *International Workshop on Hybrid Metaheuristics*, *NA*, 1–15.

Duque, P. M., Castro, M., Sörensen, K., & Goos, P. (2015). Home care service planning. the case of landelijke thuiszorg. *European Journal of Operational Research*, *243*, 292–301.

Fikar, C., & Hirsch, P. (2017). Home health care routing and scheduling: A review. *Computers & Operations Research*, *77*, 86–95.

Frifita, S., Masmoudi, M., & Euchi, J. (2017). General variable neighborhood search for home healthcare routing and scheduling problem with time windows and synchronized visits. *Electronic Notes in Discrete Mathematics*, *58*, 63–70.

Gamst, M., & Jensen, T. S. (2012). A branch-and-price algorithm for the long-term home care scheduling problem. *Operations research proceedings 2011*, *NA*, 483–488.

Gendreau, M., & Potvin, J.-Y. (2010). *Handbook of metaheuristics* volume 2. Springer.

Hiermann, G., Prandtstetter, M., Rendl, A., Puchinger, J., & Raidl, G. R. (2015). Metaheuristics for solving a multimodal home-healthcare scheduling problem. *Central European Journal of Operations Research*, *23*, 89–113.

Ichoua, S., Gendreau, M., & Potvin, J.-Y. (2003). Vehicle dispatching with time-dependent travel times. *European journal of operational research*, *144*, 379–396.

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *science*, *220*, 671–680.

Lin, C.-C., Hung, L.-P., Liu, W.-Y., & Tsai, M.-C. (2018). Jointly rostering, routing, and rerostering for home health care services: A harmony search approach with genetic, saturation, inheritance, and immigrant schemes. *Computers & Industrial Engineering*, *115*, 151–166.

Macintyre, C. R., Ruth, D., & Ansari, Z. (2002). Hospital in the home is cost saving for appropriately selected patients: a comparison with in-hospital care. *International Journal for Quality in Health Care*, *14*, 285–293.

Maire, S., & Amanda, B. (2014). *Receiving care at home*. Statistics Canada.

Manerba, D., & Mansini, R. (2016). The nurse routing problem with workload constraints and incompatible services. *IFAC-PapersOnLine*, *49*, 1192–1197.

Mendoza, J. E., Rousseau, L.-M., & Villegas, J. G. (2016). A hybrid metaheuristic for the vehicle routing problem with stochastic demand and duration constraints. *Journal of Heuristics*, *22*, 539–566.

Muter, İ., Birbil, Ş. İ., & Şahin, G. (2010). Combination of metaheuristic and exact algorithms for solving set covering-type optimization problems. *INFORMS Journal on Computing*, *22*, 603–619.

Nickel, S., Schröder, M., & Steeg, J. (2012). Mid-term and short-term planning support for home health care services. *European Journal of Operational Research*, *219*, 574–587.

Rest, K.-D., & Hirsch, P. (2016). Daily scheduling of home health care services using time-dependent public transport. *Flexible Services and Manufacturing Journal*, *28*, 495–525.

Ropke, S., & Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation science*, *40*, 455–472.

Rosing, K., & ReVelle, C. (1997). Heuristic concentration: Two stage solution construction. *European Journal of Operational Research*, *97*, 75–86.

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

Shaw, P. (1998). Using constraint programming and local search methods to solve vehicle routing problems. *International Conference on Principles and Practice of Constraint Programming*, *NA*, 417–431.

Torres-Ramos, A., Alfonso-Lizarazo, E., Reyes-Rubiano, L., & Quintero-Araújo, C. (2014). Mathematical model for the home health care routing and scheduling problem with multiple treatments and time windows. *Proceedings of the 1st International Conference on Mathematical Methods & Computational Techniques in Science & Engineering*, *NA*, 140–145.

Trautsamwieser, A., & Hirsch, P. (2014). A branch-price-and-cut approach for solving the medium-term home health care planning problem. *Networks*, *64*, 143–159.

Yalçındağ, S., Cappanera, P., Scutellà, M. G., Şahin, E., & Matta, A. (2016). Pattern-based decompositions for human resource planning in home health care services. *Computers & Operations Research*, *73*, 12–26.

Zhang, T., Yang, X., Chen, Q., Bai, L., & Chen, W. (2018). Modified aco for home health care scheduling and routing problem in chinese communities. *Networking, Sensing and Control (ICNSC), 2018 IEEE 15th International Conference on*, *NA*, 1–6.

Appendix

	CL	CL + NW		CL + NW + $Heur_{SPP}$	
	Value	Value	Gap	Value	Gap
Small_01	753650.31	693571.51	**-7.97%**	655170.32	**-13.07%**
Small_02	388578.40	333827.24	**-14.09%**	259383.96	**-33.25%**
Small_03	261321.75	259734.41	**-0.61%**	256827.88	**-1.72%**
Small_04	307137.08	302363.23	**-1.55%**	276264.58	**-10.05%**
Small_05	324180.21	320682.19	**-1.08%**	298047.69	**-8.06%**
Small_06	602088.16	539567.85	**-10.38%**	479109.71	**-20.43%**
Small_07	273176.12	265288.10	**-2.89%**	251640.48	**-7.88%**
Small_08	1528431.83	1527824.96	**-0.04%**	1524747.94	**-0.24%**
Small_09	394693.73	390526.27	**-1.06%**	370272.71	**-6.19%**
Small_10	469118.08	398505.77	**-15.05%**	383520.66	**-18.25%**
Small_11	283330.11	273207.25	**-3.57%**	255041.27	**-9.98%**
Small_12	938840.89	872176.95	**-7.10%**	790714.41	**-15.78%**
Small_13	244551.83	234983.10	**-3.91%**	229196.10	**-6.28%**
Small_14	860348.70	797670.64	**-7.29%**	734774.10	**-14.60%**
Small_15	993613.49	934921.45	**-5.91%**	899489.84	**-9.47%**
Small_16	447580.61	428907.60	**-4.17%**	410401.47	**-8.31%**
Small_17	1096303.37	937055.30	**-14.53%**	799396.40	**-27.08%**
Small_18	559169.93	464071.39	**-17.01%**	459508.10	**-17.82%**
Small_19	521073.62	440239.02	**-15.51%**	421205.03	**-19.17%**
Small_20	881554.92	715901.54	**-18.79%**	639114.14	**-27.50%**
Mean gap			**-7.63%**		**-13.76%**

Table 5: Comparison of the scenarios for the small instances

	CL	CL + NW		CL + NW + $Heur_{SPP}$	
	Value	Value	Gap	Value	Gap
Medium_01	1588963.97	1387766.74	**-12.66%**	1081305.18	**-31.95%**
Medium_02	828816.02	764033.19	**-7.82%**	649277.75	**-21.66%**
Medium_03	1286191.22	1066939.71	**-17.05%**	826778.72	**-35.72%**
Medium_04	800638.52	698931.17	**-12.70%**	659917.34	**-17.58%**
Medium_05	618752.83	609918.10	**-1.43%**	594481.40	**-3.92%**
Medium_06	887273.58	704931.93	**-20.55%**	627804.64	**-29.24%**
Medium_07	888716.31	889382.71	0.07%	844053.29	**-5.03%**
Medium_08	785631.13	695492.88	**-11.47%**	559769.34	**-28.75%**
Medium_09	685023.22	655755.50	**-4.27%**	540411.70	**-21.11%**
Medium_10	786320.41	773141.13	**-1.68%**	738833.99	**-6.04%**
Medium_11	937630.59	873310.67	**-6.86%**	834727.72	**-10.97%**
Medium_12	596877.70	574721.15	**-3.71%**	524259.20	**-12.17%**
Medium_13	1039973.26	816774.91	**-21.46%**	658848.83	**-36.65%**
Medium_14	708509.23	606313.06	**-14.42%**	541961.29	**-23.51%**
Medium_15	801160.06	671760.95	**-16.15%**	595359.66	**-25.69%**
Medium_16	845822.10	826898.21	**-2.24%**	705922.75	**-16.54%**
Medium_17	776339.87	716435.76	**-7.72%**	638842.83	**-17.71%**
Medium_18	2207257.99	1933365.62	**-12.41%**	1510478.67	**-31.57%**
Medium_19	607654.99	560813.02	**-7.71%**	532946.69	**-12.29%**
Medium_20	844354.06	683449.57	**-19.06%**	605117.08	**-28.33%**
Mean gap			**-10.06%**		**-20.82%**

Table 6: Comparison of the scenarios for the medium instances

	CL	CL + NW		CL + NW + $Heur_{SPP}$	
	Value	Value	Gap	Value	Gap
Large_01	1315840.27	1150528.85	**-12.56%**	1001922.79	**-23.86%**
Large_02	1271025.39	1280565.37	0.75%	1051504.59	**-17.27%**
Large_03	1275166.17	1255270.51	**-1.56%**	1048994.79	**-17.74%**
Large_04	1349729.93	1335133.74	**-1.08%**	1188032.84	**-11.98%**
Large_05	1252057.51	1215604.27	**-2.91%**	1075480.95	**-14.10%**
Large_06	1163047.20	1186513.28	2.02%	1051132.77	**-9.62%**
Large_07	1171658.52	1138382.32	**-2.84%**	966833.28	**-17.48%**
Large_08	1022707.50	1042276.78	1.91%	956056.27	**-6.52%**
Large_09	1253375.63	1183201.19	**-5.60%**	1007451.18	**-19.62%**
Large_10	1128399.05	1130063.19	0.15%	963391.38	**-14.62%**
Large_11	1249775.60	1244545.95	**-0.42%**	1087580.95	**-12.98%**
Large_12	1270174.66	1116505.44	**-12.10%**	955662.29	**-24.76%**
Large_13	1058909.80	1070766.14	1.12%	986339.54	**-6.85%**
Large_14	988281.59	997946.76	0.98%	927882.28	**-6.11%**
Large_15	1545000.49	1366852.11	**-11.53%**	1123416.86	**-27.29%**
Large_16	1239669.08	1263805.39	1.95%	1116903.18	**-9.90%**
Large_17	1036061.19	1040816.36	0.46%	939885.29	**-9.28%**
Large_18	1042017.09	1033234.67	**-0.84%**	932547.24	**-10.51%**
Large_19	1250220.56	1176267.76	**-5.92%**	1040813.51	**-16.75%**
Large_20	1128135.80	1141223.34	1.16%	1008533.86	**-10.60%**
Mean gap			**-2.34%**		**-14.39%**

Table 7: Comparison of the scenarios for the large instances

	CL		CL + NW		CL + NW + Heur	
	time (s)	Iterations	time (s)	Iterations	time (s)	Iterations
Small_Instances	149.8	100000	164.9	100000	163.8	100000
Medium_Instances	517.5	98607	597.6	79834	584	84928
Large_Instances	600	32211	600	24101	600	24258

Table 8: Comparison of the computation time and number of iteration for the three scenarios (Average over the 10 runs)

Weight	γ_1	γ_2	γ_3	γ_4	β_1	β_2
Value	1000	30	1000	100	100	50000

Table 9: Values of the weights used in the objective functions