Home Health Care Scheduling: A Case Study

Zhi Yuan · Armin Fügenschuh

Abstract This article provides a case study on the problem of scheduling nurses for home health care on a weekly basis. The list of health care tasks are available before the start of the week. Each client may require multiple visits per week, and they can specify the preferred day and the time window on each preferred day that they expect a visit. Besides, certain visits require a minimum number of days' difference in between. The nurse may also specify the preferred working day and the maximum working hours. The optimal schedule to be found should minimize the personnel cost as well as the total working time, without compromising the service quality. This problem is a combination of the staff rostering problem that consists in assigning health care tasks to a competent nurse on the appropriate day without exceeding her maximum working hours, and the vehicle routing problem with time windows, where an optimal route for each day's scheduled visits should be found respecting the time windows. We formulate this problem as an integer linear programming model based on the multi-commodity network flow formulation, and develop also problem specific greedy construction and local search approaches for it. The scalability of different approaches are studied, and a real-world instance is used for validating our proposed approach. An estimate of at least 10% cost reduction potential is observed comparing with the current manual plan.

1 Introduction

The health care service system in Germany and many other countries is facing increasing costs due to the aging population. In this work in cooperation with a local health care service provider in medium-size town (150,000 inhabitants) in Germany, we focus on the specific field of home health care, i.e., visiting and providing medical services to clients at home. These medical services range from cleaning, personal hygiene to some medical treatments, including blood pressure measuring, medication prescription, injections and so on.

The home health care scheduling can be regarded as a combination of staff rostering problem [10] and a vehicle routing problem (VRP) with time windows [8]. On the one side, one has to assign weekly visits to nurses with sufficient competence, following the preferred

Zhi Yuan and Armin Fügenschuh
Professorship of Applied Mathematics, Department of Mechanical Engineering, Helmut Schmidt University / University of the Federal Armed Forces, Hamburg, Germany
E-mail: {yuanz,fuegenschuh}@hsu-hh.de

days specified by both the nurses and clients, without exceeding the nurse maximum workload. On the other side, the workload or the working time of each nurse depends on how fast one can arrive from one client's home to another client's home within each client's specified time slots on each of their preferred days. Besides, different from most of the existing home health care scheduling problems in the literature, our scheduling task is on a weekly basis. This is because certain medical treatments require a minimum day difference in between. For example, the insulin injections of certain clients need to be given twice per week, with three days difference in between. This inter-visit day difference is also considered in our optimization process.

In this preliminary case study, the optimization objective is to cover all the weekly client visits with minimal operating cost, especially the personnel size, and also to minimize the nurses' total workload by an effective routing procedure. This should be achieved without the loss of the service quality, for example, sufficient treatment duration should be guaranteed, and the client-specified day and time slots should be obeyed as hard constraint.

This weekly based home health care scheduling problem is formulated as an integer linear programming (ILP) model as a multi-commodity flow problem. Besides, a problem specific greedy construction method is developed and a local search procedure is applied to further improve the constructed solution. The parameters of the algorithm are automatically adapted during the algorithm run, to obtain an instance-specific best parameter setting. Furthermore, the solution found by the heuristic approach is also input as an initial upper bound for a commercial ILP solver to speed up the branch-and-bound process. Real-world instances with up to 90 patients and 460 weekly treatments have been used to validate our approach.

The rest of the article is organized as follows. Section 2 provides a brief literature overview on the home health care scheduling problem. This problem will be described in more details in Section 3, and will be formulated as an ILP model in Section 4. The primal heuristics are presented in Section 5. Section 6 is dedicated to the experimental setup and the computational results for the real-world instances from our industrial partner. Finally, Section 7 provides some concluding remarks and discussions for potential future directions.

2 Literature Review

In general, the home health care scheduling problem belongs to a broader class of problems called workforce scheduling and routing problem, see for example [5] for a literature survey. Such problems can be regarded as a combination of staff rostering problem [10] and a vehicle routing problem (VRP) with time windows [8]. Particularly from the staff rostering aspect, it has many things in common with the nurse rostering and scheduling problem [6, 4] in terms of skill category, shift type, and time related constraints. However, the home health care scheduling problem has the further requirement of a routing task from patient to patient. There exists also works that focus only on the routing part of the home health care scheduling problem, for example, Kergosien et al. [14] modelled it into a multiple traveling salesman problem, and solved the resulting integer linear programming model by a branch-and-cut approach.

Begur et al. [2] presented the use of a spatial decision support system to schedule and route home health care nurses in Birmingham, Alabama, USA. The system integrates geographic information system with scheduling heuristics and databases, and it is reported an over 20,000 US Dollars saving annually for travel expenses and scheduling preparation, and

it helps improve the balance of work among nurses. A saving-type route-building heuristic is proposed and described, and a route improvement is done manually through a visual interactive system.

Cheng and Rich [7] formulated the home health care scheduling problem as a multiple depot vehicle routing problem with time windows. Both full-time and part-time nurses are considered, and each nurse starts and ends their daily service from their home. The lunch break problem is considered by adding an additional lunch node into the scheduling graph. Two mixed integer programming models were presented, and were reported to be able to solve by CPLEX problems of size up to 10 patients. A two-phase heuristic is also developed, using a randomized greedy algorithm to construct a possibly infeasible solution in the first phase, and then improve it by a problem-specific local search in the second phase.

Eveborn et al. [11] introduced a decision support software, developed to aid the staff planner creating daily schedules at a home health care organization in Sweden. Various practical constraints such as staff competence, time windows for visits, or breaks for meals were considered. A mixed-integer programming formulation is given, based on a set partitioning model. As solution method, they make use of repeated matching approach, which maps the staffs-to-home-visits problem into a matching problem, and then solves the resulting matching problem with an exact method or a repeated assignment heuristic. They estimate a total cost saving of 20%, and a reduction of 7% of the total working time.

Rasmussen et al. [22] further presented the home care crew scheduling problem in Denmark. The scheduling is also on a daily basis with temporal dependencies and some other service oriented constraints. One example of such temporal dependencies given by the authors is this: A first home carer switches on a washing machine, and a second home carer should come and empty the washing machine after two to four hours. In such case, the routing part of the problem amounts to the VRP with coupled time windows [13]. The other service oriented constraints include, for example, leaving as few visits uncovered as possible. Each visit is associated with a priority, and if some visits may have to be rescheduled or cancelled, it should not cancel the most important visits of the day. The problem is formulated as a set partitioning problem and a branch-and-price algorithm is developed for its solution.

Koeleman et al. [16] consider a home health care scheduling problem in a stochastic setting. They assume the patients arrive according to a Poisson distribution, and the probability distribution of to which class the patients belong to, and their health care duration per week, is assumed to be known. Then the decision has to be made whether they should accept, reject, or put the patient in a waiting list. This problem is modelled as a Markov decision process, and the (near-)optimal policy can be found by a trunk reservation heuristic. However, no timetabling and routing of the nurses to patients have been considered.

Bertels and Fahle [3] also stated that the home health care scheduling problem is a combination of a staff rostering and a vehicle routing problem. They developed also a hybrid algorithm that consists of linear programming, constraint programming and metaheuristics for solving the problem.

Most of the existing literature as listed above schedules home health care on a daily basis. There exists only few works that schedules in a multi-day horizon, including Nickel et al. [21] and Di Gaspero et al. [9]. Both works formulated the problem as constraint programming (CP) models, and then developed large neighborhood search metaheuristics upon the CP models. The main difference from the application perspective between the both works is that Nickel et al. [21] required the same patients to be visited by the same nurse, while this requirement is relaxed in [9].

To the best of our knowledge, none of the presented models are completely applicable to our problem, mainly for some or all of the following reasons. Our schedule is on a weekly basis, and the inter-visit day difference should be integrated into the optimization process [23]. In the literature, e.g. [2], a two-visit-per-week patient will be fixed to a Monday and Thursday or a Tuesday and Friday before the optimization process starts. Although such fixed assignments simplifies the problem, it also reduced scheduling flexibility and efficiency. In this work, we model inter-visit day difference as a constraint, and let the optimization process to determine the treatment day. Besides, not only the daily maximum working time but also the total weekly maximum working time can be imposed for each nurse. Furthermore, each nurse as well as the patient should be able to specify their individual preference on working days in advance.

3 Problem Description

For notational simplicity we will refer to the employees of the service as *nurses*, the clients as *patients*, the service activities that the clients require as *treatments*. Our task is to develop computer software to assist the health care provider to generate a nurse schedule on a weekly basis.

The optimization task is to assign each patient or treatment to a nurse with competent qualification on an appropriate day at a time within a patient-specified time window.

Patients and treatments. A patient may require a number of nurse visitations, or treatments, in a week. A treatment is a medical activity to be performed by a visiting nurse at a proper time on an appropriate day. These treatments include washing, cleaning, bandage changing, medication prescription, and giving injections. The duration of each treatment is fixed and given in advance, but the start time of each visit can be varied within a patient-specified time window.

Day preference and time windows. Each patient can specify which days are preferred for his weekly treatments, e.g., only Monday and Tuesday, or no treatments on Thursday, etc. Furthermore, on each of these preferred days, a time window within which treatment can start can be specified by the patient, e.g., the cleaning can start on Monday from 9:00 to 11:00 or Thursday from 8:00 to 9:00. Note that the time windows imposed here are hard constraints, which means that if the nurse is scheduled to arrive earlier than the given lower bound of the time window, he or she would have to wait.

Inter-visit day difference. For some patients who need several visits per week, there may be some pairs of visits between which at least one to three days' difference must be retained. Examples of such pairs of treatments include some injections that must be given twice a week with at least three days break in between.

Nurses. There are mainly two types of nurses distinguished by our application partner, namely, the professional nurse and the assistant nurse, differing in qualification and also salary cost. Becoming a professional nurse in Germany requires three years' education, and one must pass the national exam to obtain the medical certificate. These nurses are able to perform all types of treatments. The assistant nurse usually receives a short-term on-the-job training. They can perform simple treatments such as cleaning or changes of bandages, etc. However, they are not allowed to perform treatments such as prescriptions or injections. The assistant nurses' salary is also lower than the professional nurses' salary, and it is estimated to be 70% of what the professional nurses receive.

Working date and time. Nurses' everyday work starts at 7 a.m. in the headquarter of the organization with a car, and the car must return to the headquarter by the end of the

day's work. According to the organization's legal regulation, the daily working hour of each nurse cannot exceed 6 hours, while the total weekly working hours for each nurse cannot be over 30 hours. Each nurse can also choose on which day (normally working days from Monday to Friday) and at most how many hours she is willing to work on a day or within a week. Working time includes the treatment time that a nurse spends at a patient, the driving time between patients and from or to the headquarter, as well as the waiting time, if she arrives at a patient too early.

Goals and objectives. In this work, our objective is to reduce the operating cost and the nurses' workload without compromising the service quality, e.g., the treatment duration, patient-specified date and time and certain required treatment qualification need to be kept. Our primary optimization goal is to reduce the personnel cost, i.e., to use as few staff as possible to carry out all the tasks. The secondary optimization goal is to reduce the total working time of all nurses.

4 Mathematical Models

We formulated the home health care scheduling problem as an integer linear programming (ILP) model based on the multi-commodity flow model with the Miller-Tucker-Zemlin constraints to handle the time consistency [19].

4.1 A graphical example

A graphical example of the home health care scheduling problem is shown in Figure 1. Each depot node is depicted as a triangle, $d^{i,j}$, and represents a nurse i on a day j. Each circle represents a treatment node. Here, nurse 1 carries out treatments t_1, t_2, and t_5 on day 1, and carries out treatment t_4 on day 2. Nurse 2 has treatments t_6 and t_3 on day 1, and has a free day on day 2. We also follow the terms used in multi-commodity flow, meaning that the arcs from a depot node to a treatment node, e.g., from $d^{1,1}$ to t_1, are called *pull-out arcs*, the arcs between two treatment nodes, e.g., t_1 to t_2, are called *deadhead arcs*, and the arcs from a treatment node to depot node, e.g., t_5 to $d^{1,1}$, are called *pull-in arcs*.

4.2 Sets

We denote the set of all nodes with N. There are two types of nodes in the example shown in Figure 1, namely, depot nodes and treatment nodes.

- N_{dp}: the set of depot nodes where a nurse starts and ends his or her day at;
- N_{trm}: the set of treatment nodes that should be visited by one of the nurses.

Similarly, we denote the set of all arcs with A, and distinguish three types of arcs existing in our graphical example, namely, pull-out arcs, deadhead arcs, and pull-in arcs.

- $A_{pullout}$: a pull-out arc starts from a depot node and ends at a treatment node;
- $A_{deadhead}$: a deadhead arc starts and ends both at a treatment node, representing a trip from one patient to another patient;
- A_{pullin}: a pull-in arc starts from a treatment node and ends at a depot node;

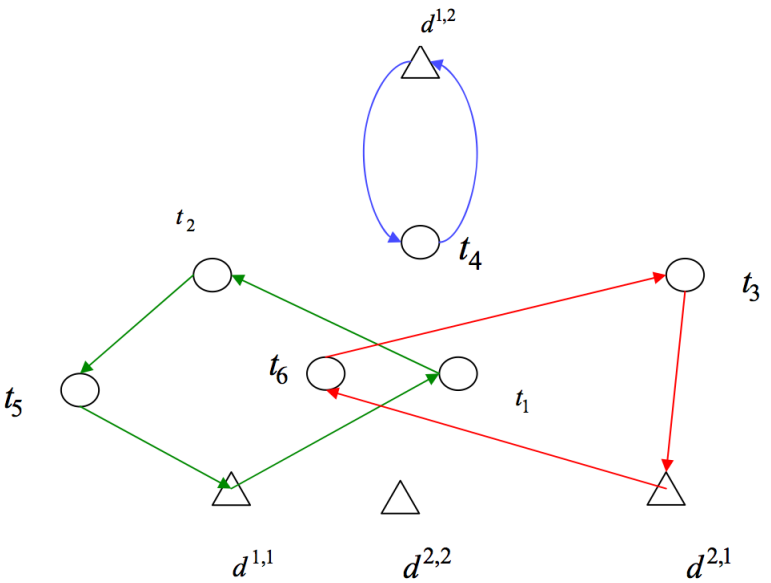

Fig. 1 A graphic example of the home health care problem. Triangles represent depot nodes, where node $d^{i,j}$ represents a nurse i on a day j. Note that some nurse may be totally free on some day. Circles represent treatment nodes with its index. Each tour must start and end at a depot node.

Apart from the sets that are visible in the figure, there are some other sets outside the graph, such as

- K: the set of nurses;
- D: the set of working days, i.e., $D := \{1,\ldots,5\}$, representing Monday to Friday. Note that we consider each nurse per day as a depot in our model;
- $R \subset N_{trm} \times N_{trm}$: the set of pairs of treatments that are related, in the sense that if two treatments i and j are related, $(i,j) \in R$, then treatments i and j must have some days' difference between each other.

4.3 Parameters

The parameters used in the ILP model are listed below:

- c^k_{start}: the starting cost of each nurse $k \in K$.
- $u^{k,d}_{i,j}$: the upper bound for the flow capacity on each arc $(i,j) \in A$ for a nurse $k \in K$ on a day $d \in D$. The value of u is binary, since each treatment node must be visited once and only once, and each depot can deploy only one unit of flow. Besides, we also use this parameter to remove some infeasible arcs in preprocessing, e.g.,
  1. if a treatment $i \in N_{trm}$ cannot be taken by a nurse $k \in K$, then we set all the arcs that are incident to node i to be infeasible by assigning their upper bounds to 0, i.e., $u^{k,d}_{i,j} := 0, u^{k,d}_{j,i} := 0, \forall j \in N, d \in D$.
  2. If a treatment $i \in N_{trm}$ cannot be taken on a day $d \in D$, similarly we set the upper bounds of all arcs incident to node i on day d to 0.
  3. If a nurse $k \in K$ is not possible to work on day $d \in D$, then all arcs on this layer are set to infeasible, $u^{k,d}_{i,j} := 0, \forall (i,j) \in A$.

- $\underline{t}_i$ and $\bar{t}_i$: the lower bound and the upper bound of the time window for a treatment i.
- $\delta_{i,j}$: the necessary duration from the start of treatment i until the start of treatment j, if j is taken by the same nurse on the same day subsequently after i. It consists of two parts, the service time for treatment i, and the trip duration from i to j, i.e., $\delta_{i,j} = \delta_i^{service} + \delta_{i,j}^{deadhead}$.
- $T_{day}^{k,d}$: the daily maximum working time of a nurse k on day d.
- T_{week}^{k}: the weekly maximum working time of nurse k.
- $\sigma_{i,j}$: the necessary day difference between two related treatments $(i,j) \in R$. The value of σ is typically restricted by $1 \leq \sigma \leq 3$.

4.4 Variables, Objective, and Constraints

The decision variables are listed as follows:

- $x_{i,j}^{k,d} \in \{0,1\}$: the flow variable representing whether an arc $(i,j) \in A$ is traveled by a nurse $k \in K$ on day $d \in D$;
- $t_i \in \mathbb{Z}^+$: the starting time of a treatment $i \in N_{trm}$;
- $s^k \in \{0,1\}$: binary variable to indicate whether a nurse $k \in K$ has been deployed.
- $\tau^{k,d} \in \mathbb{Z}_0^+$: the working time of a nurse k on a day d minus the starting time of 420, i.e., 7 a.m. If a nurse on a day does not start any tour, then its value equals to 0.

The objective function consists of two hierarchical parts. Firstly, in this work, the primary objective for the organization is to reduce the personnel cost (without compromising the service level, e.g., service time and nurse qualification). Imprecisely speaking, the goal is to use as few nurses as possible to take care of all the clients. Note that nurses with different qualifications may require different starting cost, then it is the total starting cost to be minimized. Secondly, at a subsidiary level, we wish to shorten the nurse's daily working hours. So the objective is formulated as follows:

$$\min \quad \sum_{k \in K} c_{start}^{k} \cdot s^k + \sum_{k \in K, d \in D} \tau^{k,d}, \tag{1}$$

subject to the following constraints:

- bundle constraint, every treatment will be visited exactly once,

$$\sum_{k \in K, d \in D, i \in N} x_{i,j}^{k,d} = 1, \qquad \forall j \in N_{trm}, \tag{2}$$

- flow capacity constraint, the flow cannot exceed the upper bound on each arc,

$$x_{i,j}^{k,d} \leq u_{i,j}^{k,d}, \qquad \forall (i,j) \in A, k \in K, d \in D, \tag{3}$$

- flow conservation constraint, the inflow of each treatment node should equals its outflow,

$$\sum_{j \in N} x_{j,i}^{k,d} = \sum_{j \in N} x_{i,j}^{k,d}, \qquad \forall i \in N_{trm}, k \in K, d \in D, \tag{4}$$

- time window constraint, which the starting time of each treatment should obey,

$$\underline{t}_i \leq t_i \leq \bar{t}_i, \qquad \forall i \in N_{trm}, \tag{5}$$

– time compatibility constraint, i.e., the starting time difference between two consecutive nodes should not be smaller than its necessary amount, for the three types of arcs, namely,
  1. deadhead arcs:

$$t_i + \delta_{i,j} + M \cdot (x_{i,j}^{k,d} - 1) \leq t_j, \qquad \forall (i,j) \in A_{deadhead}, k \in K, d \in D, \tag{6}$$

  with sufficiently large value for M;
  2. pull-out arcs:

$$420 + \delta_{i,j} + M \cdot (x_{i,j}^{k,d} - 1) \leq t_j, \qquad \forall (i,j) \in A_{pullout}, k \in K, d \in D, \tag{7}$$

  where every nurse is supposed to start their daily work at 7 am (420 minutes after 0:00); and
  3. pull-in arcs:

$$t_i + \delta_{i,j} + M \cdot (x_{i,j}^{k,d} - 1) - 420 \leq \tau^{k,d}, \qquad \forall (i,j) \in A_{pullin}, k \in K, d \in D, \tag{8}$$

  where the daily working time $\tau^{k,d}$ of a nurse k on a day d is measured by the difference between the ending time of her work at the depot and the starting time of her work at 7 a.m.;
– the maximum daily working time should not be exceeded,

$$\tau^{k,d} \leq T_{day}^{k,d}, \qquad \forall k \in K, d \in D, \tag{9}$$

– the maximum weekly working time should not be exceeded,

$$\tau^{k} \leq T_{week}^{k}, \qquad \forall k \in K, \tag{10}$$

– the nurse deployment indicator s^k is determined by checking each pull-out arc,

$$s^k \geq \sum_{(i,j) \in A_{pullout}, d \in D} x_{i,j}^{k,d}, \qquad \forall k \in K, \tag{11}$$

– and the necessary day difference between a pair of related treatments is modeled in two steps:
  1. the former treatment i should not start too late so that the latter treatment j can be visited during the week, i.e., i should start latest on the day of $(5 - \sigma_{i,j})$:

$$\sum_{h \in N, k \in K, d \in \{1,\ldots,5-\sigma_{i,j}\}} x_{h,i}^{k,d} = 1, \qquad \forall (i,j) \in R, \tag{12}$$

  2. then the latter treatment j should be started no earlier than the day $d' \geq d + \sigma_{i,j}$,

$$\sum_{h \in N, k \in K} x_{h,i}^{k,d} + \sum_{h' \in N, k' \in K, d' \in \{1,\ldots,d+\sigma_{i,j}-1\}} x_{h',i}^{k',d'} \leq 1, \qquad \forall (i,j) \in R, d \in \{1,\ldots,5-\sigma_{i,j}\}. \tag{13}$$

To sum up, the home health care scheduling problem can be formulated as follows:

$$\begin{aligned} \text{minimize} \quad & (1), \\ \text{subject to} \quad & (2),(3),(4),(5),(6),(7),(8),(9),(10),(11),(12),(13), \\ & x \in \{0,1\}^{|A| \times |K| \times |D|}, t \in \mathbb{Z}_+^{|N_{trm}|}, s \in \{0,1\}^{|K|}, \tau \in \mathbb{Z}_0^{+\,|K| \times |D|}. \end{aligned}$$

5 Primal Heuristics

Our primal heuristics include a tailored greedy construction, different local search operators, and a PGreedy mechanism for automatic adaptation of the greedy parameters.

5.1 Greedy Construction Heuristic

The greedy construction heuristic builds a complete solution from scratch, by iteratively choosing an immediate best solution component, until a complete solution is generated. There are two greedy decisions to be made during the construction: choosing a nurse to start, and choosing a next treatment node to append.

There are two types of nurses. An assistant nurse can handle only a subset of "simple" treatments and is less expensive than professional nurses. If there are only very few (less than 20) simple treatments left, we will always start with the professional nurse. Otherwise, given the proportion of simple treatments as P_s, we select an assistant nurse by the probability $P_{assist} := P_s \cdot \frac{c_{prof}}{c_{assist}}$, where c_{prof} and c_{assist} refer to the cost of a professional and an assistant nurse, respectively. Within the same nurse type, we start with the nurse with the most maximum weekly working time.

After the nurse is chosen, all his or her allowed working days are randomized. Then for each day, we first filter the treatments that are feasible for that day, and then build the schedule by iteratively appending the next best treatment node. For selecting the next treatment, the following factors are considered:

Link time. $\Delta t_j := \min\{\delta_{i,j}, \underline{t}_j - t^{current}\}$, i.e., the earliest possible starting time of the next node j minus the current time;

Time window size. $t_j^{day} := \bar{t}_j - \max\{\underline{t}_j, t^{current} + \delta_{i,j}\}$, i.e., how flexible the treatment can start during the day, since the treatments that are less flexible may be more preferred during the construction, while the more flexible treatment may be treated at a later time;

Time window in other days. t_j^{other}, which is the sum of the time window size of any other days. The same rationale above applies here: if a treatment can be only taken on this day, it should have a higher priority than the ones that are more flexible.

Then the greedy best next node j^* is selected as follows:

$$j^* = \arg\min_{i \in N_{feasible}} \quad \Delta t_j + \alpha \cdot t_j^{day} + \beta \cdot t_j^{other}, \tag{14}$$

where the two greedy weighting parameters α and β are determined dynamically by a PGreedy approach described below.

5.2 Local Search

We have applied two types of local search operators: a *node insertion* and a *nurse type exchange* operator.

Node insertion. Since our primary objective is to reduce the number of nurses, this is also the goal for the node insertion. We first sort all the nurses in the constructed schedule by the number of treatments handled. Then we iteratively try to insert a treatment node from the least-loaded nurse to all the nurses that are more-loaded. The best-improvement

strategy is applied, i.e., if there are multiple positions with multiple nurses where a treatment can be inserted, then the one with the least additional working hours is chosen. Ties are broken randomly. Note that after a node has been inserted, the corresponding time windows of the treatment nodes have to be updated by a constraint propagation. After all the nodes from the least-loaded nurse have been tried, then we resort all the nurses except the least-loaded nurse, and we start from the second least-loaded nurse, and repeat inserting all its nodes to more-loaded nurses as described above. Once a node is inserted to the schedule of another nurse, we check immediately whether the least-loaded nurse can insert its nodes to the nurse with a decremented node. This process ends, when all the nurses but the most-loaded one have been tried to insert their nodes. Note that this node insertion local search does not necessarily reduce the overall objective value, but its goal is mainly to try to reduce the number of nurses. It happens in some cases that the objective value may increase after the node insertion operation, since the total working time increases. However, it is very effective in reducing the number of nurses.

Nurse type exchange. After the node insertion operation, a nurse type exchange is performed. It starts by checking all the treatments of each professional nurse, if all the treatments taken by a professional nurse can be taken by an assistant nurse, then an assistant nurse is deployed instead.

5.3 Online Parameter Adaptation

The best greedy criterion is usually unknown in advance, besides, each instance may have a different best greedy criterion. The weighting parameters in the greedy construction, α and β can be adapted while running the algorithm. This is done by following the procedure described as Parameterized Greedy (PGreedy) [12]. The PGreedy algorithm can find the best greedy criterion while running on a particular instance, by using a black-box optimization algorithm to search the greedy parameter space. One example of such black-box search algorithm proposed in [12], and adopted in our work, is the improving hit-and run (IHR) procedure [24].

The heuristic procedure works as follows. The improving hit-and-run is applied to adapt the best-so-far greedy parameter setting to start each iteration. In each iteration, it runs a greedy construction (Section 5.1) with the adapted parameter setting, followed by a local search (Section 5.2). If a new best-so-far solution is found, we also update the best-so-far greedy parameter setting to the new one as in the IHR procedure. It leaves to define the range of the parameters. In our preliminary experiments, α is set to $[0, 0.2]$, and β is set to $[0, 0.1]$.

6 Experiments and Results

The primal heuristics are implemented in Java. The code is executed on a personal computer with AMD AthlonXP CPU at 1.5 GHz and 256 MB DDR RAM, running Windows XP as operating system. The mathematical model is formulated using the Zimpl language [15], and the Zimpl generated ILP model is then solved by ILOG CPLEX 10. Zimpl and CPLEX ran on a computing server with 32 GB RAM and 8 $\times$ 2.4 GHz AMD Opteron 880 CPU, running SUSE 10.2 Linux OS.

Table 1 The instances used in our case study, extracted from a current real-world home care schedule. The entire instance has 99 patients, 460 treatments, and 9 nurses. Smaller instances were extracted, including from 22 to 285 treatments. Below it lists the size of the problems including the number of patients, treatments, nurses and related pairs, as well as the size of the ILP models built by Zimpl, including the number of variables, constraints and non-zero coeffecients.

Instance	Problem size				ILP model size		
	#patient	#treatment	#nurse	#related-pair	#variable	#constr	#non-zero
`nurse22`	5	22	2	0	5.3K	8.4K	4.3K
`nurse75`	15	75	3	4	87K	99K	124K
`nurse110`	25	110	3	3	185K	218K	202K
`nurse153`	33	153	4	5	474K	542K	583K
`nurse210`	46	210	4	10	891K	962K	1.4M
`nurse285`	60	285	6	9	2.1M	2.1M	3.0M
`nurse460`	99	460	9	17	8.5M	8.9M	69.8M

6.1 The Instances

One data set is available from our application partner, a local home health care service provider in the whole town area. This instance is a real-world weekly nurse schedule currently in use. It contains 99 patients, 460 treatments, and 9 nurses, and is named `nurse460`. The number of pairs with inter-visit day difference is already greatly reduced in our dataset preprocessing phase, and 17 pairs are still left. Each treatment has a patient-specific time window, which is given in advance. The size of the time window varies greatly, ranging from 30 minutes to 360 minutes. The resulting ILP model from Zimpl consists of over 8 millions variables, 9 millions constraints, and 70 millions non-zeros, as listed in the last row of Table 1. CPLEX is not able to even finish one run of the simplex algorithm for the LP relaxation model within our computation time limit, which is set to 6 hours.

In order to better study the problem scalability, we further extract smaller instances from the real-world instance `nurse460`. These instances were extracted by randomly selecting a number of patients, and then including all the treatments and related pairs of the selected patients. The size of the extracted instances ranges from 5 patients with 22 treatments to 60 patients with 285 treatments. The number of nurses has an important influence on the computational difficulty of the ILP model, since each individual nurse determines a specific commodity layer in our multi-commodity network, hence the size of the network model is proportional to the number of nurses. For more details on this issue, we refer to [23]. Therefore, we tried to minimize the number of nurses for the extracted instances by taking the number of nurses of the best found heuristic solution. The size of the ILP model built by Zimpl, including the number of variables, constraints, and non-zero coefficients for each instance is also presented in Table 1.

6.2 Computational Results of the Primal Heuristics

For each of the extracted subinstances, 10 trials of the heuristic are run, and each trial was allowed 30 seconds CPU time. We reported the best solution of the 10 trials in Table 2, and reported its overall computation time as $30 \times 10 = 300$ seconds.

It is worth mentioning that for the real-world instance `nurse460` a good scheduling solution can be found by our heuristic algorithm within 5 minutes. This schedule requires only eight nurses, which improves the original manual nurse schedule currently in use by

Table 2 The computational results of the primal heuristics. Reported is the best solution out of 10 runs of 30 CPU seconds each. The objective value, total number of nurses and number of nurses of the two different types (professional or assistant), total working time and the average working time per nurse are listed.

Instance	Comp. Time	Obj. Value	Nurses (prof. / asst.)	Total #nurse	Total Work Time	Average Work Time
`nurse22`	300	1702200	1 / 1	2	2200	1100
`nurse75`	300	2702071	2 / 1	3	2071	690
`nurse110`	300	2704577	2 / 1	3	4577	1526
`nurse153`	300	3705336	3 / 1	4	5336	1334
`nurse210`	300	3706175	3 / 1	4	6175	1544
`nurse285`	300	5707509	5 / 1	6	7509	1252
`nurse460`	300	7713293	7 / 1	8	13293	1662

Table 3 The computational results of the commercial ILP solver. Two LP relaxation strategies were used: simplex and interior point method. The computation time in seconds, the best integer solution (upper bound), best node (lower bound) and the gap is reported.

Instance	Simplex				Interior Point			
	Comp. Time	Best integer	Best node	Gap	Comp. Time	Best integer	Best node	Gap
`nurse22`	33	1702200	1702200	0%	27	1702200	1702200	0%
`nurse75`	21600	-	1003438	-	21600	2702103	2000098	26%
`nurse110`	21600	-	1000359	-	21600	-	1000359	-
`nurse153`	21600	-	1000000	-	21600	-	1333692	-
`nurse210`	21600	-	700633	-	21600	-	701800	-
`nurse285`	21600	-	0	-	21600	-	1000000	-
`nurse460`	21600	-	0	-	21600	-	0	-

reducing one nurse. It would be also interesting to compare the secondary optimization goal, the total working time, with the manual schedule, but unfortunately, this information is not available from our application partner.

6.3 Computational Results of the ILP Solver

The commercial ILP solver CPLEX is also used to solve the model of the instances presented above. Two different strategies were applied to solve the subsidiary LP relaxation in the branch-and-bound procedure: the default simplex method and the interior point method.

As shown in Table 3, the solver CPLEX has difficulty in solving the ILP model. Only the smallest instance `nurse22` can be solved to optimality. The interior point method appears to be a better alternative for solving the LP relaxation problem by providing better lower bounds compared to the default simplex method. Therefore, the interior point method is used instead of simplex for solving LP relaxation. For instances that are larger than 75 treatments, no feasible integer solution was found. In such case, the branch-and-bound procedure cannot prune any subtree, and it results in an exhaustive search. Therefore, the heuristic approach is essential for solving this problem. Not only can the heuristic provide feasible solutions, it may also be used as an initial upper bound to help the branch-and-bound procedure to prune provable inferior subtrees.

Table 4 The computational results of initializing the ILP solver CPLEX by heuristic solution. The second and third column presents the best heuristic solution and its gap with respect to the lower bound calculated in Table 3; the fourth to seventh column shows the computation time (in seconds), best integer (objective value), best node (lower bound), and the root gap of the ILP solver with initialization of the heuristic solution; the eighth and ninth column shows the number of nurses, and the total working minutes (and the reduced minutes with respect to best heuristic solution).

Instance	Primal heuristic		ILP solver with heuristic initinalization				Final Solution	
	Obj. Value	Gap	Comp. Time	Best integer	Best node	Gap	#Nurse	Work Time (diff.)
`nurse22`	1702200	0%	30	1702200	1702200	0%	2	2200 (0)
`nurse75`	2702071	26%	21600	2702001	2000144	26.0%	3	2001 (3.4%)
`nurse110`	2704577	63%	21600	2704525	1000359	63%	3	4525 (1.1%)
`nurse153`	3705336	64%	21600	3705336	1500359	60%	4	5336 (0)
`nurse210`	3706175	81%	21600	3706175	701800	81%	4	6175 (0)
`nurse285`	5707509	82%	21600	5707509	1000000	82%	6	7509 (0)
`nurse460`	7713293	100%	21600	7713293	0	100%	8	13293 (0)

6.4 Initialize ILP Solver by Heuristic Solution

As indicated in the previous section, although the ILP solver CPLEX is equipped with various heuristic approaches for finding feasible solution, it is unable to find feasible solution within our computation time limit. One possible speed-up is to provide the best heuristic solution as an initial feasible solution for the ILP solver. The advantage of doing so is twofold: on the one side it may help the branch-and-bound procedure to reach more nodes by discarding some provable inferior subtrees; and on the other hand, the ILP solver may help further improve the heuristic solution.

The results are shown in Table 4. Comparing to the lower bound listed Table 3, the lower bound in `nurse75` and `nurse153` is improved. Especially the lower bound improvement of `nurse153` leads to over 4% improvement of the root gap. From the upper bound perspective, although the number of nurses calculated by the heuristic is not further improved, we observe that the total working time in `nurse75` and `nurse110` has been improved by 3.4% and 1.1%, respectively.

7 Conclusions and Future Works

The current work presented a preliminary case study of the home health care scheduling problem for a local health care provider in Germany. At the current stage, it is important to evaluate the scalability of different solution approaches, as well as the potential cost saving for the organization. In this work, we formulated the weekly based home health care scheduling problem as an integer linear programming model, and applied a state-of-the art ILP solver to it. Unfortunately, the current model is only able to be solved optimally up to a tiny instance size. The heuristics on the other hand perform very well, and are able to obtain good solutions within five minutes. Initializing the branch-and-bound process with a heuristic solution seems to speed up the solving process, but the gap for a real-world size instance is still vast. The results also indicate at least 10% cost saving potential for using a computerized optimization process.

As this is still an ongoing work, there are many directions that the current work can be extended. The future directions can be categorized into two aspects, methodology and

application. From the methodological aspect, although this work shows that local search approaches are more preferable for the real-world size problem than the exact approaches, it is still interesting to obtain a provable optimality or a lower bound to assess the local search approaches. A potential direction is to reformulate the current mathematical model into a set partitioning problem and solve it by a branch-and-price approach [1], since the branch-and-price approaches have proven to be successful for solving the VRP with time windows [8]. As for the heuristic approach, it will be interesting to further explore the more effective local search techniques. Our current node insertion operator is mainly aimed at reducing personnel size. It will be interesting to start a second phase of the local search aiming at also reducing the total working time, and rebalancing the workload among different nurses. To this end, local search operators such as node exchange and swapping can be also effective. In fact, it will also be interesting to iterate these local search operators in a variable neighborhood descent fashion [20], so that the local optimum found is the local optimum with respect to all the local search operators. Besides, instead of restarting the heuristic in each iteration from scratch, it will be more effective to perturb a small part from the best-so-far solution, and then perform local search afterwards without a full construction, as done in the iterated local search [18].

From the application aspect, currently we consider mainly minimizing operating cost as the optimization objectives. In fact, there are still other objectives that our application partner cares about other than costs, such as follows.

Workload balancing. Each nurse should have more or less the same amount of workload, such that each nurse has a similar number of patients or treatments to handle. The current schedules generated by the node insertion local search may cause certain nurses with very heavy workload while some other nurses have only very few treatments to handle. This can be done by adding a balancing objectives into the mathematical model, and by also applying node insertion to insert treatments from heavily-loaded nurses to less-loaded nurses.

Dynamic reallocation. The scheduling tool should be robust in the presence of changes. If any changes happen, the reallocation of a new schedule should be made in a short time with as few modifications to the original schedule as possible. There are various possible changes, for instances, new clients joining in, clients changing their preferred visitation dates or time windows, or nurse staff's availability. Some of these changes are known one week before, but in some urgent cases, which are not rare, patients may call in a short time to change the visit appointment to another time, nurses may call in sick shortly before the schedule starts, or a damage to their car suddenly occurs. In such cases, especially the last minute cases, a new schedule should be generated in a short time, with as few changes to the original schedule as possible. In such case, a local search procedure is important.

Consistent nurse. One important feature to improve the service quality is to always assign the same nurse to the same patient's treatments. From the nurse's perspective, it is easier to catch up with the patient's situation, while from the client's perspective, they also desire more consistency. Keeping the nurse consistency may reduce scheduling flexibility, and may result in higher operating cost or longer working time. Technically, each node insertion will need to consider inserting a set of treatments of the same patient instead of inserting one treatment node. To this end, Kovacs et al. [17] also provides a nice survey of vehicle routing problems with consistency considerations.

Acknowledgements This work is supported by BMBF Verbundprojekt E-Motion.

References

1. C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh, and P. H. Vance. Branch-and-price: Column generation for solving huge integer programs. *Operations research*, 46(3):316–329, 1998.
2. S. V. Begur, D. M. Miller, and J. R. Weaver. An integrated spatial DSS for scheduling and routing home-health-care nurses. *Interfaces*, 27(4):35–48, 1997.
3. S. Bertels and T. Fahle. A hybrid setup for a hybrid scenario: combining heuristics for the home health care problem. *Computers & Operations Research*, 33(10):2866–2890, 2006.
4. E. K. Burke, P. De Causmaecker, G. Vanden Berghe, and H. Van Landeghem. The state of the art of nurse rostering. *Journal of scheduling*, 7(6):441–499, 2004.
5. J. A. Castillo-Salazar, D. Landa-Silva, and R. Qu. Workforce scheduling and routing problems: literature survey and computational study. *Annals of Operations Research*, pages 1–29, 2014.
6. B. Cheang, H. Li, A. Lim, and B. Rodrigues. Nurse rostering problems – a bibliographic survey. *European Journal of Operational Research*, 151(3):447–460, 2003.
7. E. Cheng and J. L. Rich. A home health care routing and scheduling problem. Technical report, Technical Report TR98-04, Department of CAAM, Rice University, 1998.
8. J. F. Cordeau, G. Desaulniers, J. Desrosiers, M. M. Solomon, and F. Soumis. VRP with Time Windows. In P. Toth and D. Vigo, editors, *The vehicle routing problem*, pages 157–193. Society for Industrial and Applied Mathematics, 2001.
9. L. Di Gaspero and T. Urli. A CP/LNS approach for multi-day homecare scheduling problems. In M. J. Blesa et al., editors, *Hybrid Metaheuristics*, volume 8457 of *Lecture Notes in Computer Science*, pages 1–15. Springer International Publishing, 2014.
10. A. T. Ernst, H. Jiang, M. Krishnamoorthy, and D. Sier. Staff scheduling and rostering: A review of applications, methods and models. *European journal of operational research*, 153(1):3–27, 2004.
11. P. Eveborn, P. Flisberg, and M. Rönnqvist. LAPS CARE – an operational system for staff planning of home care. *European Journal of Operational Research*, 171(3):962–976, 2006.
12. A. Fügenschuh. Parametrized greedy heuristics in theory and practice. In *Proceedings of Hybrid Metaheuristics*, volume 3636 of *Lecture Notes in Computer Science*, pages 21–31. Springer, 2005.
13. A. Fügenschuh. The vehicle routing problem with coupled time windows. *Central European Journal of Operations Research*, 14(2):157–176, 2006.
14. Y. Kergosien, C. Lenté, and J.C. Billaut. Home health care problem: An extended multiple traveling salesman problem. In *Proceedings of 4th Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA)*, 2009. 8 pages.
15. T. Koch. *Rapid Mathematical Prototyping*. PhD thesis, Technische Universität Berlin, 2004.
16. P. M. Koeleman, S. Bhulai, and M. Van Meersbergen. Optimal patient and personnel scheduling policies for care-at-home service facilities. *European Journal of Operational Research*, 219(3):557–563, 2012.
17. A. A. Kovacs, B. L. Golden, R. F. Hartl, and S. N. Parragh. Vehicle routing problems in which consistency considerations are important: A survey. *Networks*, 64(3):192–213, 2014.
18. H. Lourenço, O. Martin, and T. Stützle. Iterated local search. *Handbook of metaheuristics*, pages 320–353, 2003.
19. C. E. Miller, A. W. Tucker, and R. A. Zemlin. Integer programming formulation of traveling salesman problems. *Journal of the ACM*, 7(4):326–329, 1960.
20. N. Mladenović and P. Hansen. Variable neighborhood search. *Computers & Operations Research*, 24(11):1097–1100, 1997.
21. S. Nickel, M. Schröder, and J. Steeg. Mid-term and short-term planning support for home health care services. *European Journal of Operational Research*, 219(3):574–587, 2012.
22. M. S. Rasmussen, T. Justesen, A. Dohn, and J. Larsen. The home care crew scheduling problem: Preference-based visit clustering and temporal dependencies. *European Journal of Operational Research*, 219(3):598–610, 2012.
23. Z. Yuan. Solving Real-World Vehicle Routing Problems Using MILP and PGreedy Heuristics, Diplomarbeit, Technische Universität Darmstadt, 2007.
24. Z. B. Zabinsky, R. L. Smith, J. F. McDonald, H. E. Romeijn, and D. E. Kaufman. Improving hit-and-run for global optimization. *Journal of Global Optimization*, 3(2):171–192, 1993.