Task scheduling in long-term care facilities: A client-centered approach

Alexander Lieder[*◇], Dennis Moeke[⋆†], Ger Koole[†], Raik Stolletz[*]

[*]Chair of Production Management, University of Mannheim, Germany

[⋆]School of Healthcare, Windesheim University of Applied Sciences, The Netherlands

[†]Department of Mathematics, VU University Amsterdam, The Netherlands

[◇]Corresponding author, lieder@uni-mannheim.de
Phone: +49-621-181-1578   Fax: +49-621-181-1579

Abstract

In this study, we analyze a task scheduling problem with small time windows and care workers with different levels of qualification in a nursing home. A set of care tasks has to be assigned to a given number of care workers, so that the total earliness and tardiness from the nursing home residents' preferred time is minimized.

To optimally solve this scheduling problem, we formulate a mixed integer program (MIP) and develop a dynamic programming (DP) approach. The numerical analysis shows the reliability of this optimization approach as well as of a heuristic DP approach. A sensitivity analysis with real-world demand data shows the impact of hierarchical qualification levels and large-scale scheduling.

Key words: residential care, task scheduling, dynamic programming

1 Introduction

The share of public spending on health care has become a central issue in most western countries. An important reason for this is that the number of elderly citizens in need of long-term care is expected to rise significantly over the next decades (Geerts et al., 2012; Colombo et al., 2011). As nursing homes play an important role in the provision of long-term elderly care (Spilsbury et al., 2011), they face the challenge of providing reliable, high-quality care while at the same time lowering their costs. According to Moeke and Verkooijen (2013), most nursing home facilities struggle to give substance to these two conflictive goals.

When it comes to nursing home care, the concept of *quality of care* is difficult to capture (Spilsbury et al., 2011; Courtney et al., 2009). As stated by Moeke and Verkooijen (2013) and Moeke et al. (2014), the goal of a client-centered nursing homes is to provide their residents with the care they require in such way that it becomes possible for them to live the life they prefer. The optimization problem that arises in this context is to design schedules for the delivery of health care tasks that meet the time preferences of the clients as closely as possible.

We distinguish between two types of health care tasks: (1) Tasks for which it is possible to make a fairly detailed planning in advance, based on the needs and preferences of the client, *e.g.,* giving medicine and help with getting out of bed in the morning (*care by appointment*), (2) tasks in response to stochastic demand, *e.g.,* assistance with toileting (*care on demand*, see van Eeden et al., 2014). In practice, both problems are usually treated separately with dedicated care workers.

In this study we focus on *care by appointment*. We model the clients' time preferences for each task under consideration by a preferred start time which is surrounded by a penalty function for earliness and tardiness and by a narrow time window. Each task has a known duration and requires a certain qualification level (QL) of the respective care worker. The QLs are hierarchical, *i.e.,* a care worker with a higher QL is able to perform tasks with a lower qualification requirement as well. This is also referred to as *skill substitution* (De Bruecker et al., 2015). An overview of the QLs and examples for the respective tasks, as can be found in practice, are given in Table 1.

We aim to assign each task to a qualified care worker and to a start time within the time window and as close as possible to the preferred start time. Our objective is to minimize the total earliness/tardiness penalties of all task assignments from the

1

QL	Typical tasks
QL 1	Bringing food and drinks, cleaning, transferring, bed cleaning
QL 2	Getting in and out of bed, eating, toileting, making the beds, washing
QL 3	Giving medication, simple medical check ups

Table 1: Overview of care worker qualification levels

preferred start time. For this decision, the shifts for the care workers are already defined.

To model this scheduling problem, we present a mixed integer programming (MIP) formulation. As standard solvers are not able to solve real-world instances to optimality, we develop a dynamic programming (DP) approach. We use real-world demand data to show that the obtained task schedules significantly increase the service quality (*i.e.,* the total earliness and tardiness in the respective schedule) compared to First Come First Serve (FCFS) scheduling heuristics, which closely resemble the current way of working.

The contribution of this study can be summarized as follows:

- We define a practice-driven task scheduling problem with time windows and hierarchical QLs,

- we propose a MIP formulation, an exact dynamic programming approach, and an heuristic dynamic programming approach, and

- we test our approach using real-world demand data and perform a sensitivity analysis.

The remainder of this paper is structured as follows. In the following section, we provide an overview of related literature. In Sections 3 and 4 we present a formal description of the underlying optimization problem and develop a DP solution approach. Next, in Section 5 we examine our approach numerically. Section 6 concludes this paper and points out possible directions for further research.

2 Related literature

2.1 Research in nursing homes

Although nursing home residents are often highly dependent on assistance with basic activities in daily living, they still have little or no influence on the actual delivery of health care (Simmons et al., 2011; Persson and Wasterfors, 2009). Because very little research has been undertaken on task scheduling in nursing homes, scheduling in practice is still done manually without explicitly taking into account the time preferences of the residents.

In the Netherlands, most nursing homes consist of several independent care providing departments (*Decision Making Units* or DMUs), each of which are responsible for their own workforce planning. Often, a DMU is further divided into several clusters and each cluster schedules its care tasks separately, *i.e.,* they perform *small-scale scheduling*. However, by *pooling* the demand of clusters or departments it becomes more likely that high demand from one resident will be balanced out by low demand from another (Eppen, 1979; Wallace and Whitt, 2004). The studies by Vanberkel et al. (2012) and Wright and Mahar (2013) examines pooling effects on the nursing staff in a hospital setting. The study of Moeke et al. (2014) examines scale efficiencies in a nursing home context.

Furthermore, most nursing homes make use of so-called *differentiated practice*, *i.e.,* health care tasks are assigned *only* to care workers with that specific QL (Jansen et al., 1997). Studies on *skill substitution* (also referred to as *skill-mix* or *hierarchical qualifications*) are provided by Gibbs et al. (1991) and Spilsbury et al. (2011) in a hospital setting, and by Moeke et al. (2014) in a nursing home setting. Different from this paper, these studies do not provide an optimization approach for the task scheduling problem at hand and they focus on the impact of skill substitution on staffing costs. The paper at hand focuses on improvements of service quality.

2.2 Research on task scheduling problems

Task scheduling can be considered a subproblem of workforce planning, see the framework for workforce planning in Ernst et al. (2004). They separate the task scheduling problem from the superordinate planning problems by combining "in-

dividual tasks into task sequences that could usefully be carried out by one person" to derive an aggregated demand. There are dedicated literature reviews on workforce planning in health care (Burke et al., 2004; Cheang et al., 2003), but they focus solely on staffing decisions. As they consider an aggregated demand for workers rather than individual tasks, they do not consider detailed scheduling with respect to the clients' time preferences.

De Bruecker et al. (2015) provide a broad overview of papers on workforce planning, including both, planning of individual tasks and planning with hierarchical skills, but none of the references therein considers the same combination of assumptions as the problem at hand. Baker and Scudder (1990) review scheduling problems with homogeneous resources, earliness/tardiness penalties, and a preferred starting time but without a hard time window. Task scheduling problems with multiple resources which take either different qualifications (also referred to as skills) or time preferences into account can be found in the literature for different areas of application:

Task scheduling with different qualifications

Bellenguez-Morineau and Néron (2007) present a project scheduling problem that assumes a given workforce with hierarchical skill levels. As they do not consider time preferences for individual tasks, the objective is to minimize the makespan. Krishnamoorthy et al. (2012) present a task scheduling problem with hierarchical skill levels. They assume fixed start times for all tasks as a hard constraint. Their objective is to find the minimum required workforce to obtain a feasible schedule. This paper, on the contrary, assumes a given workforce composition and variable start times for tasks.

Schimmelpfeng et al. (2012) present a task scheduling approach for rehabilitation hospitals with different qualifications and precedence constraints between tasks, but they do not consider time preferences for individual tasks. A general difference between a hospital and a nursing home is that patients usually visit hospitals only for a short period of time, therefore hospitals can focus on high resource utilization rather than on meeting time preferences.

Task scheduling with time windows

Gertsbakh and Stern (1978) discuss task scheduling with time windows for a homogeneous workforce. They do not consider earliness/tardiness penalties. The objective is to find the minimum required workforce to obtain a feasible schedule.

Mankowska et al. (2014) discuss the Home Health Care Routing and Scheduling Problem as a Vehicle Routing Problem with Time Windows. Clients are visited in their homes, therefore sequence-dependent travel times are taken into account. They consider time preferences and different skills. However, they assume half-open time windows that do not allow earliness while tardiness is penalized, but not limited. The objective is to minimize the weighted sum of total travel times, total tardiness, and maximum tardiness.

There are other Vehicle Routing applications with time windows, but at least one of the characteristics of the assignment problem at hand is missing. For example, many papers discuss the Aircraft Landing Problem (ALP), e.g., Beasley et al. (2000) and Lieder et al. (2015). The predominant objective is to minimize the total earliness/tardiness penalties, but the resources (runways) are assumed to be identical.

Appointment scheduling

The problem discussed in this paper has similarities to the appointment scheduling problem, that also aims to minimize the waiting time of clients for a task to be performed. A review of appointment scheduling literature is given by Cayirli and Veral (2003). Cayirli et al. (2006) evaluate different *appointment rules* for two classes of clients in a simulation study, Kaandorp and Koole (2007) derive appointment schedules with a local search approach, Begen and Queyranne (2011) calculate optimal start times for a fixed sequence of tasks on a discrete time axis, and Erdogan and Denton (2013) consider a dynamic appointment scheduling problem, where demand occurs throughout the workday. The main difference to the task scheduling problem discussed in this paper is that, in appointment scheduling, the clients cannot choose their preferred start times freely. The are assigned to a start time or can choose a start time out of a limited number. Furthermore, a single service provider, stochastic task durations, no-shows, and unpunctual clients are assumed.

3 Problem description and model formulation

In the task scheduling problem at hand, a given set of care tasks has to be assigned to a set of care workers. For each care task, the respective client has announced a preferred start time with a feasible time window around. Furthermore, each task has a known duration and requires a certain qualification level of the respective

care worker. Each care worker has a qualification level. These qualification levels are assumed to be hierarchical, i.e., care workers with a given level are allowed to perform tasks that require either the respective level or a lower level.

The scheduling decision to be made is to assign each task to a care worker and to a start time. To obtain a feasible task schedule, it has to be ensured, that each task is assigned to a care worker with (at least) the required qualification level and to a start time within the time window around the preferred start time. A care worker can only perform one task at a time and preemption of tasks is not allowed.

For each task that is assigned to an earlier or later start time than the preferred start time, an earliness/tardiness penalty is incurred according to a convex penalty function. Our objective is to derive a feasible task schedule that minimizes the total penalties of all task assignments and thereby focuses the clients' time preferences.

This problem can be formulated as a mixed-integer programming (MIP) model. The following model is based on Beasley et al. (2000) but considers hierarchical QLs. The sets, parameters, and decision variables are summarized in Table 2.

Sets:

J	tasks
R	care workers
$V \subset J \times J$	pairs of tasks with pre-determined order

Parameters:

p_j	preferred start time of task j
e_j	earliest start time of task j
l_j	latest start time of task j
d_j	duration of task j
c	cost coefficient
q_j^{min}	minimum qualification level required to perform task j
q_r	qualification level of care worker r
M	sufficiently large number

Variables:

x_j scheduled start time of job j

$$\delta_{j,j'} = \begin{cases} 1, & \text{if task } j \text{ is performed before job } j' \\ 0, & \text{otherwise.} \end{cases}$$

$$z_{j,j'} = \begin{cases} 1, & \text{if tasks } j \text{ and } j' \text{ are performed by the same worker} \\ 0, & \text{otherwise.} \end{cases}$$

$$y_{j,r} = \begin{cases} 1, & \text{if task } j \text{ is performed by worker } r \\ 0, & \text{otherwise.} \end{cases}$$

Table 2: Sets, parameters, and variables of the MIP

$$\text{Min. } Z = \sum_{j \in J} f(x_j - p_j) \tag{1}$$

subject to

$$e_j \le x_j \le l_j \qquad\qquad \forall\, j \in J \tag{2}$$

$$\delta_{jj'} + \delta_{j'j} = 1 \qquad\qquad \forall\, j \ne j' \in (J \times J) \tag{3}$$

$$\delta_{jj'} = 1 \qquad\qquad \forall\, (j, j') \in V \tag{4}$$

$$x_j + d_j \cdot z_{jj'} \le x_{j'} + M \cdot \delta_{j'j} \qquad\qquad \forall\, j \ne j' \in (J \times J) \tag{5}$$

$$\sum_{r \in R} y_{jr} = 1 \qquad\qquad \forall\, j \in J \tag{6}$$

$$z_{jj'} = z_{j'j} \qquad\qquad \forall\, j \in J \quad \forall\, j' \in J \tag{7}$$

$$z_{jj'} \ge y_{jr} + y_{j'r} - 1 \qquad\qquad \forall\, j \in J \quad \forall\, j' \in J \quad \forall\, r \in R \tag{8}$$

$$(q_r - q_j^{min}) \cdot y_{jr} \ge 0 \qquad\qquad \forall\, j \in J \quad \forall\, r \in R \tag{9}$$

$$a_j \ge 0 \qquad\qquad \forall\, j \in J \tag{10}$$

$$\delta_{jj'},\, z_{jj'},\, y_{jr} \in \{0,1\} \qquad\qquad \forall\, j \in J \quad \forall\, j' \in J \quad \forall\, r \in R \tag{11}$$

The Objective (Eq. 1) is to minimize the total earliness/tardiness penalties of all task assignments. The general cost function, $f(x_j - p_j)$, calculates the earliness/tardiness penalty for each task, j, depending on the difference between scheduled start time, x_j, and preferred start time, p_j. For example, it can be defined as

$$f(x_j - p_j) = c \cdot |x_j - p_j|, \tag{12}$$

where c is a given cost coefficient. Eq. (2) ensures that all tasks start within their respective time windows. Eq. (3) defines the order in which the jobs are performed.

We can divide tasks into *task types*, depending on their duration and required QL. It can be shown that, for problem instances that have a feasible solution, there is always an optimal solution in which all tasks of the same type are assigned in FCFS discipline (Briskorn and Stolletz, 2014). Hence, we can define a set of all

pairs of tasks with a pre-determined order of start times, either because they have the same duration and QL or because their order is implied by their time windows:

$$V = \{j, j' \in J \times J | (d(j) = d(j') \wedge (q_j^{min} = q_{j'}^{min} \wedge p_j < p_{j'}) \vee (l_j < e_{j'} + d_{j'})\} \quad (13)$$

This partial order is fixed by Eq. (4). Eq. (5) makes sure that two tasks, performed by the same care worker, cannot overlap. Eq. (6) assures that each task is performed by exactly one worker. Eq. (7) assures consistency of the variables $z_{jj'}$ and $z_{j'j}$, that indicate whether a pair of tasks is performed by the same care worker. Eq. (8) relates the y_{jr}-variables, that indicate which task is performed by which care worker, to the $z_{jj'}$-variables. The binary $z_{jj'}$ is ensured to be 1 of both tasks are done be the same worker. Eq. (9) assures that tasks are performed by a care worker with at least the requested QL. The remaining equations (Eq. 10 and 11) define the domains of the variables.

4 Dynamic Programming Approach

In the following, we define the states and transitions of the DP. Afterwards, we discuss how the actual number of states could be significantly reduced. To cope with large problem instances, we also present heuristic solution approaches that build upon this DP approach and on FCFS rules.

Lieder et al. (2015) developed an optimization algorithm for the Aircraft Landing Problem with homogeneous recourses which we adapt to model the task scheduling problem with hierarchical skills as a DP. An additional difference is that we allow earliness, *i.e.,* the assignment of start times with $x_j < p_j$ is allowed.

4.1 States and transitions

We divide the set of tasks to be scheduled into task types $a \in A$, depending on their duration and required qualification level. We denote the task type of task j by $a(j)$ and the set of tasks of type a by J_a. The planning horizon, T, is divided into small time intervals, $t \in T$, of the same length, such that the task durations are multiples of the interval length, and the preferred start times are at the beginning of an interval.

Each **state**, s, of the dynamic program is defined as a tuple

$$(n_1, ..., n_a, ..., n_{|A|}, t_1, ..., t_r, ..., t_{|R|}), \tag{14}$$

where

- $n_a \in \{0, ..., |J_a|\}$ is the number of tasks of type $a \in A$, that are already assigned and

- $t_r \in T$ is the point in time, when care worker $r \in R$ is available for the next task assignment.

The tuple $(t_1, ..., t_{|R|})$ is referred to as a *resource occupation profile (ROP)*. These profiles were introduced by Baptiste (2000), who proposed a polynomial algorithm for scheduling identical tasks on parallel resources. Note that the information on *how many* tasks of a particular type have already been assigned implies *which* of the respective tasks are assigned, because we can assume FCFS for tasks of the same type. The initial state, s_0, of the program is $(0^{|A|}, t_0^{|R|})$, *i.e.,* no tasks are assigned yet and all workers are available at the beginning of the shift under consideration, namely t_0.

A **state transition**

$$\underbrace{(n_1, ..., n_{|A|}, t_1, ..., t_{|R|})}_{s} \xrightarrow[(j,r,t)]{} \underbrace{(n'_1, ..., n'_{|A|}, t'_1, ..., t'_{|R|})}_{s'} \tag{15}$$

corresponds to adding an assignment, (j, r, t), to the partial schedule represented by state s. That is, we have

- $n'_{a'} = n_{a'}$ for each $a' \neq a(j)$,

- $n'_a = n_a + 1$ for $a = a(j)$ and with $n'_a \leq |J_a|$,

- $t'_{r'} = \max\{t, t'_{r'}\}$ for each $r' \neq r$, and

- $t'_r = t + d_j$ with $t_r \leq t$, $e_j \leq t \leq l_j$ and $q_r \geq q_j^{min}$.

In essence, transitions exist for all assignments, (j, r, t), where j is the earliest unassigned task of type a, care worker r is available and qualified, and t lies

10

within task j's time window. By forcing the ROP $(t_1, ..., t_{|R|})$ to always be not smaller than t, we enforce a forward-directed scheduling.

The transition is associated with the penalty $c(s_j - t)$ for the additional assignment. The total penalties of state s can be calculated via a **Bellman recursion** as

$$Z(s) = \min_{\Pi(s,j,r,t)} (Z(s') + c(s_j - t)) \tag{16}$$

where $\Pi(s, j, r, t)$ is the set of states which can be transformed into state s by adding assignment (j, r, t), and $Z(s_0) = 0$. We can derive the schedule corresponding to a state, s, by tracking all transitions (and the corresponding task assignments) from the initial state, s_0, to s.

Let Π_{term} be the set of *terminal states* with $s = (|J_1|, ..., |J_{|A|}|, t_1, ..., t_{|R|})$ for each $s \in \Pi_{term}$, *i.e.,* all states with all tasks being assigned and with no outgoing transitions. To solve the task scheduling problem to optimality, (*i.e.,* to derive a complete feasible schedule with minimal total penalties), we need to find a state, s^*, out of Π_{term} with minimal $Z(s^*)$.

4.2 State space reduction

In the following, we present four methods to reduce the problem's number of states and transitions in order to improve the computation time of the DP approach.

(1) Avoid creating sub-optimal states:
To ensure optimality, we have to assign a particular task, j, to all possible start times within its time window, T_j. However, there are two exceptions:

(a) It cannot be optimal to assign j to a start time later than the preferred start time, $(x_j > p_j)$, unless the respective worker, r, is busy up to x_j. Otherwise, penalty could be reduced and the respective *ROP* could be improved by scheduling j earlier.

(b) If task j' is the direct predecessor of task j, *i.e.,* both tasks are assigned to the same worker with no task scheduled in between, and j' is scheduled early (*i.e.,* $x_{j'} < p_{j'}$), then task j has to be assigned immediately afterwards $(x_j = x_{j'} + d_{j'})$. Otherwise, j' could have been scheduled later, thereby reducing penalty without changing the *ROP*.

11

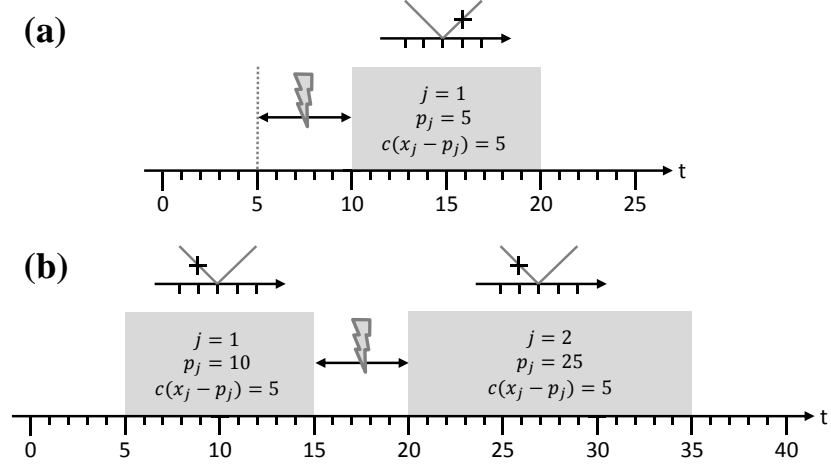Figure 1 shows examples of these sub-optimal assignments.

Figure 1: Examples of sub-optimal task assignments

(2) Detect and remove sub-optimal states:
We know that a state, s, cannot be transformed into the only minimum-cost final state, if there exists another state, s', that fulfills the following criteria:

- At least the same number of tasks of each task type has been assigned $(n_a(s') \geq n_a(s) \ \forall a \in A)$,

- all workers are available at the same or earlier $(t_r(s') \leq t_r(s) \ \forall r \in R)$, and

- the total penalties so far are not higher $(Z(s') \leq Z(s))$.

In this case, s' can be transformed into a final state with the same total penalties as s, or less. Therefore, s can be removed from the state space without losing optimality.

(3) Detect symmetry:
Whenever there are two or more care workers with the same QL, there are multiple states that represent the same schedule, only with a different assignment of tasks to care workers. To eliminate this redundancy, we sort the availability times of each *ROP* in non-descending order within each qualification level. This way, the pairwise comparison described in (2) also detects and removes symmetric states.

(4) Detect infeasibility:

For each state, before developing its successors, we check whether it is possible to assign the first unassigned task of each task type to a qualified worker within its respective time window. If that is not the case for at least one task, it is not possible to transform the current state into a feasible solution. Thus the state can be removed from the state space.

4.3 Heuristic solution approaches

Truncated DP approach

The state space of the presented DP approach has a high dimensionality that can result in a very large number of states on each stage, despite the described checks for sub-optimality, feasibility, symmetry, and dominance. Thus, depending on the size and structure of the problem instance, the computation time of the approach may become prohibitively long. We can reduce the number of states on each stage, and therefore reduce the computational effort, by pruning all but a limited number of states, s^{max}, per stage. To decide, which states to keep and which to prune, we store all states, s, on the same stage in non-descending order of their respective total penalties so far, $Z(s)$, and we prune all but the s^{max} states with the lowest total penalties.

FCFS scheduling rules

We will compare the results of both DP-approaches with two simple FCFS scheduling rules. The list of tasks is assigned in non-decreasing order of preferred start times p_j. For each task, we check if a qualified care worker is available at the task's preferred start time in FCFS rule (a). In this case, the task is assigned to the respective worker and the preferred start time. Otherwise, the qualified care worker who becomes available next is assigned immediately to this task. If there is no worker available with the required (or a higher) qualification between p_j and l_j, the rule results in an infeasible solution.

As rule (a) does not use early assignments, it reduces the size of the tasks' feasible time windows. In FCFS rule (b), the earliest time of the task e_j is used instead of the preferred start time p_j. To reduce earliness penalties in a feasible solution, we go through the schedule for each care worker backwards. The start time of a

task with earliness is shifted towards the preferred start as long as the task does not overlap with the successive task.

5 Numerical study

5.1 Study design

In this numerical study, we first compare the performance of MIP, exact DP, truncated DP, and the FCFS rules presented in this article (Section 5.2). Afterwards, we examine the benefits of *skill substitution* over an inflexible worker assignment, as performed in practice (Section 5.3) and the benefits of *large-scale scheduling* (Section 5.4).

For the purpose of this study, we use real demand data from two separate DMUs of a Dutch nursing home. Each DMU consists of two clusters with separate rooms for each client and supplied demand data from six work days. We do not have the actual workforce compositions and task schedules at our disposal, hence we chose workforce compositions that are in line with staffing policies applied in the Netherlands (Hingstman et al., 2012). The optimized task schedules are compared to an FCFS scheduling policy. This approximates the current policy how tasks are performed in nursing homes.

We assume a hard time window of ± 15 minutes around the preferred start time of each task and a penalty of $c = 1$ MU per minute of earliness/tardiness ($f(x_j - p_j) = |x_j - p_j|$; see Figure 2). All task durations and preferred start times are given in 5-minute intervals.
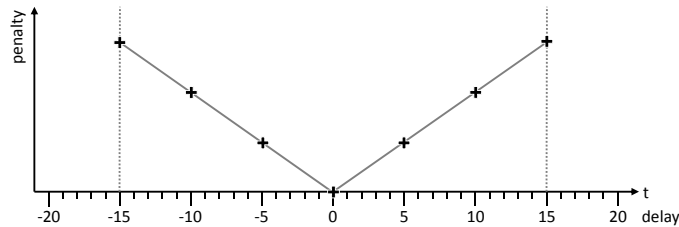

Figure 2: Convex, piecewise linear penalty function

We analyze a morning shift (from 7:00 am to 11:00 am), as a significant part of the daily care is provided during this period of the day (Moeke et al., 2014;

14

Sloane et al., 2007). The size of the data sets (in terms of number of tasks and task types) and the respective numbers of present care workers are shown in Table 3. All computations were performed on an Intel i5 CPU with 2.5 GHz and 8 GB RAM. The MIP was implemented in GAMS and solved by CPLEX 12.2, the DP approach was implemented in Java JDK 1.7. The time limit for all calculations was set to three hours.

Cluster	Day	DMU 1				DMU 2			
		No. of tasks		No. of	Workforce	No. of tasks		No. of	Workforce
		Q1, Q2, Q3	total	task types	(Q1,Q2,Q3)	(Q1,Q2,Q3)	total	task types	(Q1,Q2,Q3)
1	1	1, 19, 13	33	10	1, 2, 2	11, 13, 15	39	12	1, 2, 2
	2	1, 17, 12	30	9	1, 2, 2	8, 12, 14	34	12	1, 2, 2
	3	1, 17, 12	30	9	1, 2, 2	8, 10, 14	32	11	1, 2, 2
	4	4, 20, 13	37	12	1, 2, 2	11, 15, 16	42	13	1, 2, 2
	5	2, 17, 12	31	10	1, 2, 2	13, 12, 16	41	12	1, 2, 2
	6	2, 17, 12	31	10	1, 2, 2	9, 12, 15	36	12	1, 2, 2
2	1	4, 16, 8	28	12	1, 2, 1	6, 6, 11	23	6	1, 1, 1
	2	4, 15, 8	27	12	1, 2, 1	4, 8, 8	20	10	1, 2, 1
	3	4, 15, 9	28	12	1, 3, 1	3, 7, 8	18	7	1, 1, 1
	4	6, 15, 9	30	13	1, 2, 1	9, 9, 9	27	10	1, 1, 1
	5	7, 15, 8	30	12	1, 3, 1	4, 9, 10	22	7	1, 2, 1
	6	6, 17, 8	31	13	1, 2, 1	7, 10, 8	25	10	1, 1, 1

Table 3: Properties of the problem instances

5.2 Comparison of solution approaches

Table 4 shows the total penalties and the computation times for the MIP, the exact DP approach, the truncated DP with $s^{max} = 5,000$, and for the FCFS heuristics.

The MIP hit the time limit for 12 instances. The exact DP approach hit the time limit for 3 instances. The truncated DP approach solved the problem to optimality in 23 problem instances. In one case, the total penalties are 5 *MU* higher than in the optimal solution. Its computation time was below 15 minutes for all instances.

The schedules derived by FCFS rule *(a)* resulted in infeasible solutions for 14 problem instances. For one instance we obtained an optimal schedule, the remaining instances have additional penalties of up to 40 *MU*. Rule *(b)* returns optimal schedules for two instances and non-optimal schedules with additional penalties up to 70 *MU* for 16 instances. For the remaining six instances, no feasi-

15

ble solution was found. Thus, FCFS does not seem to be an appropriate solution method.

In conclusion, the truncated DP shows near optimal solutions in very short computation times. Therefore, we use this approach in the following studies.

| DMU | Cluster | Day | MIP | | | Exact DP | | Truncated DP | | FCFS | |
			Total penalty	CPU time (s)	Gap to LB	Total penalty	CPU time (s)	Total penalty	CPU time (s)	Total penalty Rule (a)	Rule (b)
1	1	1	55	>3h	40%	55	116	55	90	95	60
		2	45	>3h	17%	45	9	45	9	65	50
		3	45	>3h	16%	45	10	45	9	65	50
		4	75	>3h	43%	75	142	75	96	infeas.	95
		5	45	>3h	22%	45	16	45	9	65	50
		6	45	>3h	22%	45	16	45	9	65	50
1	2	1	75	1.899	0%	75	4	75	2	infeas.	infeas.
		2	70	133	0%	70	3	70	2	infeas.	infeas.
		3	70	41	0%	70	142	70	99	infeas.	infeas.
		4	105	7.952	0%	105	3	105	2	infeas.	infeas.
		5	70	15	0%	70	33	70	17	infeas.	infeas.
		6	70	416	0%	70	5	70	3	infeas.	infeas.
2	1	1	60	>3h	33%	—	>3h	60	795	infeas.	120
		2	55	>3h	54%	55	10.404	55	686	infeas.	95
		3	55	>3h	18%	55	5.042	55	389	infeas.	85
		4	80	>3h	63%	—	>3h	85	624	infeas.	155
		5	55	>3h	37%	—	>3h	55	782	infeas.	105
		6	70	>3h	63%	70	2.645	70	405	infeas.	110
2	2	1	40	2	0%	40	<1	40	<1	infeas.	45
		2	10	2	0%	10	<1	10	<1	10	10
		3	15	1	0%	15	<1	15	<1	30	25
		4	20	2	0%	20	<1	20	<1	25	20
		5	20	2	0%	20	1	20	1	25	25
		6	30	2	0%	30	1	30	1	50	40

Table 4: Comparison of solution approaches

5.3 Impact of skill substitution

Without skill substitution, each problem instance can be decomposed into separate problems for each qualification level. For the decomposed problem, the total penalties of all problem instances are shown in Table 5. The used truncated DP

approach with $s^{max} = 5,000$ solved all instances to optimality in less than a second. The reduced flexibility results in a significant increase in total penalties (up to 45 additional *MU*) compared to the solutions with *skill substitution*.

Cluster	Day	DMU 1: Total penalty					DMU 2: Total penalty				
		without skill substitution				with skill	without skill substitution				with skill
		QL1	QL2	QL 3	total	substitution	QL1	QL2	QL 3	total	substitution
1	1	0	45	25	70	55	10	40	40	90	60
	2	0	40	20	60	45	10	45	35	90	55
	3	0	40	20	60	45	10	20	35	65	55
	4	0	85	20	105	75	10	70	35	115	80
	5	0	40	20	60	45	10	60	30	100	55
	6	0	40	20	60	45	5	45	35	85	70
2	1	20	65	20	105	75	5	5	40	50	40
	2	20	60	20	100	70	0	0	10	10	10
	3	20	35	35	90	70	0	30	10	40	15
	4	20	60	40	120	105	0	30	15	45	20
	5	30	55	20	105	70	0	10	15	25	20
	6	30	65	20	115	70	0	40	15	55	30

Table 5: Impact of skill substitution

5.4 Economies of scale

In the following we show the possible benefits of *large scale scheduling, i.e.,* scheduling two neighboring clusters of the same DMU together instead of scheduling each cluster separately. We pooled the tasks and the workforce of both clusters of the respective DMU and work day and solved the resulting 12 problem instances with the truncated DP approach with $s^{max} = 5,000$. We assume that travel times between the clusters are negligible, because the clusters are compact and located close to each other. The results in terms of total penalties and computation times are shown in Table 6.

We observe a *pooling effect* due to the increase in flexibility of the assignments. Compared to the separate schedules of the respective clusters (Table 4). The computation time increased considerably, but the total penalties could be reduced by up to 40 *MU*.

Day	DMU 1					DMU 2				
	Penalty (clusters sched. sep.)			Common schedule		Penalty (clusters sched. sep.)			Common schedule	
	Cluster 1	Cluster 2	Total	Penalty	CPU time (s)	Cluster 1	Cluster 2	Total	Penalty	CPU time (s)
1	55	75	130	130	3,534	60	40	100	60	4,540
2	45	70	115	100	2,740	55	10	65	40	4,616
3	45	70	115	110	4,282	55	15	70	50	2,437
4	75	105	180	180	4,017	80	20	100	90	4,544
5	45	70	115	100	3,754	55	20	75	55	6,112
6	45	70	115	100	3,850	70	30	100	65	4,119

Table 6: Economies of scale

6 Conclusions and further research

In this paper, we defined the problem of scheduling care tasks in a nursing home setting using a given workforce as a mixed-integer program (MIP). We presented dynamic programming (DP) approaches to obtain schedules that minimize the total earliness/tardiness penalties for the scheduled start times of all tasks. The optimizing DP approach with reduction of the state space is applicable to small problem instances while the truncated DP provides near optimal solutions in short computation times.

From a practical perspective, this study provides insights into the effects of skill substitution and large scale scheduling on the overall quality of service by using real-life nursing home data. Questions with respect to scale are especially prominent as *(1)* there is an ongoing tendency towards smaller clusters when it comes to the provision of nursing home care and *(2)* most nursing homes apply small-scale scheduling. The results of this study show that, from a client perspective, using large-scale scheduling and skill substitution is beneficial to the overall quality of service in terms of total earliness/tardiness in the obtained task schedules.

Further research could be directed into the integration of other client-perspectives, for example, related to the preferred care workers for some of the tasks. The integration of the presented approach into a personnel planning framework could be another fruitful direction for further research.

References

Baker, K. R. and G. D. Scudder (1990). Sequencing with earliness and tardiness penalties: a review. *Operations research 38*(1), 22–36.

Baptiste, P. (2000). Scheduling equal-length jobs on identical parallel machines. *Discrete Applied Mathematics 103*(1), 21–32.

Beasley, J. E., M. Krishnamoorthy, Y. M. Sharaiha, and D. Abramson (2000). Scheduling aircraft landings - the static case. *Transportation Science 34*(2), 180–197.

Begen, M. A. and M. Queyranne (2011). Appointment scheduling with discrete random durations. *Mathematics of Operations Research 36*(2), 240–257.

Bellenguez-Morineau, O. and E. Néron (2007). A branch-and-bound method for solving multi-skill project scheduling problem. *RAIRO-Operations Research 41*(02), 155–170.

Briskorn, D. and R. Stolletz (2014). Aircraft landing problems with aircraft classes. *Journal of Scheduling 17*(1), 31–45.

Burke, E., P. De Causmaeacker, G. Van den Berghe, and H. Van Landeghem (2004). The state of the art of nurse rostering. *Journal of Scheduling 6*(7), 441–499.

Cayirli, T. and E. Veral (2003). Outpatient scheduling in health care: a review of literature. *Production and Operations Management 12*(4), 519–549.

Cayirli, T., E. Veral, and H. Rosen (2006). Designing appointment scheduling systems for ambulatory care services. *Health Care Management Science 9*(1), 47–58.

Cheang, B., H. Li, A. Lim, and B. Rodrigues (2003). Nurse rostering problems-a bibliographic survey. *European Journal of Operational Research 151*(3), 447–460.

Colombo, F., A. Llena-Nozal, J. Mercier, and F. Tjadens (2011). Help wanted? providing and paying for long-term care. *OECD Policy Studies*.

Courtney, M., M. O'Reilly, H. Edwards, and S. Hassall (2009). The relationship between clinical outcomes and quality of life for resident of aged care facilities. *Australian Journal of Advanced Nursing 4*(26), 49–57.

De Bruecker, P., J. Van den Bergh, J. Beliën, and E. Demeulemeester (2015). Workforce planning incorporating skills: State of the art. *European Journal of Operational Research 243*(1), 1–16.

Eppen, G. D. (1979). Note-effects of centralization on expected costs in a multi-location newsboy problem. *Management Science 25*(5), 498–501.

Erdogan, S. A. and B. Denton (2013). Dynamic appointment scheduling of a stochastic server with uncertain demand. *INFORMS Journal on Computing 25*(1), 116–132.

Ernst, A., H. Jiang, M. Krishnamoorthy, and D. Sier (2004). Staff scheduling and rostering: A review of applications, methods and models. *European Journal of Operations Research 1*(153), 3–27.

Geerts, J., P. Willeme, and E. Mot (2012). Long-term care use and supply in europe: Projections for germany, the netherlands, spain and poland. *ENEPRI Research Report (No. 116)*.

Gertsbakh, I. and H. I. Stern (1978). Minimal resources for fixed and variable job schedules. *Operations Research 26*(1), 68–85.

Gibbs, I., D. McCauhan, and M. Griffiths (1991). Skill-mix in nursing: a selective review of the literature. *Journal of Advanced Nursing 16*(2), 242–249.

Hingstman, T., M. Langelaan, and C. Wagner (2012). De dagelijkse bezetting en kwaliteit van zorg in instellingen voor langdurige zorg. Technical report, NIVEL.

Jansen, P., A. Kerkstra, H. Huijer Abu-Saad, and J. Van der Zee (1997). Differentiated practice and specialization in community nursing: a descriptive study in the netherlands. *Health and Social Care in the Community 4*(5), 219–226.

Kaandorp, G. C. and G. Koole (2007). Optimal outpatient appointment scheduling. *Health Care Management Science 10*(3), 217–229.

Krishnamoorthy, M., A. Ernst, and D. Baatar (2012). Algorithms for large scale shift minimisation personnel task scheduling problems. *European Journal of Operational Research 219*(1), 34–48.

Lieder, A., D. Briskorn, and R. Stolletz (2015). A dynamic programming approach for the aircraft landing problem with aircraft classes. *European Journal of Operational Research 243*(1), 61–69.

Mankowska, D., F. Meisel, and C. Bierwirth (2014). The home health care routing and scheduling problem with interdependent services. *Health Care Management Science 17*(1), 15–30.

Moeke, D., G. Koole, and L. Verkooijen (2014). Scale and skill-mix efficiencies in nursing home staffing: Inside the black box. *Health Systems 3*(1), 18–28.

Moeke, D. and L. Verkooijen (2013). Doing more with less: A client-centred approach to healthcare logistics in a nursing home setting. *Journal of Social Intervention Theory and Practice 22*(2), 167–187.

Persson, T. and D. Wasterfors (2009). Such trivial matters: How staff account for restrictions of residents' influence in nursing homes. *Journal of Aging Studies 23*(1), 1–11.

Schimmelpfeng, K., S. Helber, and S. Kasper (2012). Decision support for rehabilitation hospital scheduling. *OR Spectrum 34*(2), 461–489.

Simmons, S., A. Rahman, L. Beuscher, V. Jani, D. Durkin, and J. Schnelle (2011). Resident-directed long-term care: Staff provision of choice during morning care. *The Gerontologist 51*(6), 867–875.

Sloane, P., L. Miller, J. Rader, K. Swafford, and S. Hiatt (2007). Provision of morning care to nursing home residents with dementia: opportunity for improvement? *Journal of Alzheimer's Disease and Other Dementias 5*(22), 369–377.

Spilsbury, K., C. Hewitt, L. Stirk, and C. Bowman (2011). The relationship between nurse staffing and quality of care in nursing homes: A systematic review. *International Journal of Nursing Studies 48*(6), 732–750.

van Eeden, K., D. Moeke, and R. Bekker (2014). Care on demand in nursing homes: a queueing theoretic approach. *Health care management science*, 1–14.

Vanberkel, P. T., R. J. Boucherie, E. W. Hans, J. L. Hurink, and N. Litvak (2012). Efficiency evaluation for pooling resources in health care. *OR Spectrum 34*(2), 371–390.

Wallace, R. B. and W. Whitt (2004). Resource pooling and staffing in call centers with skill-based routing. *Operations Research 7*(4), 276–294.

Wright, D. and S. Mahar (2013). Centralized nurse scheduling to simultaneously improve schedule cost and nurse satisfaction. *Omega 41*(6), 1042–1052.