Continuity of care in home health care scheduling: a rolling horizon approach

Şeyma Güven-Koçak[1] · Aliza Heching[2] · Pınar Keskinocak[1] · Alejandro Toriello[1]

Abstract

This work addresses a home health care scheduling problem faced by home care agencies. In home health care scheduling, there is a desire to retain consistency with respect to the home health aide servicing each patient; this consistency is referred to as *continuity of care*. To address this preference for continuity of care, we propose a rolling horizon approach to the scheduling problem and introduce the consistent home health care scheduling problem (Con-HHCSP). The goal is to be able to quantify and control the deviation of the new schedule suggested each day from the existing schedule in place, so that some of the existing assignments may be retained in the new schedule that is produced. We present two different methods to solve Con-HHCSP on a daily basis: an integer programming-based method with approximations and a variant of a petal heuristic. We discuss the performance and computational efficiency of these methods.

Keywords Home health care services · Continuity of care · Nurse assignment · Scheduling · Consistency

1 Introduction

Due to the increasing average age of the population, especially in developed countries, there is a growing need for long-term medical care and assistance for elderly people. It is advantageous—both to providers and to patients—to provide this care in the patient's home as long as possible. For providers, a long-term stay in hospitals or nursing homes is more costly than providing care and assistance to patients in their homes. Meanwhile, patients feel more comfortable in their own homes, resulting in improved quality of life. Therefore, programs for *aging in place*, which include an individual's freedom to choose the place to live regardless of age or level of ability, are widely endorsed by social and health services providers. The corresponding growth in home care services, i.e., services delivered outside of a structured setting, motivates the study of the scheduling of aides who visit patient homes.

Home care services vary with respect to the scope and structure of the care they provide. In this work, we consider *hospice care*, which is aimed at relieving terminally ill patients from pain, stress, and suffering. In the USA, in order for a patient to be hospice-eligible their physician must certify that their expected remaining lifetime is less than six months if their condition were to run its normal course. Hospice care focuses on pain management and improved patient quality of life during this time period rather than focusing on curative care. We address a real-world home health care scheduling problem (HHCSP) faced by a home care agency in the USA. The agency has operated for more than two decades and has branches in over twenty states. To lay out the details of the problem and provide effective solutions using real-world data, we collaborated with one of the agency's branches. The branch provides a number of different services in accordance with the patients' needs and plan of care including social work, nursing, and chaplain visits. Patients are located in settings including their home, skilled nursing facilities, and assisted living facilities, among others. The

✉ Şeyma Güven-Koçak
  seymaguven@gatech.edu

  Aliza Heching
  ahechi@us.ibm.com

  Pınar Keskinocak
  pinar@isye.gatech.edu

  Alejandro Toriello
  atoriello@isye.gatech.edu

[1] H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA

[2] Thomas J. Watson Research Center, Yorktown Heights, NY, USA

service most commonly provided is patient personal care; this service is performed by home health aides (HHAs) who are trained and certified health care workers.

Any proposed solution method to the scheduling problem must consider the unique characteristics of home care service. One of the most critical characteristics is that over-time aides develop a personal relationship with the patients and families they are servicing. As such, there is a desire to retain consistency with respect to the HHA servicing each patient. This consistency in care, referred to as *continuity of care*, serves to increase the overall patient quality of care and patient and family satisfaction. Continuity of care allows a HHA to better monitor patient status without loss of information, so that patients perceive a better quality of service with greater predictability and coherence (Haggerty, 2003). Patients also get comfortable with their aides over time and prefer to be served by the same aide due to the private and sensitive nature of the service they get. It is therefore desirable to maintain consistency in the patient–aide assignments.

Continuity of care policy has usually been addressed in the literature by developing models for a long planning period and keeping the aide–patient assignments consistent over time. However, the goal in a long-term plan is to keep the assignments consistent, with an assumption that there is no change in the data set. However, in real applications, the patient data set changes over time. Here, we acknowledge this and aim for continuity of care in a changing environment. In the hospice care setting, there is a frequent need to change the schedules due to unpredictable changes in the patient set. This is due to the nature of the hospice care environment, where a patient's expected remaining lifetime is less than six months. In fact, according to 2014 data, average patient length of stay in hospice is approximately 70 days with a median of around 23 days and often is as short as seven days or less (NHPCO, 2016). Moreover, 28.2% of the patients leave the service within 7 days, 26.5% of them leave after 7 but within 30 days, and only 13.1% of them stay longer than 180 days (NHPCO, 2016). The patients leaving the service or entering the service are not predictable. Thus, it is not possible to schedule for future periods, where the set of patients is unknown. This reveals the need for adopting a rolling horizon approach in the solution method, where we solve a daily HHCSP problem when there is a change in the data set and keep the new patient–aide assignments consistent with the previous schedule.

The need for frequent schedule changes coupled with the desire to maintain continuity of care motivates us to develop a solution method that updates an existing schedule rather than creating a new schedule from scratch. In this work, we present two different constructive methods to solve HHCSP on a daily basis: an integer programming-based method with approximations and a variant of a petal heuristic. We also implement various improvement heuristics, such as single

swap, double swap, combined single and double swap, and large neighborhood search (LNS) heuristic. Using our heuristic methods for HHCSP, we observed that the daily cost of operations is decreased by $3,268, which constitutes around 43% improvement over the current schedule in operation, with 76% decrease in labor cost (cost of idle time for full-time aides and working hours for part-time aides), 40% decrease in travel costs, slight increase in over-time cost, and 33% increase in preference violation cost. We present adjustments on these methods to address the schedule updating problem, where the goal is to be able to quantify and control the deviation from the existing schedule in place, so that some of the existing assignments may be retained in the new schedule that is produced. We discuss the performance and computational efficiencies of these methods.

The contributions of this paper are threefold. First, we introduce the consistent home health care scheduling problem (Con-HHCSP) with multiple features including patient preference, assignment consistency, and heterogeneous employee working time. Second, to the best of our knowledge, this work is the first to present solution methods and analysis for Con-HHCSP for a multi-aide problem using a rolling horizon approach. Although continuity of care has been taken into consideration in the literature, the vast majority of the proposed methods aim to create a long-term schedule with a consistency in the aide–patient assignments without considering a dynamic approach. Finally, quantifying the deviation from the prior period's schedule will provide business decision-makers with a quantifiable measure of the actual cost of the continuity of care policy and allow for relaxations or limitations in the policy based on this cost analysis.

2 Literature review

The home health care scheduling problem (HHCSP) can be seen as a variant of a multi-depot, multi-vehicle routing problem (VRP) with various labor-related and time-related constraints. However, HHCSP has some aspects that are unique to scheduling such as a desire to retain consistency in the HHA–patient assignments, a need to assign HHAs with preferred skills, and a need to consider specific labor-related constraints.

The uniqueness of the HHCSP problem, coupled with the growth in home health services, has resulted in a growing body of HHCSP literature. Comprehensive literature reviews of home health care logistics problems and workforce scheduling problems can be found in (Fathollahi-Fard et al., 2020; Grieco et al., 2020; Gutiérrez & Vidal, 2013; Castillo-Salazar et al., 2014; Fikar & Hirsch, 2017; Yalçındag et al., 2011; Hulshof et al., 2012). Many variants of HHCSP have also been studied. Some of the key differences between the problems are whether HHAs are heterogeneous or homo-

geneous (e.g., with respect to skills, availability), length of the planning horizon (e.g., daily or weekly planning horizon), whether or not to consider time windows, and the need to model simultaneous/interdependent visits. Moreover, the objective functions may differ among the problems. Although most studies consider minimizing travel cost as a part of the objective, some studies also consider maximizing HHA utilization, maximizing patient satisfaction, or simultaneously optimizing multiple objectives.

The main characteristic of our work that differentiates it from the existing literature is its focus on ensuring continuity of care using a rolling horizon approach. In addition, our problem includes complexities such as having part-time and full-time aides with different constraints, and patient preferences that are handled as soft constraints. We explain the similarities and differences of the most relevant papers with similar key features and complexities below, and summarize some key features in Table 8 in the Appendix. We include a group of papers which study consistency requirements, as well as some papers which do not have consistency requirements, but may have other similarities in our review.

To the best of our knowledge, Bennett & Erera's (2011) paper is the first to consider a rolling horizon myopic planning approach to handle continuity of care in a home health care context. However, our work differs from theirs in the following important aspects. First, they consider a single-nurse case of the problem, while we propose solution methods for a multiple-nurse case as the service areas of the nurses overlap in our problem setting. Second, they consider a weekly schedule with a very strict requirement of consistency in the visit times, by forcing the visit times occurring on the same weekdays to be at the same time from week to week. On the other hand, we focus on the consistency in nurse assignments, since, based on the information we obtained from the home health care agency, we concluded that consistency in nurse assignments is critical in the hospice care context. Third, they propose various heuristic methods to handle the problem, while we propose both a MIP model and related heuristics, which means we can benchmark the performance of one with the other. Lastly, we were able to use a real data set and had an opportunity to see the performance and cost effectiveness of our methods in a real-world situation.

Our rolling horizon approach to handling continuity of care is conceptually similar to the idea of generating a master plan and updating it to generate daily operational plans. In this approach of master and operational schedules, the goal is to generate a master schedule which is used in the long-run and to create daily schedules to account for the daily changes in the data sets or operations. However, our approach differs in that our primary objective is to maintain consistency with the previous schedule, rather than adhering strictly to a predefined master schedule. By utilizing a rolling horizon approach, we address the challenge of conti-

nuity of care while incorporating flexibility and adaptability. Instead of developing a fixed master schedule that may not fully account for dynamic factors and unforeseen changes, we employ a continuously updating process that considers the existing schedule as a baseline. (Nickel et al., 2012) and (Jensen, 2012) study master plan and daily operational plans for the home health care scheduling problem. Although our problems are similar, there are some key differences. They assume that the daily changes that are handled via the operational plan are usually insignificant. However, in our context, the change may be significant so that a new master plan is needed. Moreover, they mainly focus on consistency in terms of visit times—since it is more critical in their context. In our setting, consistency of patient–aide assignments is more critical and we therefore ensure that our methodology handles this requirement.

Cappanera and Scutellà (2014) have made an important contribution to the home health care literature by creating a joint model to handle assignment, and scheduling aspects of the home health care problem. Although both our mathematical model for HHCSP and the model they propose consist of similar components, there are several aspects that differentiate our work from theirs. First, they do not consider the consistency requirement in their work whereas we place significant emphasis on this requirement in the current work. Although they aim to create a weekly schedule where the continuity of care is ensured by keeping the nurse assignments the same throughout the week, they do not consider preserving the consistency in the next period when they need to reoptimize the schedule. Moreover, their MIP-based model is not able to solve some of the instances, or may take a significant amount of time to solve some instances due to the complex nature of the problem. We propose both a MIP-based model and heuristics to enable fast and high-quality solutions for a variety of instances. Therefore, our work has unique contributions in the context of home health care problems as well as in a wider context of VRP where consistency may be present as an important component of the problem.

The continuity of care has mostly been handled via solving the problem in a long planning horizon, rather than using a rolling horizon approach, as in the work of (Carello & Lanzarone, 2014) and (Yalçındağ et al., 2016). As mentioned earlier, this approach is not applicable in all settings such as hospice care. There are also some papers which account for patient preferences when solving the daily home care scheduling problem, but they do not consider continuity of care or updating the schedule with consistency concerns (Braekers et al., 2016; Rasmussen et al., 2012). Thus, our work differs from these papers in terms of the main focus on updating the schedules in case of significant changes in the dataset.

Besides the home health care setting, consistency concerns appear in many different fields. In the vehicle routing

problem (VRP) studied in transportation literature, consistency concerns became important after a shift in practice from cost-oriented operations to customer-oriented operations (Groër et al., 2009). Many companies prefer drivers to visit the same customers roughly at the same time when the customers need service, since this helps drivers develop relationships with customers, resulting in better customer service (Wong, 2008). Groër et al. (2009), Kovacs et al. (2014), and Coelho et al. (2012) studied variants of VRP, where the consistency constraints are combined with the traditional VRP constraints. Moreover, consistency has been addressed in the traveling salesman problem (TSP) (Subramanyam & Gounaris, 2017) and the dynamic scheduling problem for aircraft landings (Beasley et al., 2004).

3 Problem description and model: consistent home health care scheduling problem (Con-HHCSP)

Home health care scheduling problem (HHCSP) is a variant of a multi-depot, multi-vehicle routing problem with labor-related and time-related constraints. We begin by describing the problem. Each day, each HHA is assigned a set of patients to be visited at scheduled times. The HHA travels from his home to deliver care to his assigned patients in accordance with the assigned schedule. The HHA travels from patient to patient; following completion of his daily assignments, the HHA returns home. We consider deterministic travel times and distances between patient locations (as well as between aide and patient locations); these distances may be calculated using a geographic information system tool. Service times are also considered to be deterministic, having a value roughly estimated based on prior experience. In our problem setting, we assume common service times across all patients.

Time-related constraints enforce regular working hours, i.e., these constraints enforce that all visits begin no earlier than a specified start time and end no later than a specified end time. Labor-related constraints are specific to the employee type: the workforce is comprised of part-time (PT) and full-time (FT) HHAs and each is bound by different rules. FT-HHAs are scheduled to work a fixed number of hours per week; a FT-HHA who works more than his regular scheduled hours is paid for his excess hours at an hourly overtime rate. However, the sum of regular hours and overtime hours may not exceed a maximum allowable number of working hours. PT-HHAs are paid at an hourly rate and may work up to a specified weekly regular number of hours with this hourly rate. If a PT-HHA works beyond the maximum number of permissible hours per week, then the PT-HHA must receive additional benefits such as insurance; the cost of these benefits is covered by the employer. Our labor-related constraints are different from

similar home health care scheduling problems discussed in the literature. Although most studies consider heterogeneous aides (e.g., skilled nurses), this heterogeneity mostly affects the assignment decisions. In our case, the heterogeneity in cost calculations and time restrictions introduces additional complexity to the problem.

In addition to the labor-related constraints, we consider patient preferences when assigning patients to the PT-HHA and FT-HHAs. Namely, patients may specify one or more HHAs that they prefer and/or characteristics of HHAs that they prefer (e.g., gender, languages spoken). These preferences are treated as soft constraints where a penalty is introduced to the objective function for each preference violation.

Similar to the labor-related constraints, having soft constraints for the assignment decisions is not common in the literature. In most cases, staff assignments are restricted due to hard constraints such as skills requirements. Modeling preferences as a soft constraint and introducing a penalty for preference violation introduces the challenge of appropriately selecting a penalty. Based on discussions with the company, we note that patient preferences are seriously considered (as these may directly impact customer satisfaction) but are balanced against costs incurred in meeting these preferences. We therefore introduced these preferences as a soft constraint and impose a violation penalty equal to an average cost of visiting a patient.

The consistent home health care scheduling problem (Con-HHCSP) arises when there is a need to solve HHCSP to create a new schedule due to a change in the dataset (e.g., change in available HHAs, change in patients requiring care) and there is a desire to retain *consistency* between the patient–aide assignments in the new schedule and the patient–aide assignments in the prior period schedule. Here, consistency is measured by the ratio of the patient–aide assignments that remain consistent across the current and prior period schedules. When calculating this ratio, note that we only consider the patients requiring service in both the prior period and the current period. Based on the problem setting, the consistency preference may be very strict—such that all of the patients requiring service in both periods must be served by the same aide in the current period (as the aide who provided service in the prior period)—or it may be loose, such that only some of the patient–aide assignments must be retained between the prior period and current period schedules. In other words, the cost associated with changing the patient–aide assignments affects the scheduling decisions made.

In addition to the consistency preference, this problem has a dynamic nature. If all data parameters remained constant over time, then decisions could be made for a long-term planning horizon. However, frequent changes in the dataset do not allow for long-term planning; for example, changes in aide availability (due to high staff turnover that character-

izes the industry) as well as changes in the patients requiring care (due to the nature of the hospice industry) are both common. Thus, a method for dynamically updating schedules is required. In this study, we only consider changes in the set of patients requiring care and assume that the set of aides and other data parameters remain constant.

In an effort to further decrease the size of the feasible region, we analyze the structure of our problem and observe that the maximum number of patients that may be visited by an HHA each day is $M = \lfloor H / \min_{j \in J}\{u_j\} \rfloor$, where H is the maximum number of working hours and u_j is the service time for patient j belonging to the set of patients J. Since service time is the main component of the total time and service times are assumed constant over all patients in our problem setting, this upper bound is very useful. In fact, when we solve the model without this bound we observe that most good quality solutions have all HHAs visiting at most M patients. (In rare cases, we observed HHAs visiting $M + 1$ patients; these cases involved overtime costs.) In addition, in considering the agency's current practice we observe that to establish fairness in workload among the HHAs, the agency assigns at most M patients to each HHA. Thus, we add this cut to the model (constraint (1c)) as a useful tool for cutting the feasible region as well as to balance workload and limit overtime costs.

We don't consider break and rest rules explicitly in our problem, but we believe that they can be easily incorporated into the schedule. We create the schedules in a way that the total number of working hours do not exceed the limit. If there is a lunch break, or other rest rules, they can be added into the schedule. Since we don't have time windows in this problem, addition of the breaks does not make the schedule infeasible, or affect the overall costs. If the part-time aides have other restrictions in terms of the working hours or days, these requirements can be handled by changing the available aide set for the specific day, or changing the maximum number of hours allowed for that particular aide. For example, if a set of aides cannot work on Wednesdays, then they can be removed from the aide set for that day. Similarly, if an aide can only work in the afternoon for 2 h on a certain day, the maximum time limit can be set to 2 h in the constraint which ensures that, and the resulting schedule can be shifted to the appropriate time window for that aide.

We now describe the mathematical model of Con-HHCSP, which we refer to as *Con-MIP*, using the notation presented in Table 1. The key model assumptions are: (i) patients do not have time windows, (ii) patients require only one HHA to be present in a service visit, (iii) all visits to the same patient have the same average duration, (iv) an HHA's day begins when he arrives at his first patient and ends when he leaves his last patient, and (v) part-time HHAs and full-time HHAs have different associated time (availability) restrictions and labor costs. The objective is to minimize the sum of the travel costs (from the patient homes to the homes of the aides), idle time cost which is the time that is not spent on duty for full-time aides, labor cost for part-time aides, overtime costs for all aides, overuse cost for part-time aides when their daily workload exceeds a specified amount, the costs associated with patient preferences, and the penalty for inconsistency. The objective function includes all of these seven components in their explained order below. Note that labor cost for full-time aides is not part of the equation as it is fixed.

$$\min \sum_{\substack{g \in G \\ (a,b) \in Loc}} c_T y_{gab} D_{ab} + \sum_{f \in F} c_F \gamma_f + \sum_{p \in P} c_P w_p$$

$$\sum_{f \in F} c_{OF} O_f + \sum_{p \in P} c_{OP} \theta_p + \sum_{j \in J} \lambda^P P_j + \sum_{\substack{j \in J^0 \\ g \in G}} \lambda d_{gj} \quad (1a)$$

$$\text{s.t.} \sum_{g \in G} x_{gj} = \sum_{g \in G_j} x_{gj} + P_j = 1 \qquad \forall j \in J \quad (1b)$$

$$\sum_{j \in J} x_{gj} \leq M \qquad \forall g \in G \quad (1c)$$

$$w_g = \sum_{j \in J} \left(s_j x_{gj} + \sum_{i \in J} T_{ij} y_{gij} \right) \qquad \forall g \in G \quad (1d)$$

$$w_g \leq \alpha H \qquad \forall g \in G \quad (1e)$$

$$\gamma_f \geq \max(0, H - w_f) \qquad \forall f \in F \quad (1f)$$

$$O_f \geq \max(0, w_f - H) \qquad \forall f \in F \quad (1g)$$

$$(\alpha H - B^d)\theta_p \geq O_p \geq \max(0, w_p - B^d) \qquad \forall p \in P \quad (1h)$$

$$\sum_{b \in J \cup \{g\}} y_{ggb} = \sum_{a \in J \cup \{g\}} y_{gag} = 1 \qquad \forall g \in G \quad (1i)$$

$$\sum_{a \in J \cup \{g\}} y_{gaj} = \sum_{b \in J \cup \{g\}} y_{gjb} = x_{gj} \qquad \forall g \in G, \forall j \in J \quad (1j)$$

$$\sum_{a \in S, b \in S} y_{gab} \leq |S| - 1 \qquad \forall g \in G, \forall S \subseteq J \quad (1k)$$

$$x_{gj}^0 - x_{gj} \leq d_{gj} \qquad \forall g \in G, \forall j \in J^0 \quad (1l)$$

$$d_{gj} \in \{0, 1\} \qquad \forall g \in G, \forall j \in J^0 \quad (1m)$$

$$x_{gj} \in \{0, 1\} \qquad \forall g \in G, \forall j \in J \quad (1n)$$

$$y_{gab} \in \{0, 1\} \qquad \forall g \in G, \forall (a, b) \in Loc \quad (1o)$$

$$\theta_p \in \{0, 1\} \qquad \forall p \in P \quad (1p)$$

Constraints (1b) ensure that every patient is visited once. They also assign the value of the preference violation variable P_j. Constraints (1c) establish an upper bound on the number of patients that may be visited by an aide, based on the service time and the total working hours. Constraints (1d) calculate the actual working time in a day, which is equal to the sum of travel times between patients plus visit durations. Constraints (1e) define an upper bound for the total working hours. Constraints (1f) define idle time for full-time aides and restrict the idle time to be non-

Table 1 Notation for the Con-MIP model of Con-HHCSP	**Index sets**	
	J	Set of patients requiring service in the current period
	J^0	Patients who require care in both the prior period and the current period. (Intersection of the current set of patients and the prior period set of patients.)
	F	Full-time home health aides (FT-HHAs)
	P	Part-time home health aides (PT-HHAs)
	G	All HHAs. $G = F \cup P$
	G_j	HHAs preferred and available to serve patient $j \in J$. $G_j \subseteq G$
	Loc	The set of all pairs of nodes that can be sequentially visited by any aide in a tour, including from one patient to another $J \times J$, the starting leg of the trip for all aides $G \times J$ from their homes and the ending leg of the trip $J \times G$ $Loc = \{J \times J \ \cup G \times J \ \cup \ J \times G\}$
	Data parameters	
	H	Length of regular workday, in hours
	B^d	Maximum number of daily hours a part-time employee may be scheduled before changing to full-time status, where $B^d = B/5$, given that the limit on the number of weekly working hours is B.
	M	The upper bound on the maximum number of patients an aide can visit on a day, as explained in Sect. 4.1
	α	Maximum ratio between a journey with overtime hours and the regular workday length H. E.g.: $\alpha = 1.25$, $H = 8$ implies that HHAs may not work more than 10 hours a day, including overtime
	s_j	Average duration of visit time at patient $j \in J$
	s	Minimum duration of visit times at patients $s = \min_{j \in J} s_j$
	D_{ab}	distance from location a to location b, for $(a, b) \in Loc$
	T_{ab}	travel time from location a to location b, for $(a, b) \in Loc$
	c_T	Cost of transportation per mile driven
	c_F	Cost of labor per hour worked for full-time HHA
	c_P	Cost of labor per hour worked for part-time HHA
	c_{OP}	Associated cost increase if a part-time HHA works over $5B$ hrs/wk (B hrs/day) and is converted to a full-time HHA (health insurance, etc.)
	c_{OF}	Hourly penalty associated with overtime work for full-time HHAs
	λ^P	Penalty associated with violating aide–patient assignment preferences
	λ	Inconsistency penalty associated with changing the aide–patient assignment when updating the schedule
	x^0_{gj}	Indicator data equal to 1 if and only if aide $g \in G$ visits patient $j \in J^0$ in the previous period
	Variables	
	x_{gj}	Binary variable equal to 1 if and only if HHA $g \in G_j$ visits patient $j \in J$
	y_{gab}	Binary variable equal to 1 if and only if HHA $g \in G$ traverses the arc $(a, b) \in Loc$. Note that, $y_{ggg} = 1$ if the aide $g \in G$ does not work on that day.
	w_g	Total working hours for HHA $g \in G$
	θ_p	Binary variable equal to 1 if and only if $w_p > B$ for $p \in P$.
	O_g	Total daily overtime hours for HHA $g \in G$
	γ_f	Idle time for FT-HHA $f \in F$ ($\gamma_f = \max(0, H - w_f)$)
	P_j	Binary variable equal to 1 if and only if patient $j \in J$ is assigned a non-preferred HHA
	d_{gj}	Inconsistency variable, binary variable equal to 1 if and only if patient $j \in J^0$ was assigned to aide $g \in G$ in the prior period and is reassigned to a different aide in the current period.

negative. Constraints (1g) define extra working hours for full-time HHAs as total hours worked by the HHA minus the number of hours FT-HHAs are scheduled to work and restricts this value to be nonnegative. Constraints (1h) sets the overuse indicator equal to one if working hours of a PT-HHA exceeds the maximum permissible number of hours per week. Constraints (1i) and (1j) are the routing constraints for travel between patients. Each HHA will begin from home and return to home as the start point and end point of his daily work "route". Routing flows will be balanced. Constraints (1k) are used to eliminate subtours. Note that there are exponentially many subtour elimination constraints, so that a cutting plane algorithm is implemented to iteratively identify violated inequalities, and to add them to the model to improve the solution quality and eliminate subtours..

To calculate the value of the inconsistency variables d_{gj}, $\forall j \in J$, $\forall g \in G$, we add the following set of constraints (11) to the model. d_{gj} takes a value of 1 if and only if HHA g was assigned to patient j in the prior period but is not assigned in the current period (and patient $j \in J^0$). The inconsistency penalty is applied for each positive d_{gj} variable. Note that we write these constraints only for the patients who are in both the prior period dataset and in the current period dataset. Moreover, the variable d_{gj} is not set equal to 1 if HHA g was not assigned to patient j in the prior period but is assigned in the current period, since this would result in double counting the changes in assignments.

In order to improve the formulation, we add some cuts to the model. Constraints (2a) ensure that the variable y_{ggg} becomes 1 when an aide $g \in G$ does not visit anyone, and becomes 0 when he visits at least one patient. Constraints (2b) set the overuse variable θ_p for part-time aide p to 1 if the aide makes more visits than the maximum number of visits that can be assigned to a part-time aide, which is found by B^d/s–a maximum time that a part-time aide can work, divided by the minimum service time.

$$\frac{1}{M} \sum_{j \in J} x_{gj} \le 1 - y_{ggg} \le \sum_{j \in J} x_{gj} \qquad \forall g \in G \qquad (2a)$$

$$\sum_{j \in J} x_{pj} - \frac{B^d}{s} \le \left(M - \frac{B^d}{s}\right)\theta_p \qquad \forall p \in P \qquad (2b)$$

4 Methodology

As HHCSP is a variant of VRP, it is NP-hard. Solving to optimality is possible only for very small size instances. This motivates the development of approximate or heuristic methods. We develop two main constructive methods to generate high-quality solutions. The first method is a mixed-integer programming model with approximations.

The second method is a petal-based heuristic adapted for this specific problem. We also use various improvement heuristics, such as swap and LNS heuristics. These methods, described in this section, are computationally tested, with results outlined in Sect. 5.

4.1 Mixed integer programming model with approximations

As the original Con-HHCSP is intractable, we reformulate the problem by integrating some approximations in order to simplify it. Our aim is to be able to solve the problem for larger instances while obtaining a high-quality solution. We impose three main approximations: clustering, within-cluster travels, and maximum number of visits. These approximations are explained below.

Clustering The original arc-based formulation of a HHCSP requires all combinations of edges to be considered as a possible route. However, most good solutions are less likely to include routes going back and forth between the farthest points. For example, suppose A and B are two nodes in City 1 and C is a node in City 2, which is far away from City 1. If an aide is scheduled to visit locations A, B, and C on a given day, one would not expect the visit order to be A-C-B, or B-C-A (since it would be more reasonable to visit two nodes in City 1 and to then travel to City 2). Since we don't have time windows, this assumption is reasonable in our problem setting—considering both time and travel cost. We therefore restructure our graph as a set of clusters of patients and consider routing constraints to be between clusters and within each cluster rather than between all patients. In other words, an HHA travels between the clusters by visiting each cluster at most once. When an HHA visits a cluster, he may travel within the cluster (between the patients). Note that an aide may or may not visit all patients within a cluster he is visiting. The clusters are generated based on the location of the patients only, whereas the decision of assigning an aide to a patient includes not only the location associated costs but also the consideration of patient preferences, and continuity of care. Therefore, we adopt a flexible clustering approach, which is used to divide the network into large or small neighborhoods and not necessarily into a group of patients to be visited by the same aide. This reformulation enables us to decrease the number of routing variables and constraints in our model.

Within-cluster visits We cluster the patients using a hierarchical clustering with a "complete" (or "maximum") linkage method. In this method, the distance between two clusters u and v is calculated as follows: $d(u, v) = \max(d(u[i], v[j]))$ for all points i in cluster u and j in cluster v. In our model, we choose a small distance as a clustering criterion (i.e., 3 miles where the average distance between all patients is 32.5 miles), so that the distance between all nodes in a cluster

Table 2 Notation to be used in Con-MIP-A in addition to the notation presented in Table 1

Index sets	
K	Clusters of patients
J_k	Patients in cluster $k \in K$
K_g	Clusters that HHA $g \in G$ can visit (i.e., there are patients in that cluster who prefer HHA $g \in G$)
Loc	The set of all pairs of nodes that can be sequentially visited by any aide in a tour, including from one cluster to another $K \times K$, from a patient within a cluster to another in the same cluster $J[k] \times J[k]$, the starting leg of the trip for all aides from their homes to a cluster $G \times K$, and the ending leg of the trip $K \times G$ $Loc = \{K \times K \cup G \times K \cup K \times G\} \cup \{K \times J[k] \cup J[k] \times J[k] \cup J[k] \times K\}$
Data parameters	
D_{ab}	distance from location a to location b, for $(a, b) \in Loc$
T_{ab}	travel time from location a to location b, for $(a, b) \in Loc$
$O(j)$	The predetermined visiting order of the patient $j \in J$ in its cluster as explained in Sect. 4.1.
Variables	
x_{gk}	Binary variable equal to 1 if and only if aide $g \in G_j$ visits cluster $k \in K$

is very small. Consequently, a travel distance between two routes within the same cluster is very small. The result is many symmetric solutions in our model; we add an integer cut (constraint (3n)) to eliminate some of these symmetric solutions and to obtain a smaller feasible region. To decide which symmetric solution to promote, we solve a TSP for patients within each cluster and save the optimal TSP solution as a preferred order. In the mathematical model we set equal to zero all arcs that violate the preferred order. This method is inspired from Cappanera & Scutellà's work (2014).

Based on the approximations explained above, we modify the original MIP model to obtain mixed-integer programming model with approximations (Con-MIP-A). The changes in the notation, index sets, data parameters, and the variables are explained in Table 2.

$$\min \sum_{\substack{g \in G \\ (a,b) \in Loc}} c_T y_{gab} D_{ab} + \sum_{f \in F} c_F \gamma_f + \sum_{p \in P} c_P w_p$$

$$+ \sum_{f \in F} c_{OF} O_f + \sum_{p \in P} c_{OP} \theta_p + \sum_{j \in J} \lambda^P P_j + \sum_{\substack{j \in J^0 \\ g \in G}} \lambda d_{gj} \quad (3a)$$

$$\text{s.t.} \sum_{g \in G} x_{gj} = \sum_{g \in G_j} x_{gj} + P_j = 1 \qquad \forall j \in J \quad (3b)$$

$$\sum_{j \in J} x_{gj} \leq M \qquad \forall g \in G \quad (3c)$$

$$w_g = \sum_{k,l \in K} T_{lk} y_{glk} + \sum_{k \in K} \sum_{i,j \in J_k} T_{ij} y_{gij}$$
$$+ \sum_{j \in J} s_j x_{gj} \qquad \forall g \in G \quad (3d)$$

$$w_g \leq \alpha H \qquad \forall g \in G \quad (3e)$$

$$\gamma_f \geq \max(0, H - w_f) \qquad \forall f \in F \quad (3f)$$

$$O_f \geq \max(0, w_f - H) \qquad \forall f \in F \quad (3g)$$

$$(\alpha H - B^d)\theta_p \geq O_p \geq \max(0, w_p - B^d) \qquad \forall p \in P \quad (3h)$$

$$x_{gj} \leq x_{gk} \leq \sum_{j \in J_k} x_{gj} \qquad \forall g \in G, \forall k \in K, \forall j \in J_k \quad (3i)$$

$$\sum_{b \in K \bigcup \{g\}} y_{ggb} = \sum_{a \in K \bigcup \{g\}} y_{gag} = 1 \qquad \forall g \in G \quad (3j)$$

$$\sum_{b \in J_k \bigcup \{k\}} y_{gkb} = \sum_{a \in J_k \bigcup \{k\}} y_{gak} = 1 \qquad \forall g \in G, \forall k \in K \quad (3k)$$

$$\sum_{a \in K \bigcup \{g\}} y_{gak} = \sum_{b \in K \bigcup \{g\}} y_{gkb} = x_{gk} \qquad \forall g \in G, \forall k \in K \quad (3l)$$

$$\sum_{a \in J_k \bigcup \{k\}} y_{gaj} = \sum_{b \in J_k \bigcup \{k\}} y_{gjb} = x_{gj}$$
$$\forall g \in G, \forall k \in K, \forall j \in J_k \quad (3m)$$

$$y_{gab} = 0 \, \forall g \in G, \forall (a, b) \in Loc : O(a) > O(b) \quad (3n)$$

$$\sum_{a \in S, b \in S} y_{gab} x \leq |S| - 1 \qquad \forall g \in G, \forall S \subseteq K \quad (3o)$$

$$x_{gj}^0 - x_{gj} \leq d_{gj} \qquad \forall g \in G, \forall j \in J^0 \quad (3p)$$

$$d_{gj} \in \{0, 1\} \qquad \forall g \in G, \forall j \in J^0 \quad (3q)$$

$$x_{gj} \in \{0, 1\} \qquad \forall g \in G, \forall j \in J \quad (3r)$$

$$y_{gab} \in \{0, 1\} \qquad \forall g \in G, \forall (a, b) \in Loc \quad (3s)$$

$$\theta_p \in \{0, 1\} \qquad \forall p \in P \quad (3t)$$

$$x_{gk} \in \{0, 1\} \qquad \forall g \in G, \forall j \in J, \forall k \in K \quad (3u)$$

The objective function (3a) of Con-MIP-A remains the same as in Con-MIP. Note, however, that the routing variable y_{gab} is defined on different arcs Loc set as defined in Table 2. Since we cluster patients and consider routing between clusters and between patients within each cluster, the routing constraints

and time constraints are modified as follows. Constraints (3d) calculate the actual work done by aides, by adding the travel times between the visited clusters / patients and the service time for each patient visited. Constraints (3i) ensure that a cluster assignment variable is set to one if at least one patient in that cluster is visited by an aide, and that no patients in that cluster are visited if the cluster assignment variable is zero. Like patient assignment variables, cluster assignment variables are also binary variables as indicated by constraints (3r). Constraints (3j)–(3m) ensure the connectivity in the between-cluster and within-cluster routes. Constraints (3n) are used to eliminate symmetric solutions in within-cluster routing as explained in Sect. 4.1. Constraints (3o) are used to eliminate subtours between the clusters. As mentioned before, a cutting plane algorithm is implemented to handle the exponential number of subtour elimination constraints.

As in Con-MIP model, we add some cuts to Con-MIP-A as follows. As in Con-MIP, constraints (4a) ensure that the variable y_{ggg} becomes 1 when an aide $g \in G$ does not visit anyone and becomes 0 when he visits at least one patient. Similarly, constraints (4b) ensure that the variable y_{gkk} becomes 1 when an aide $g \in G$ does not visit anyone in a cluster $k \in K$, and becomes 0 when he visits at least one patient in that cluster. Constraints (4c) set the overuse variable θ_p for part-time aide p to 1 if the aide makes more visits than the maximum number of visits that can be assigned to a part-time aide.

$$\frac{1}{M} \sum_{j \in J} x_{gj} \leq 1 - y_{ggg} \leq \sum_{j \in J} x_{gj} \qquad \forall g \in G \tag{4a}$$

$$\frac{1}{M} \sum_{j \in J_k} x_{gj} \leq 1 - y_{gkk} \leq \sum_{j \in J_k} x_{gj}gj \quad \forall g \in G, \forall k \in K \tag{4b}$$

$$\sum_{j \in J} x_{pj} - \frac{B^d}{s} \leq (M - \frac{B^d}{s})\theta_p \qquad \forall p \in P \tag{4c}$$

Con-MIP-A remains intractable despite these approximations. We therefore impose a computing time limit and obtain a potentially sub-optimal solution. We apply a post-processing approach to the solution obtained using Con-MIP-A to arrange the schedule and minimize costs. We obtain the HHA–patient assignment information from the Con-MIP-A solution and solve a TSP to recreate the routes for each HHA. Since the number of patients in an HHA's route is limited, solving this TSP can be achieved efficiently. In other problem settings, where routes may be larger, a TSP heuristic may be utilized. In addition to further minimizing costs, this post-processing step also supports more accurate comparisons among different schedules by eliminating any idle times in the schedules and revealing the actual costs.

4.2 Petal-based heuristic for Con-HHCSP

The petal heuristic was first proposed by (Foster & Ryan, 1976) to address the vehicle scheduling problem. This solution approach involves two stages. In the first stage, a set of possible "good" routes is created. In the second stage, the assignment of the routes is made by solving a set partitioning problem. This approach is appropriate for our problem setting, as the number of daily visits is limited. In our instances, a single HHA can visit at most five patients each day so routing is simplified once the visit assignment is determined. We therefore develop a variant of this heuristic, denoted *Con-PH*, to address Con-HHCSP.

We first develop an algorithm to create a"good" set of clusters. Here,"good" clusters refer to groups of patients who are highly likely to be visited by the same aide (due to the patients' similarities with respect to location and preferences). When creating the clusters of patients, we exclude infeasible sets, i.e., sets of patients who cannot be visited by the same HHA on a single day within regular working hours. Algorithm 1 is used to generate a "good" set of clusters of patients who can be visited by the same HHA on a single day. Here, we aim to group patients into clusters without considering the order in which they are visited. The routes are created after the clusters are formed.

The pseudo-code of our cluster generation algorithm is written in Algorithm 1. Steps 1-5 are used to generate all possible singletons and pairs of patients; steps 6-16 are used to generate clusters of 3, 4, and 5 patients. Note that in our instance an HHA may visit at most five patients each day, so we limit the size of the clusters to equal five. In other instances, the size of the clusters may be set to a larger value in step 7 of the algorithm. In step 10, we generate C_x number of clusters of size x. Note that this value is predetermined based on preliminary experiments. In general, C_x should be large enough to provide a wide variety of options to choose from but small enough to be computationally feasible. Given a set of patients, the number of clusters of size x increases as x gets larger. Therefore, C_x should be larger for larger values of x in order to include more samples from the larger set of clusters. We repeat the second part of the algorithm (steps 7-16) twice for modified and unmodified distance matrices. The unmodified distance matrix is a location based distance matrix obtained from Google Maps.

We modify this distance matrix based on patient preferences and continuity of care. Namely, for each pair of patients, we add the preference cost to the distance between them if all of their preferred aides are different. In this way, we aim to cluster patients with similar aide preferences. This also introduces more flexibility in options when we assign

---

Algorithm 1: Algorithm for Cluster Generation

---

1 Initial Solution: $C = \emptyset$ (set of clusters), Ignore=NULL ;

2 (Create all possible clusters of one and two patients) **forall the** $i \in Patients$ **do**

3      $C = C \cup \{(i)\} \cup \emptyset$ /; **forall the** $j \in Patients \setminus \{i\}$ **do**

4   **if** $\{(i, j)\} \notin C$ **then**

5            $C = C \cup \{i, j\}$

6 **for** *modified and unmodified distance matrices* **do**

7   **forall the** $x \in [3, 4, 5]$ **do**

8   Create clusters of x patients **forall the** $i_1 \in Patients$ **do**

9   Ignore=i_1 ;

10   **while** *There are less than C_x clusters including patient i_1* **do**

11   C_x: the number of clusters of size x that each patient belongs to **while** *The size of the cluster is less than x ($|c| < x$)* **do**

12   Use Nearest Insertion to find $c = \{i_2, ..., i_x\} \in$ Patients \ Ignore. ;

13   Ignore=Ignore $\cup \{i_2, ..., i_x\}$;

14   **if** $\{i_1, i_2, ..., i_x\} \notin C$ *is a feasible route* **then**

15                    $C = C \cup \{i_1, i_2, ..., i_x\}$

16               $c = \{i_1, i_2, ..., i_{x-j}\}$

17   Keep the first j elements of the cluster the same, and create more clusters, where $j \in \{2, ..., x - 1\}$. Do this twice for each j, starting with $j = x - 1$.

---

the clusters to HHAs using a set partitioning formulation. To account for the deviations from the previous schedule, we modify the distance matrix as follows. For each pair of patients, we add the deviation cost to the distance between them if any of them is visited by a different aide in the previous assignment. This way, we aim to cluster patients who have been visited by the same aide in the previous assignment. This approach to modify the distance matrix is novel. These modifications to the distance matrix help to create clusters based on location, preferences, and previous assignments.

We further create clusters that guarantee highest consistency. That is, we cannot guarantee that the clusters generated using these modified and unmodified distance matrices will provide an option of retaining the previous schedule. Therefore, in addition to the clusters generated by Algorithm 1, we generate clusters that focus on consistency using the regret-based repair operators as follows. First, we take the routes of the previous schedule as an initial set of clusters and remove patients who are not in the current data set. Then, we use the regret operator to insert each new patient into an aide's daily route by considering the regret cost of postponing an insertion to later iterations (Potvin & Rousseau, 1993). In this way, we generate a route for each aide that retains all of the previous assignments.

The regret operator calculates the cost for several insertion positions for each patient and finds the difference between the cost of the best insertion position and the cost of the alternative insertion positions as follows:

$$\Delta_i = \sum_{k=2}^{\min(r,m)} (c_i^k - c_i^1).$$

Here, r is the regret operator parameter that refers to the number of routes to be considered in comparison and m is the total number of possible routes; c_i^k refers to the kth cheapest insertion cost for patient i. Based on this cost calculation, the regret operator inserts the patient with the largest difference between the cost of the best insertion position and alternative insertion positions. Namely, $i^* = \arg\max_{i \in I \setminus \{i\}} \Delta_i$ is inserted to its cheapest position. The process is repeated until all new patients are inserted. In our implementation, we set the regret parameter equal to half the size of the new patients.

After the clusters are created, we calculate the "assignment costs" of each HHA to each cluster. The assignment cost includes travel cost (total distance within the cluster plus distance to / from HHA's home from / to the nearest patient in the cluster), overtime cost, idle time cost for FT-HHA, hourly labor cost for PT-HHA, and the penalty for any preference violations (penalty for one preference violation, multiplied by the number of patients in the corresponding cluster who do not prefer that HHA). The main steps involved in creating the routes and calculating the costs of each cluster-HHA assignment are: (i) For each cluster, create a route using TSP and calculate the minimum travel distance and the travel time for the route. (ii) If the total travel time for the route exceeds the daily time limit then delete this route. Otherwise, add this route to the set of "good" routes. (iii) For each HHA, record the cost of visiting each route if the total travel time of the route is within the daily time limits.

The assignment cost for each aide and each route includes travel, labor (work time and idle time), overtime, preference penalty, and inconsistency penalty costs. As mentioned above, we first create a TSP tour with the set of patients within each cluster generated with Algorithm 1. Next, using the cheapest insertion algorithm, we create a route for each aide by determining the first and last patients to visit on the TSP tour of patients, based on the home locations of the aide and the patients. Once all possible routes are created, total travel costs, labor costs, and preference penalty costs are calculated as in the objective function described in Sect. 3. We further add inconsistency cost to the overall assignment cost between each cluster and aide. For a given cluster–aide pair, if there are patients in the cluster who were visited by another aide in the previous schedule, then the unit cost of inconsistency is multiplied by the number of patients in this situation and added to the assignment cost. This encourages

the set partitioning model solution to be consistent with the previous schedule.

After the assignment costs are calculated, a variant of a Set Partitioning Problem is solved to assign the routes to aides such that the cost of the resulting schedule is minimized. The notation used in the set partitioning model is explained in Table 6 and the model is presented in Appendix 1. Once the route–aide assignments are made, the schedules are created by calculating the arrival time for each patient in the predetermined routes.

To improve the solutions we obtain via Con-PH or Con-MIP-A, we also develop fast and simple improvement heuristics. Specifically, we implemented various swap heuristics and a Large Neighborhood Search heuristic adapted from Bent & Van Hentenryck's work (2004). Since its original version did not perform well due to the special characteristics of our problem, we adjusted Bent and Van Hentenryck's heuristic for our problem. The detailed explanation and the algorithms of our adjusted versions of swap heuristics, as well as the LNS heuristic, can be found in the Online Appendix.

4.3 Extensions

Since the home health care research has been motivated by practice, the problems addressed in the literature possess different needs and requirements (Fikar & Hirsch, 2017). Since real-world data has mostly been used to test the methods, the proposed methods were tailored to a particular problem setting. Some main differences considered in various studies are homogeneous vs. heterogeneous aides, and daily vs. weekly schedules. In terms of the consistency aspect of the problem we study here, consistency in visit times could also be desired in some settings. The petal-based heuristic we present is flexible to handle a wider range of problem features. In this section, we present possible extensions to the methodology to handle different requirements.

In many home health care problems, there are heterogeneous sets of aides with different skill set which make them eligible or ineligible for some visits. In some cases, hierarchy between these skills are considered. For example, an aide who can take care of the medications can also take care of the household duties, while the opposite is not true. The concept of skilled aides is similar to the "preferred aides" concept we presented in our problem setting, except that assigning based on skills needs to be a hard constraint while assigning based on preferences can be a soft constraint. To handle this use case, we can set the preference penalties high enough such that the cost of accepting a mismatch of skills in an assignment is higher than the total cost of any other schedule.

In some health care scheduling problems, patients do require visits less than 5 days a week, so that creating a weekly schedule is needed. There could be different use cases of this situation. If the patients specify the days they want to be visited, then daily solution methods can be applied. However, if the patients only specify the number of days they need to be visited in a week, then the visit days is also a decision to be made and a weekly schedule is needed. We can adjust our heuristic to handle such use cases. The patients in the network can be defined multiple times to represent the number of visits they need. Namely, if they need to be visited 2 days a week, then the patient node is doubled. The cluster generation algorithm can be adjusted to generate larger clusters with enough patients to be covered in 5 working days of an aide. Since the duplicate nodes have the same address, they would be chosen to be on the same cluster and visited by the same aide throughout the week. After the clusters are assigned to the aides, then TSP routes can be generated to find the sequence of the visits. Rearranging the sequence of the nodes in each cluster is needed to make sure that a patient is not visited twice on the same day, but rather visited by the same aide on different days.

In this work, we emphasize the consistency in the patient–aide assignments, but visit time consistency may also be desired in some settings. In this case, the regret operators explained in Sect. 4.2 can be used to create a schedule where the routes are kept mostly the same, and the new patients are inserted into the schedule. Since the travel times between the patients are different, simple insertion of the new patients to the empty spots may not generate a feasible solution. Thus, the routes would need to be recreated to achieve feasibility. To minimize the visit time deviation from previous visit times, a time consistency penalty would need to be applied. This way, consistency in both patient–aide assignments and visit times could be obtained. Moreover, we consider consistency between two consecutive periods in this study, but consistency over multiple periods can also be explicitly modeled if it is desired. In this case, the consistency constraint in the MIP model, or the consistency costs handled in the heuristic can be adjusted to consider the multiple periods in the past instead of only the last period.

5 Computational experiments

In this section, we describe the results of a computational study aimed at comparing the performance of the methods described in Sect. 4 and exploring the effect of the continuity of care requirement on scheduling decisions. For the purposes of this study, we obtained data from a real-world home health care agency operating in the USA. The experiments have been performed on an Intel(R) Core(TM) Processor i7-4785T (CPU 2.20 GHz) with 16 GB of memory, using the solver Gurobi 7.0.1 with a maximum time limit of 12 h for mixed-integer programming models. In the results, the computational times are expressed in seconds of CPU time.
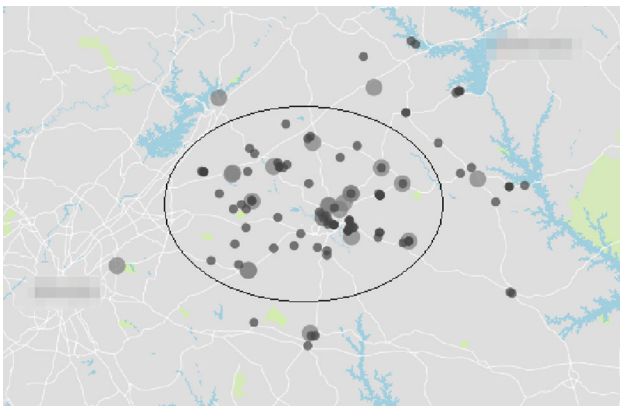
Fig. 1 Map of the real-world data. Circled area represents the dataset from which the "intense" data instances are sampled. Small markers represent patient locations; larger markers represent aide locations

5.1 Data

We address a real-world home health care scheduling problem faced by a hospice agency that operates in the USA. This hospice agency has operated for more than two decades and has branches in more than 20 states. We collaborated with one of the agency's branches and used their data as the basis for our computational experiments. This dataset, associated with one city-branch of the agency, contains information on 98 patients and 24 aides. All data has been masked in accordance with HIPAA requirements. The approximate locations of the patients and the aides are shown on the map in Fig. 1. The dataset is available at https://github.com/seymagk/HomeHealthcareScheduling.

Per government eligibility requirements, a patient is deemed hospice-eligible if he has an expected remaining lifetime of six months or less if his disease were to run its normal course. The objective of hospice care is not to provide curative care, but rather, to increase patient quality of life and provide the patient with end-of-life comfort and reduced pain. To increase patient and family comfort during this final phase in life, the agency asks the patients if they have any preferences regarding the aide who provides care. In our dataset, 18 of the 98 patients expressed preferences. These preferences yielded a list of aides that match the preferred characteristics expressed by the patients. Of the 18 patients with a list of preferred aides, 16 patients have a resulting preference list containing only a single aide; the remaining two patients have multiple aides in their preference list. 80 patients do not have any preferences. The agency wishes to respect patient preferences, to the extent possible.

The main operational costs include travel costs and the labor costs. Aides drive to the patients' homes and receive reimbursement from the company on a per-mile basis. Full-time aides receive wages in accordance with a 40 h work week as well as additional benefits; part-time aides receive hourly

wages. The costs and time parameters are listed in Table 7 in the Appendix. Travel distance and travel time between each location is calculated using Google Maps.

5.2 Instance generation

We generate smaller size instances from our full dataset, as described below. We consider three different geographical structures to be included in our computational experiments: (i) urban area, characterized by dense data points in the city center and sparse data points in the suburbs, (ii) city center, the center of the urban area with dense data points, and (iii) rural area, characterized by sparse data points. The small instances are characterized approximately by two fifths and three fifths of the number of patients in the full data set; these resulting datasets are similar in size to instances considered in the literature (see, for example, Cappanera and Scutellà 2014, Jensen 2012, Nickel et al. 2012). In addition to the full size instance, we generate 10 medium and 10 large size instances for each of these three network structures. In total, we generate 63 instances to test. The performance of the methods on medium size and large size instances is measured by taking the average performance over 10 instances in each category. We also create one set of small size instances with urban area structure to solve our Con-MIP models to optimality and compare the optimal solutions. Using this experimental setup, we aim to analyze the differences between the performance of our methods in different settings representing different geographical structures. Namely, we analyze how the sparsity of the network and the deviation of the distances between the nodes affect the performances of the methods.

The patients for the *urban area* instances are sampled uniformly at random from the full set of patients. If any of the randomly selected patients have aide preferences, these preferred aides are added to the dataset. The remaining aides are sampled uniformly at random from the full set of aides. In order to obtain a city center structure, we first identify a dense area including 74 patients and 18 aides in our full dataset (depicted in Fig. 1). To make this dataset more dense—so that it is representative of a *city center*—we halve the distances between all data points. The smaller instances are sampled uniformly at random from this set of 74 patients. After preferred aides are added to the dataset, the remaining aides are sampled uniformly at random from the set of 18 aides in the dense dataset. The rural area instances include data points that are far from each other. To obtain a *rural area* of n patients, we first cluster the full dataset using a hierarchical clustering (Jones et al., 2001) algorithm and obtain n clusters. In the hierarchical clustering algorithm, we used the maximum distance criteria (i.e., the maximum absolute value of the distances between points in different clusters). Then, one patient is chosen uniformly at random from each cluster and added to the dataset. The aides are selected as before, i.e.,

Table 3 Instances used in the computational experiments

# Instances	# Patients	# Aides	Instance name
Type 1: (Urban area, 32, 40) ∗			
1	98	24	u-98-24
10	60	15	u-60-15
10	40	10	u-40-10
20	16	4	u-16-4
20	18	5	u-18-5
Type 2: (City center, 16, 20)			
1	74	18	c-74-18
10	60	15	c-60-15
10	40	10	c-40-10
Type 3: (Rural area, 48, 60)			
1	98	24	r-98-24
10	60	15	r-60-15
10	40	10	r-40-10

∗ (Geometry, Average distance between two points (miles),Average time between two points (mins))

after the preferred aides are added, the remaining aides are sampled uniformly at random from the full set of aides. We also double the distances between all data points, to increase the sparsity. As a result, the full data set of 98 patients that is used as an *urban area* instance is also used as a *rural area* instance with an adjusted pairwise distances. The instances are listed in Table 3.

5.3 Results

We compare the performance of our Con-MIP formulation with Con-MIP-A and with the petal-based heuristic solution. In this computational study, our heuristic solution consists of the schedule obtained using the petal-based heuristic and improved using the combined single and double swap heuristic. We denote this solution as H throughout the paper. This section presents the results of both Con-MIP formulation and heuristics for Con-HHCSP, as well as consistency and cost trade-off analysis over shorter and longer time periods. Before presenting the detailed results of the computational study, we also compared the schedule generated by our heuristic methods for HHCSP with the schedule that is used by the agency in order to see the value of our methods in practice. In this comparison, we observed that the daily cost of operations is decreased by $3,250, which constitutes around 42% improvement over the current schedule in operation.

In this section, we present our results for Con-HHCSP methods. For this analysis, we generate an updated set of instances from our original instances (presented in Table 3). The updated dataset represents the data of the next period of the original dataset. To generate the updated datasets, we assume that 10% of the patients in the original dataset change

in the next period while the set of aides and data parameters in Table 7 remain the same. We sample uniformly at random from the existing set of patients to identify the patients that leave the dataset. The patients added to the dataset are sampled uniformly at random from the set of patients in the full dataset; patients are only added to the updated instance set if one of their preferred aides is in the existing set of aides. The total number of patients does not change when we update the instance set. We first update the schedules based on the updated dataset, using both Con-H and Con-MIP-A, and compare the cost of the schedules and the solution times of both methods.

We test the methods on 60 sets of medium and large instances having different network structures, as explained in Sect. 5.2. Further, we present our analysis of the impact of the inconsistency penalty on the cost of the schedule; this analysis sheds insight on the cost and consistency trade-off in this problem. Finally, we analyze the long-term cost effect of the consistency policy on the cost, by generating 50 updated dataset instances and updating the schedules over 50 periods. The results are presented below.

Comparing Con-MIP and the heuristic

We first compare the results of Con-MIP and the heuristics. The heuristic method, denoted by Con-H, consists of solving the problem using Con-PH and then applying the combined single and double swap heuristic to improve the schedule. Note that we didn't apply LNS heuristic here, as we observed a small gap in the results obtained after the swap heuristics. Also, since the solution time for Con-MIP is small, we wanted to keep it comparable for the heuristic as well. We present the gap of the heuristic method compared to the MIP model, which we quantify as $\Delta H^D = \frac{v_{\text{Con-H}} - v_{\text{Con-MIP}}}{v_{\text{Con-MIP}}} \times 100$. Let $v_{\text{Con-MIP}}$ be the cost of the schedule obtained with Con-MIP, and $v_{\text{Con-H}}$ be the cost of the schedule obtained via the heuristic method. We define ΔH^D as the percentage gap of heuristic method compared to the MIP model. For each instance set, we take the average of the ΔH^D values of 10 instances within the instance set. Note that here we set the inconsistency penalty (λ) equal to half of the preference penalty, i.e., $\lambda = \lambda^P/2$.

Table 4 presents the comparison between the heuristic (Con-H) and the MIP formulation (Con-MIP). We observe that the heuristic performs well compared to the MIP formulation on average in all instances, with gaps of less than 4.2% in all instances. One interesting observation is the solution times. Although we would usually expect a MIP model take more time, it is different in the consistent version of the problem. In half of the instances, Con-MIP is faster than the heuristic due to the consistency requirement, which eliminates many symmetric solutions. We present the mean and the median solution times for Con-MIP. Average solution time is significantly higher due to a couple of outlier instances, where Con-MIP is not able to achieve an optimal solution

Table 4 Results: The heuristic Con-H (petal-based heuristic and combined single and double swap heuristics) vs. Con-MIP

Instance	Con-MIP-A (1 h) ΔH^D (%) (avg,min,max)	Gap (%)	Mean $t_{\text{Con-MIP}}$	Median $t_{\text{Con-MIP}}$	Median $t_{\text{Con-H}}$
c-40-10	(0.67,0.0,2.6)	12	729	9	199
c-60-15	(1.22,0.2,3.7)	0	48	20	548
r-40-10	(0.72,0.0,4.1)	2	412	65	214
r-60-15	(0.75,0.0,3.6)	0	377	250	625
u-40-10	(0.63,0.1,2.3)	0	15	15	203
u-60-15	(1.08,0.0,3.0)	0	72	61	552

Run times are reported in seconds. $t_{\text{Con-MIP}}$: Con-MIP Solution Time, $t_{\text{Con-H}}$: Con-H Solution Time

even after 1 h. Depending on the instance, Con-MIP solution time can be significantly high, while the heuristic solution times are more consistent. Another observation is that the center instances have the smallest solution time, and the rural instances have the largest. This can be explained by the distance matrix, where the center instances have the smallest distances between the nodes. Recall that in these instances, only 10% of the patients in the data set are replaced with new set of patients, so that the majority of the schedule can remain the same. This fact decreases the solution time for Con-MIP. The results for patient set changes which are higher than 10% are presented in Table 5. Since the solution time for Con-MIP is small, we don't report the results for Con-MIP-A. However, for certain instances or for different parameters (e.g., smaller inconsistency penalty or higher rate of change in the dataset), where the performance of Con-MIP may be lower, Con-MIP-A can be utilized.

Sensitivity analysis for the change in the dataset

We analyze the effect of the percentage change in the dataset on the performances of Con-H and Con-MIP-A. Based on the information given by the home health care company, the percentage change in the dataset ranges between 10 to 60%. We design an experiment accordingly, with instance sets of u-40-10 and u-60-15, where we generate 10 instances for each of the different change percentages. The results are presented in Table 5. We observe that Con-H outperforms Con-MIP-A on average in all of the instance sets, and has a reliable performance with positive improvement over Con-MIP-A in majority of the instances. We also observe less variability in terms of the solution time for Con-H. As expected, the solution time for Con-MIP-A increases as the change percentage increases. This is due to the fact that a larger part of the schedule needs to be recreated as the change percentage increases.

Sensitivity analysis for the cost vs. consistency trade-off

Next, we analyze how the consistency and cost are affected by the use of different inconsistency penalties in Con-PH. Here, consistency is measured by the number of patients assigned to a different aide in the new schedule as a percentage of the total number of patients who are included both in the previous and the new datasets. Here, we assume
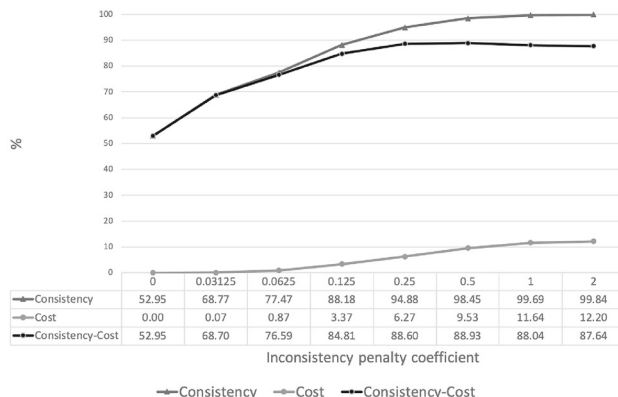
Fig. 2 The consistency–cost trade-off, as the inconsistency penalty increases (average over all instances)

that 20% of the patients in the original dataset are different in the new dataset. The inconsistency penalty (λ) is obtained by multiplying the inconsistency penalty coefficient (c_λ) and the penalty for the preference violation (λ^P), i.e. $\lambda = c_\lambda \lambda^P$, where the preference penalty is determined based on the average distance ($avg(D)$) and average travel time ($avg(T)$) between all the points: $\lambda^P = 2 * (c_T * avg(D) + c_{OF} * avg(T))$. When we set $c_\lambda = 0$, we obtain a schedule without consistency restriction. We compare the cost of the schedules we obtain with positive $c_\lambda = \lambda$ values (defined as v^λ_{Con-PH}) to the cost of the schedule we obtain with $c_\lambda = 0$ (defined as v^0_{Con-PH}) as follows:

$$\Delta C_\lambda = \frac{v^\lambda_{Con-PH} - v^0_{Con-PH}}{v^0_{Con-PH}} \times 100$$

We present a summary of the results in Fig. 2 for the percentage change in the costs of the schedules as well the percentage change in the consistency of the schedules, where the x-axis represents different λ penalty values and the y-axis represents the percentage change. For each instance set, we take the average of the ΔC_λ values of 10 instances within the instance set, in order to find the average percentage change in the cost of the schedules for each penalty. Similarly, we present the average consistency change, which is defined as

Table 5 Results: The comparison of heuristic Con-H (petal-based heuristic) and Con-MIP-A for different percentage changes in the patients dataset

| Change (%)	ΔH^A(%) (avg, min, max)	Gap (%)	Mean $t_{\text{Con-MIP}}$	Mean $t_{\text{Con-H}}$
	urban-40-10			
10	(4.36, 18.79, 6.03)	0.00	15	202
20	(5.22, 1.37, 9.24)	0.00	243	210
30	(4.49, 1.90, 9.01)	0.41	406	236
40	(5.85, 1.06, 11.52)	0.28	960	263
50	(3.80, −1.76, 8.79)	1.46	2037	308
60	(3.40, −2.02, 8.37)	2.44	2481	334
	urban-60-15			
10	(3.37, −1.71, 10.57)	0.00	72	546
20	(4.60, 2.88, 5.85)	0.99	1267	586
30	(5.65, 1.54, 8.86)	1.98	2134	680
40	(3.24, 0.45, 6.87)	2.15	2850	761
50	(4.95, −0.49, 12.38)	6.31	3602	883
60	(4.79, 2.83, 9.53)	6.47	3049	980

All run times are reported in seconds. $t_{\text{Con-MIP}}$: Con-MIP Solution Time, $t_{\text{Con-H}}$: Con-H Solution Time

the average percentage change in the number of patients assigned to a different aide as a function of the λ penalty. As observed, both the consistency and the cost increase as the inconsistency penalty coefficient is increased, illustrating the trade-off between cost and consistency. We also present the percentage difference between consistency and cost, and choose the value, among the values we try, that maximizes this difference, which is $c_\lambda = 0.5$. This means that for a given cost, the gain from the consistency is maximized, i.e., the consistency and the cost trade-off is experienced less, at $c_\lambda = 0.5$. This result highlights the importance of the selection of the inconsistency penalty. Note that, in different settings, with different time and cost parameters, this value might differ. Our results indicate that the best inconsistency penalty needs to be found for each setting. The detailed results of this analysis are presented in the Online Appendix.

For an inconsistency penalty of 0.5, the percentage difference between the consistency and the cost for all instances are shown on Fig. 3. As we observe, the percentage difference between the consistency and the cost is above 80% for 93.33% of the instances. Also, we see a balanced distribution of difference values among all instances. Therefore, for our long-term analysis presented below, we use the inconsistency penalty value of 0.5.

Long-term effect of the consistency policy:

Finally, we analyze the long-term effect of the consistency policy. In other words, we analyze how the cost is affected if we continue updating the schedule with the consistency constraints over the long term. For each instance, we generate 25 updated instances by changing the set of patients by 10% at each period. In the analysis presented above, we found that $c_\lambda = 0.5$ is the best inconsistency penalty coefficient to choose in order to decrease the cost and consistency
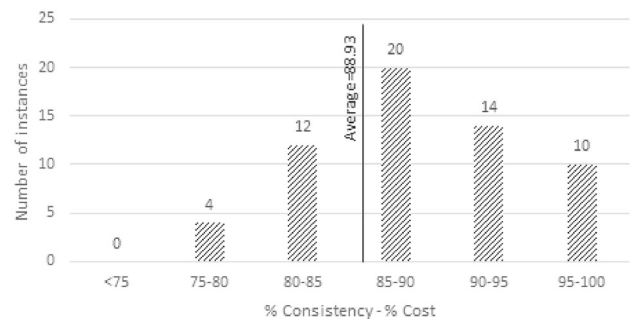

Fig. 3 The percentage difference between the consistency and the cost (excluding the inconsistency penalty), for the inconsistency penalty coefficient of 0.5

trade-off. Therefore, in this long-term study, we set $c_\lambda = 0.5$, and update the schedule 25 times. At each period t, we compare the cost of the updated schedules with the cost of the schedules obtained when $c_\lambda = 0$ as follows:

$$\Delta C^t = \frac{v_{Con-PH}^{0.5,t} - v_{Con-PH}^{0,t}}{v_{Con-PH}^{0,t}} \times 100$$

Figure 4 displays the change in overall cost over time when we update the schedule with the consistency preferences (which keeps the schedule of period t similar to the schedule of the previous period, $t - 1$). We observe that the cost difference, (ΔC^t), increases in the first eight updates; after eight updates, the cost difference appears to stabilize with some positive or negative deviations around an average percentage value. We note that on average, the cost trade-off is experienced less in areas with a city center structure. This trend may be explained by the fact that in city center structures all locations are relatively close to each other
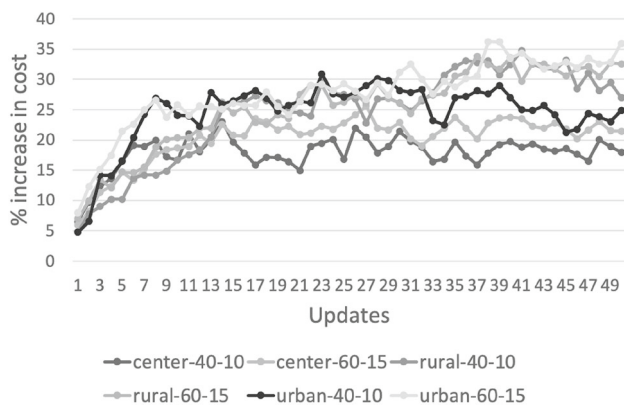
Fig. 4 The cost trade-off (ΔC^t) over 50 periods

(as compared with the other structures we studied). Therefore, the cost difference between assigning the optimal aide and assigning the aide imposed by the consistency policy is smaller since aides' locations are closer to each other. Small distances between the locations affect both the travel cost as well as time-related costs, resulting in a smaller total cost. According to the results, we conclude that the cost of updating the schedule increases for a while, peaks at a certain period and remains at the peak value for the remainder of the periods. Therefore, depending on this percentage increase in cost, it may be a better strategy to update the schedule for a few periods and then create a new schedule from scratch. When to create a new schedule may be determined based on the company's ability and willingness to absorb this cost increase.

6 Conclusions

In this paper we have addressed a daily home health care planning problem; we proposed solution methods for Con-HHCSP: a daily scheduling problem where the new schedule to be created needs to be consistent with the current schedule in use. We presented the computational results of the proposed solution methods for Con-HHCSP. To the best of our knowledge, this work is the first to introduce Con-HHCSP and propose solution methods and analysis to dynamically handle a consistency policy.

Our computational results reveal the potential of the proposed Con-MIP-A and heuristic methods in addressing home health care problems. While Con-MIP can achieve optimality in a reasonable time for some of the parameters (e.g., inconsistency penalty, the rate of change in patient dataset), MIP model can get intractable for mid to large size instances, or for different parameters of Con-HHCSP. In those cases, it is possible to find good feasible solutions by solving the approximate MIP model. Further, better solutions may be found with the heuristics in a much shorter time. The effect of

swap heuristics was not negligible and they provided a very quick way to further improve our solutions. Although we observe an improvement in the solution when LNS heuristic is applied as the final improvement heuristic, a non-negligible increase in the computation time should be noted. We compared the schedule created by our heuristic methods with the schedule that is used by the home care agency with whom we are collaborating. With our methods, the daily cost of operations is decreased by \$3,268, which constitutes around 43% improvement over the current schedule in operation.

Based on our analysis of the continuity of care policy in Con-HHCSP, we observed that there is a significant trade-off that must be considered between cost and continuity of care. As the continuity of care policy is applied more strictly, the resulting cost increases. Therefore, the selection of the inconsistency penalty coefficient, which determines the strictness of the continuity of care policy, is critical. Moreover, the effect of the continuity of care policy on the cost may grow large in the long-term. Our long-term analysis revealed a rapidly increasing trend in the cost trade-off over time in the first eight periods, and then a stabilized trend for the remainder of the periods. This analysis should be used as a basis for deciding how many times to update the existing schedule and when to solve it from scratch.

One interesting result obtained in this study is that the computation time for the MIP models of Con-HHCSP is lower when the change in the dataset is small. This is expected, since most of the schedule remains the same. This introduces another research question: would it be a good approximation to solve Con-HHCSP repetitively instead of solving a MIP model to create a new schedule from scratch for a longer planning horizon? The complexity of the problem increases as the planning horizon increases, so that obtaining a good quality solutions becomes more difficult. These results suggest that, in order to handle continuity of care requirements, solving Con-HHCSP may provide a good approximation to an exact solution for a more complex multi-period problem. We plan to analyze this further in our future studies. Other directions for future research are different variants of the problem, such as skilled nurses, time windows, and interdependent visits. We believe that the methodology we propose in this paper is adaptable to different use cases of the problem.

Appendix A: Set partitioning model

See Table 6

Table 6 Notation for the set partitioning model

Index sets	
P	Patients
A	Aides
C	Clusters
Data parameters	
w_{ac}	The cost of assigning cluster $c \in C$ to aide $a \in A$
Variables	
x_{ap}	Assigning patient $p \in P$ to aide $a \in A$
x_{ac}	Assigning cluster $c \in C$ to aide $a \in A$

$$\min \quad \sum_{a \in A} \sum_{c \in C} w_{ac} x_{ac}$$

$$\text{s.t.} \quad \sum_{a \in A} x_{ap} = 1 \qquad \forall p \in P$$

$$\sum_{c \in C} x_{ac} = 1 \qquad \forall a \in A$$

$$\sum_{a \in A} x_{ac} \leq 1 \qquad \forall c \in C$$

$$\sum_{c \ni p} x_{ac} = x_{ap} \qquad \forall p \in P; \forall a \in A$$

$$x_{ap}, x_{ac} \in \{0, 1\} \qquad \forall a \in A; \forall p \in P; \forall c \in C$$

Appendix B: Data

See Table 7

Table 7 Cost & Time parameters

Item	Value
Cost of transportation per mile driven (c_T)	$3
Full-time hourly labor cost (c_F)	$20
Part-time hourly labor cost (c_P)	$15
Penalty cost for exceeding B hours in a week for PRNs (c_{OP})	$ 2000
Hourly penalty associated with overtime work for full-time aides (c_{OF})	$5
Weekly overuse threshold forPT-HHA (OP_p)	30 h
Daily overuse threshold forFT-HHA (OF_f)	8 h
Length of regular workday (H)	8 h
Maximum weekly total hours of PTs before penalty (B)	30 h
Workday length factor with overtime (α)	1.25
Maximum length of workday with overtime	10 h

Appendix C: Literature review

See Table 8

Table 8 Summary of the relevant papers in the literature

Planning horizon	Patient–aide assignment preferences	Continuity of care	Schedule updating with consistency
Bennett and Erera (2011)			
Weekly	Single nurse assignment	N/A (there is a single nurse)	Consistency in visit times
Cappanera and Scutellà (2014)			
Weekly	Skilled nurses (hard constraints)	Limited number of aides for a patient	N/A
Nickel et al. (2012)			
Weekly and daily	Skilled nurses (hard constraints)	Limited number of aides for a patient	Consistency in visit times
Jensen (2012)			
Several weeks	Skilled nurses (hard constraints), preferred nurses (soft constraints)	Limited number of aides for a patient	Consistency in visit times
Carello and Lanzarone (2014)			
Daily	Skilled nurses (hard constraints)	Limited number of aides for a patient	N/A
Yalçındağ et al. (2016)			
Daily	Skilled nurses (hard constraints)	The same aide for a patient	N/A
Braekers et al. (2016)			
Daily	Skilled nurses (hard constraints), preferred nurses (soft constraints)	N/A	N/A
Rasmussen et al. (2012)			
Daily	Preferred nurses (soft constraints)	N/A	N/A
Heching et al. (2019)			
Weekly	Skilled nurses (hard constraints)	Same aide for a patient	Fixed variables
Our study			
Daily	Part-time and full-time nurses (hard constraints on work hours), preferred nurses (soft constraints)	The same aide for a patient: Handled via a rolling horizon approach	Consistency in patient–aide assignments

References

Beasley, J., Krishnamoorthy, M., Sharaiha, Y., & Abramson, D. (2004). Displacement problem and dynamically scheduling aircraft landings. *Journal of the Operational Research Society, 55*, 54–64.

Bennett, A. R., & Erera, A. L. (2011). Dynamic periodic fixed appointment scheduling for home health. *IIE Transactions on Healthcare Systems Engineering, 1*, 6–19.

Bent, R., & Van Hentenryck, P. (2004). A two-stage hybrid local search for the vehicle routing problem with time windows. *Transportation Science, 38*, 515–530.

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. (2014). Joint assignment, scheduling, and routing models to home care optimization: A pattern-based approach. *Transportation Science, 49*, 830–852.

Carello, G., & Lanzarone, E. (2014). A cardinality-constrained robust model for the assignment problem in home care services. *European Journal of Operational Research, 236*, 748–762.

Castillo-Salazar, J. A., Landa-Silva, D., & Qu, R. (2014). Workforce scheduling and routing problems: Literature survey and computational study. *Annals of Operations Research, 239*, 1–29.

Coelho, L. C., Cordeau, J.-F., & Laporte, G. (2012). Consistency in multi-vehicle inventory-routing. *Transportation Research Part C: Emerging Technologies, 24*, 270–287.

Fathollahi-Fard, A. M., Hajiaghaei-Keshteli, M., & Mirjalili, S. (2020). A set of efficient heuristics for a home healthcare problem. *Neural Computing and Applications, 32*, 6185–6205.

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

Foster, B. A., & Ryan, D. M. (1976). An integer programming approach to the vehicle scheduling problem. *Journal of the Operational Research Society, 27*, 367–384.

Grieco, L., Utley, M., & Crowe, S. (2020). Operational research applied to decisions in home health care: A systematic literature review. *Journal of the Operational Research Society, 72*(9), 1–32.

Groër, C., Golden, B., & Wasil, E. (2009). The consistent vehicle routing problem. *Manufacturing & Service Operations Management, 11*, 630–643.

Gutiérrez, E. V., & Vidal, C. (2013). Home health care logistics management: Framework and research perspectives. *International Journal of Industrial Engineering and Management, 4*, 173–182.

Haggerty, J. L., et al. (2003). Continuity of care: A multidisciplinary review. *BMJ British Medical Journal, 327*, 1219.

Heching, A., Hooker, J. N., & Kimura, R. (2019). A logicbased benders approach to home healthcare delivery. *Transportation Science, 53*, 510–522.

Hulshof, P. J., Kortbeek, N., Boucherie, R. J., Hans, E. W., & Bakker, P. J. (2012). Taxonomic classification of planning decisions in health care: a structured review of the state of the art in OR/MS. *Health Systems, 1*, 129–175.

Jensen, T. S. (2012). Application of metaheuristics to reallife scheduling problems. Philosophiae Doctor thesis, Department of Mathematics and Computer Science, University of Southern Denmark, Odense, Denmark.

Jones, E., Oliphant, T., Peterson, P., et al. (2001). SciPy: Open source scientific tools for Python, Package: Hierarchical clustering (scipy.cluster.hierarchy) [Online; accessed 2017-05-02]. http://www.scipy.org/.

Kovacs, A. A., Golden, B. L., Hartl, R. F., & Parragh, S. N. (2014). The generalized consistent vehicle routing problem. *Transportation Science, 49*, 796–816.

NHPCO. NHPCO's Facts and Figures - Hospice Care in America tech. rep. (National Hospice and Palliative Care Organization, 2016).

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.

Potvin, J.-Y., & Rousseau, J.-M. (1993). A parallel route building algorithm for the vehicle routing and scheduling problem with time windows. *European Journal of Operational Research, 66*, 331–340.

Rasmussen, M. S., Justesen, T., Dohn, A., & Larsen, J. (2012). The home care crew scheduling problem: Preferencebased visit clustering and temporal dependencies. *European Journal of Operational Research, 219*, 598–610.

Subramanyam, A., & Gounaris, C. E. (2017). A decomposition algorithm for the consistent traveling salesman problem with vehicle idling. *Transportation Science, 52*(2), 386–401.

Wong, R.T. (2008). In the vehicle routing problem: Latest advances and new challenges pp. 475–485 (Springer, 2008).

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.

Yalçındag, S., Matta, A., & Sahin, E. (2011). *Human resource scheduling and routing problem in home health care context: a literature review*. UK: ORAHS/Cardiff.