Vrije Universiteit Amsterdam

Research Paper

Optimizing shift scheduling and task allocation in long-term care facilities to reduce waiting times

Author:
Eva van Brummelen

Supervisor:
René Bekker

July 11, 2018

VU VRIJE UNIVERSITEIT AMSTERDAM

Optimizing shift scheduling and task allocation in long-term care facilities to reduce waiting times

Eva van Brummelen
e.van.brummelen@student.vu.nl

Vrije Universiteit Amsterdam, Amsterdam 1081 HV, NL
Host organization: Siza, Arnhem 6816 RP, NL

July 11, 2018

Abstract. The goal of this research paper is to minimize waiting times in long-term care facilities. This is done by building a model that optimizes shift and task schedules in two consecutive steps. First, a shift schedule is created using an Integer Linear Programming (ILP) model. Based on the shift schedule, a task schedule is created using a heuristic algorithm. The input of the model consists of preferred activity times of the residents of the care facility and the number of staffing hours available. The model is able to handle hierarchical qualification levels and group activities. To evaluate the model, it is applied to a long-term care facility called Siza. Results show that the model can create a shift schedule that leads to lower waiting times than Siza's original shift schedule.

Keywords: Long-term care, Client-centered, Shift scheduling, Task allocation, Hierarchical qualification levels, Group activities

Preface

This research paper is written as a part of the Master Business Analytics curriculum. It is a four-week project that is done individually. The purpose of the paper is for the student to gain experience on clearly describing a problem. The target audience of the paper is a manager who has some general knowledge about the subject. While writing the research paper, the student should assume that the manager does not have a lot of time, which requires the paper to be succinct.

The research problem that I will address in this paper is the reduction of waiting times in long-term care facilities by determining shift schedules and doing task allocation. The subject of this research paper is an extension of the course "Project Optimization of Business Processes", a project I did in January 2018 at the Vrije Universiteit.

To make the research more realistic, I had contact with Joost ter Velde, who works at Siza. Siza is a care facility that provides care for people with disabilities. Siza wishes to focus more on the waiting times of their clients when creating schedules.

I would like to thank my supervisor René Bekker from the Vrije Universiteit for his guidance, feedback and the weekly meetings that gave me new ideas for my research. I would also like to thank Dennis Moeke from the Vrije Universiteit for bringing me in contact with Joost ter Velde from Siza. Lastly, I would like to thank my fellow student Chantal Leeuwestein. She is constructing a similar research, which allowed us to engage in discussions about the research.

Eva van Brummelen
Amsterdam, June 2018

Table of Contents

1 Introduction

The number of elderly people in the Netherlands is increasing, caused by an increasing life expectancy. This results in a greater demand for long-term care, which contributes to higher costs [1][2]. Simply reducing expenses is not an option, because it will result in a poor quality of long-term care. Instead, to keep long-term care affordable, smart and efficient innovations should take place. At the same time, the clients and the quality of their care are receiving an increasing amount of attention. For the quality of long-term care it is desirable that clients have a voice in their care. For example, they want to determine at what time of the day their care takes place.

This research focuses on long-term care facilities. Residents who live in these facilities have activities scheduled throughout the day. These schedules are not always based on the preferences of the residents. The goal of this research is to build a model that creates a shift schedule and task allocation that takes into account the preferences of the residents.

The scheduled activities mentioned before are also called scheduled demand. Examples of scheduled demand are help with getting dressed, giving medicines and assistance during eating. Scheduled demand can either be individual or in a group, i.e. there are individual activities and group activities. Individual activities require one care worker for one resident, while group activities require one care worker for multiple residents. An example of a group activity is drinking coffee in a common room with multiple residents and one care worker. Next to scheduled activities, there are also unscheduled activities taking place in long-term care facilities. This is also called unscheduled demand. These are activities that happen at random times. An example is assistance with going to the bathroom. This is something that is not known in advance.

Demand requires a care worker that has a specific qualification level that suits the type of the demand. In this research paper it is assumed that qualification levels are hierarchical. This means that demand that requires a certain qualification level can also be performed by a care worker that has a higher qualification level, but not by a care worker with a lower qualification level. For example, a level 1 care worker can do level 1 tasks, a level 2 care worker can do both level 1 and 2 tasks, etc.

The aim of this research is to create a model that minimizes the absolute waiting times of residents in a long-term care facility. Waiting time is defined as the difference between the preferred starting time of an activity and the actual starting time. Thus, when an activity starts before or after the preferred starting time, there is waiting. A model is created that first creates a shift schedule, and then uses this schedule to create a task schedule for a given day. The model can handle hierarchical qualification levels and group activities. The number of available staffing hours in this model is assumed to be known in advance, which is a realistic assumption in a long-term care facility.

Bekker et al. (2015) describe a Mixed-Integer Linear Programming Model to create shift schedules in long-term care facilities [3]. This model does not take into account hierarchical qualification levels and group activities. A research by Lieder et al. (2015) focuses on task allocation in long-term care facilities where they minimize waiting times [4]. They do this using Dynamic Programming in combination with heuristics. They do not create a shift schedule themselves, because they assume it is given.

Literature about the logistics within health care for elderly people is scarce. The two papers discussed in the previous paragraph are one of the few papers about this topic. There exists some more literature about task allocation, but those papers focus on home care instead of resident homes. One of these papers is written by Mankowska et al. (2014) and describes a model to create a daily planning for a home care company [5]. The model uses Mixed-Integer Linear Programming to create routes for employees. In long-term care facilities the distances between the rooms of residents are negligible, which makes the problem differ from Mankowska's paper.

The structure of this research paper is as follows. Section 2 discusses the wishes of Siza in more detail, including the data they have provided. After that, a solution method is presented in Section

3. This solution method consists of two parts, an Integer Linear Programming model for shift scheduling and a heuristic algorithm for task scheduling. In Section 4 the solution method is applied to the specific case of Siza, and results are shown and discussed. Section 5 and 6 contain a conclusion and a discussion respectively. The conclusion also contains an advice for Siza.

2 Information about Siza

To be able to make a realistic model, this research focuses on a long-term care facility called Siza[1], and in particular on one resident house of Siza. There are 15 residents living in this house, who all have a disability but can still do quite some things independently. Siza has provided data of these 15 residents, which will be used as input for the model.

To gather more information about Siza and the way they work, Joost ter Velde is interviewed. Joost works at Siza as a team leader and knows a lot about long-term care. He also works at Academy Het Dorp, a company that develops and tests innovations and technology in health care. This section discusses the information from the interview and the data that was provided.

2.1 Long-term care at Siza

At Siza, there are two types of care workers that provide care to residents. They can have qualification level 3 or 4[2]. There are also care workers with other qualification levels, but these are not relevant for this research since they do not perform resident-related tasks, such as cleaning. Level 3 care workers can provide all basic care. Level 4 care workers can also provide basic care, but in addition they are client counselors. This means that they also have guidance conversations with the residents. In the resident house used in this research, there is one care worker who has qualification level 4. This care worker spends a part of the job on client counseling, and the remaining part on level 3 tasks. A requirement of Siza's shift schedule is that there is always at least one care worker with qualification level 3 or higher present.

The biggest part of the scheduled demand is from 7:00 to 23:00. Therefore, it is interesting to focus on that time frame. Shifts can have a duration of 3, 4, 6, 7, 8 or 9 hours. For Siza, short shifts are desirable because they allow for more flexible planning. However, employees usually prefer longer shifts. Shifts of 6 hours and longer contain a break, but this is ignored in the model for simplicity. In reality, care workers usually take a break when it suits their schedule.

As mentioned earlier, the residents in this particular resident home of Siza can still do quite some things independently and they are not in need of constant care. There are about 15 residents in this house and they all have a preference time for each of their activities. For example, Pete wants his morning care at 8 AM. The activity schedule of a resident is created by the resident and a level 4 care worker. An activity schedule contains, for each activity, the desired starting time and the estimated duration of the activity. It is desirable that the actual starting time of an activity does not deviate more than 15 minutes from the desired starting time.

Next to scheduled demand, there is also unscheduled demand. Examples of unscheduled demand are help with going to the bathroom and help with unpacking groceries. Residents have a button in their room which they can press when they need help from a care worker. When the button is pressed, the care workers receive a call and they discuss who will handle the unscheduled demand. For each resident it is known how much unscheduled demand they have on average and how long it usually takes.

The severity of the disability of a resident determines how many hours of care a resident should receive. This is also called an indication[3]. The total number of care hours available in a resident house can be calculated by adding up the care hours of all residents. This total number of available care hours is used to determine the shift schedule in the resident house.

Currently, Siza uses a planning tool to create shift schedules. Task scheduling is done manually, based on the insight of the planner and the employees. Each care worker has a list of scheduled

[1] `https://www.siza.nl/`

[2] Level 3 is called "MBO niveau 3, Verzorgende-IG" in Dutch and level 4 is called "MBO niveau 4, Verpleegkundige" (`https://www.youchooz.nl/opleidingen-verpleging-verzorging`)

[3] `https://www.monitorlangdurigezorg.nl/over-\mlz/begrippen/indicatie`

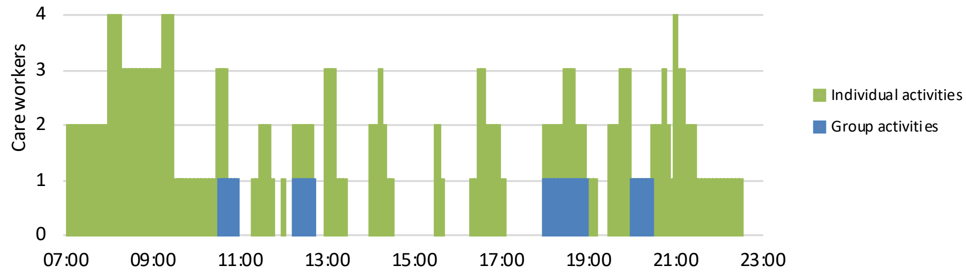
demand they have to handle on a day in a specified order. When there is unscheduled demand, the care workers consult each other to determine who will handle the request. There is no specific protocol for this.

2.2 Data of preferred activity times

Siza has provided data of the resident house mentioned earlier. This data consists of preferred activity times of the 15 residents for one week. The model is created on a daily basis, meaning that it will create a schedule for each day separately. Since the preferred activity times in the data are very similar on each day of the week, this research paper will only focus on the preferred activity times on Monday.

On Mondays there are 66 scheduled activities in total. The number of care workers needed to satisfy this demand is visualized in Figure 1. The bars indicate the number of care workers needed to satisfy the demand on each point in time.

Figure 1: Scheduled demand on Monday



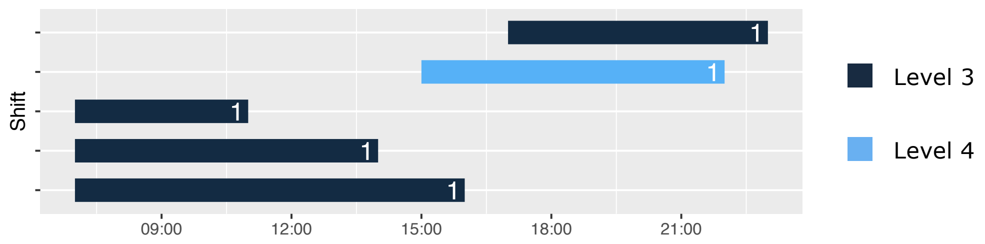
From the 66 scheduled activities, 46 are individual activities and 20 are group activities. There are four types of group activities: coffee in the morning, lunch, dinner and coffee in the evening. The residents have breakfast on their own, because not everyone wakes up at the same time. There is only one activity type that requires qualification level 4, which is a “guidance meeting”. On Mondays, there are two residents who have this activity. An example of the data is shown in Table 1.

Resident ID	Activity type	Preferred starting time	Type	Qualification level	Expected duration (in minutes)
R4	Morning care	7:05	individual	3	40
R2	Morning care	7:05	individual	3	10
R11	Medication	7:30	individual	3	15
...	...	...	...	...	...
R14	Lunch	12:15	group	3	30
R7	Lunch	12:15	group	3	30
R4	Lunch	12:15	group	3	30
...	...	...	...	...	...

Table 1: Example of activities data

The current shift schedule of Siza uses is known and is visualized in Figure 2. The number on a bar indicates the number of shifts that are of a particular type. In this case, there are five types of shifts, and each of these types appears once. The duration of all shifts together is 33 hours. Unfortunately the data does not include the qualification level of each shift. Therefore, it is assumed that the shift from 15:00 to 22:00 has qualification level 4, and the other shifts have qualification level 3. This assumption is in line with the two level 4 activities in the data and the information Siza has provided.

Figure 2: Current shift schedule of Siza



There are five residents who have unscheduled demand. Each of these residents has approximately 3 to 10 times unscheduled demand on a day. The durations of these unscheduled activities vary from 5 to 10 minutes. The demand is random, so it cannot be scheduled.

3 Solution method

Creating a shift schedule and a task schedule are two closely related problems. To be able to allocate tasks to care workers, the shift schedule of the day should be known. In this research these two tasks are therefore executed consecutively. First a shift schedule is created, which is then used for task allocation. Both shift scheduling and task allocation are done on a daily basis.

3.1 Shift scheduling

Minimizing waiting times by means of shift scheduling is done with an Integer Linear Programming (ILP) model. This is a model with an objective function, decision variables and constraints. The formulation of the model is done in a general way. When the model is applied to a specific long-care facility, the model can be customized by adding additional constraints. The model is inspired by the Mixed-Integer Linear Programming model build by Bekker et al. [3].

The ILP model is a simplification of reality. It does not distinguish unique care workers or residents. It only looks at the total amount of care worker hours and the total amount of demand hours for each activity at each point in time. The sets, parameters and decision variables used in the model are listed in Table 2. The next paragraph specifies the values of the sets and parameters for Siza.

Sets	
T	All time intervals
T^*	Time intervals in which shifts are allowed to start
I	Qualification levels
J	Shift lengths
A	Activity types
A_i	Activity types that require care worker i or higher with $\cup_{i \in I} A_i = A$
Parameters	
C_i^{min}	Minimum number of care workers with qualification level i or higher present
C	Maximum number of care worker hours available
$L_{a,t}$	Number of residents that want to do activity a at time t
S_a	Maximum number of residents that can do activity a with one care worker
Decision variables	
$x_{i,j,t}$	The number of care workers with qualification level i that start a shift of length j at the beginning of time interval t
$c_{i,t}$	The staffing level of care workers with qualification level i during time interval t
$c^*_{i,t}$	The number of care workers working at qualification level i during time interval t
$y_{a,t}$	The number of care workers that do activity a during time interval t
$l_{a,t}$	The number of residents that do activity a during time interval t
$q_{a,t}$	The backlog of activity a at the start of time interval t

Table 2: Sets, parameters and decision variables of the model

In the case of Siza, the time intervals in T are {7:00 - 7:05, 7:05 - 7:10, ..., 22:55 - 23:00}. An interval size of 5 minutes is used because all timestamps in the data of Siza are multiples of 5 minutes. This reduces complexity of the problem. Shifts are only allowed to start at the beginning of the time intervals in T^*, which is every 30 minutes {7:00, 7:30, ...}. The qualification levels in I are level 3 and 4 and the shift lengths in J are {3, 3.5, 4, ... 9} hours.

An example of how an activity is handled in the model is the following. Suppose Pete needs morning care (individual) from a care worker with qualification level 3. He wants this at 8:00 and the expected duration is 15 minutes. Then 1 is added to $L_{a,t}$ for $t \in$ {8:00 - 8:05, 8:05 - 8:10, 8:10 -

8:15} and a equal to that particular activity. Because it is an individual activity, S_a is set equal to 1.

If $S_a > 1$ then a is a group activity. In the case of Siza, group activities are always with one care worker, so setting S_a equal to the number of residents would suffice. However, in a more general setting it might be necessary to have the number of care workers increase proportionally to the number of residents doing an activity. For example, when $S_a = 3$, a second care worker should step in when a fourth resident joints the group activity.

As explained in Section 2, there needs to be at least one care worker of qualification level 3 present at all times. Therefore, $C_{lvl3}^{min} = 1$ and $C_{lvl4}^{min} = 0$. Another parameter is C, which is the available number of care worker hours. C is set equal to 33, because the current shift schedule of Siza consists of 33 hours in total. The ILP model is formulated below.

$$\text{minimize} \quad \sum_{t \in T} \sum_{a \in A} q_{t,a} \tag{1}$$

$$\text{subject to} \sum_{t \in T^*} \sum_{i \in I} \sum_{j \in J} x_{i,j,t} \cdot j \leq C \tag{2}$$

$$c_{i,t} = \sum_{t' \in T^*} \sum_{j \in J : j+t' \geq t} x_{i,j,t'} \quad \forall i \in A_i, t \in T \tag{3}$$

$$c^*_{i,t} \leq \sum_{i' \in I : i' \geq i} c_{i',t} \quad \forall i \in I, t \in T \tag{4}$$

$$\sum_{t \in T} \sum_{i \in I : i \geq i_{min}} c^*_{i,t} \leq \sum_{t \in T} \sum_{i \in I : i \geq i_{min}} c_{i,t} \quad \forall i_{min} \in I, t \in T \tag{5}$$

$$l_{a,t} \leq S_a \cdot y_{a,t} \quad \forall a \in A, t \in T \tag{6}$$

$$\sum_{a \in A_i} y_{a,t} \leq c^*_{i,t} \quad \forall i \in I, t \in T \tag{7}$$

$$q_{a,t+1} \geq q_{a,t} + L_{a,t} - l_{a,t} \quad \forall a \in A, t \in T \tag{8}$$

$$q_{a,t_{max}} = 0 \quad \forall a \in A, t_{max} = \max_{t \in T} t \tag{9}$$

$$c^*_{i,t} \geq C_i^{min} \quad \forall i \in I, t \in T \tag{10}$$

$$x_{i,j,t} \in \mathbb{Z}_0 \quad \forall i \in I, j \in J, t \in T \tag{11}$$

$$c_{i,t}, c^*_{i,t} \in \mathbb{Z}_0 \quad \forall i \in I, t \in T \tag{12}$$

$$y_{a,t}, l_{a,t}, q_{a,t} \in \mathbb{Z}_0 \quad \forall a \in A, t \in T \tag{13}$$

The objective function (1) of the model is to minimize the total backlog. For each activity and for each point in time, the backlog is the remaining work that has not been done, but ideally should have been done already. It is the backlog from the previous time interval, including the new demand and excluding the demand that has already been handled. The backlog is calculated as $q_{a,t+1} = \max\{0, q_{a,t} + L_{a,t} - l_{a,t}\}$. This is a non-linear function, which is made linear by using two constraints instead of one, namely Equations (8) and (13). In the ILP model, it assumed that activities cannot start before their preferred starting time. Therefore, the backlog cannot be negative.

The first constraint, Equation (2), models the available working hours. The total working hours are summed up, and this sum cannot exceed the number of available working hours. Equation

(3) ensures that there is a relationship between the staffing levels and the shifts. For each time interval, it is calculated which shifts are still busy at that point. In both Equation (2) and (3) the variables $x_{i,j,t}$ are defined only for $t \in T^*$ and not for $t \in T$, because because shifts can only start at beginning of the intervals in T^*

Equations (4) and (5) model the hierarchical qualification levels. The first equation says that the actual number of care workers doing a level i job should be smaller or equal to the total number of care workers that are qualified to do that job. The second equation makes sure that the care workers are not used for more than one qualification level. In the case of Siza, the constraints can be simplified to

$$c^*_{3,t} \leq c_{3,t} + c_{4,t}, \qquad c^*_{4,t} \leq c_{4,t} \qquad \forall t \in T \tag{14}$$

$$c^*_{3,t} + c^*_{4,t} \leq c_{4,t} + c_{4,t}, \qquad (c^*_{4,t} \leq c_{4,t}) \qquad \forall t \in T \tag{15}$$

Equation (6) relates the available staff to the activities. The number of residents that do activity at time t ($l_{a,t}$) should be smaller or equal to the number of employees that do that activity, multiplied with the number of residents that can do that activity with one employee. For example, suppose that activity a is morning care, and individual activity. Then S_a is equal to 1. This results in the constraint $l_{a,t} \leq 1 \cdot y_{a,t}$. If there would be a demand of $l_{a,t} = 2$, the number of care workers doing activity a should be $y_{a,t} = 2$. Furthermore, Equation (7) models the relation between the staffing levels and the number of care workers doing certain activities. The total number of employees doing tasks of qualification level i should not exceed the staffing level of employees working at qualification level i.

Equation (8) models the backlog, as explained earlier. Equation (9) makes sure that there is no backlog at the end of the day. The result from this constraint is that the problem becomes infeasible if the number of working hours is not sufficient. Equation (10) ensures that the minimum number of required care workers is present.

Depending on the preferences of a long-term care facility, the ILP model could be expanded or adjusted. Additional terms could be added to the objective function or constraints can be added or adjusted. For example, when a facility wants at least two shifts of length $j = 8$ hours, the following constraint could be added:

$$\sum_{t \in T} \sum_{i \in I} x_{i,8hr,t} \geq 2 \tag{16}$$

The ILP model in this research does not focus a lot on qualification levels, because Siza has not indicated their preferences about this. However, another long-term care facility might have specific preferences regarding qualification levels. If so, additional constraints could be added. For example, they might want exactly one shift with qualification level 4, and the shift must have a length of 8 or 9 hours. Then the following two constraints would model this:

$$\sum_{t \in T} \sum_{j \in \{8hr, 9hr\}} x_{lvl4,j,t} = 1 \tag{17}$$

$$\sum_{t \in T} \sum_{j \in J \setminus \{8hr, 9hr\}} x_{lvl4,j,t} = 0 \tag{18}$$

3.2 Task allocation

Based on the shift schedule determined by the ILP model in Section 3.1, a task schedule is created using a heuristic. This heuristic allocates activities to care workers at specific times. In order to do this, an algorithm is developed that walks through time. The time starts at 7:00 and ends at 23:00, with time steps of 5 minutes. This is consistent with the data, because all timestamps in the data are multiples of 5 minutes. While walking through time, the algorithm uses heuristic rules to determine which activities are allowed to start. Different scenarios are investigated using a tree structure.

At each point during the walk (7:00, 7:05, 7:10, etc.), several events take place and decisions need to be made. The pseudo code of this process is shown in Algorithm 1.

Algorithm 1 Heuristic Pseudo code

```
initialize clients, shifts and preferred activity schedules

sort preferred activity schedules

for time = 7:00, 7:05, ..., 23:00 do
    stop activities that are finished
    stop shifts that are finished & start shifts that should start
    start activities from the preferred activity schedules
    update the status of clients and care workers (busy/not busy)

calculate waiting times
```

The first two functions that are executed at each point in time are quite straightforward. The most interesting part of the algorithm is the "start activities from the preferred activity schedules" function. This function is the heuristic part of the algorithm. The activities consists of both individual and group activities. Several rules have been applied to determine which activities are allocated to which care worker and at what time. This is done as follows. At each point in time, the activities are inspected in a particular order that indicates their priority. The heuristic algorithm looks at the activities in that order and for each activity tries to find a care worker that is available. A care worker is available when his or her overtime resulting from doing that activity does not exceed 10 minutes and when the care worker is not busy with another activity. The order in which the activities are sorted is listed below. Table 3 shows a short example of the way activities are sorted.

1. **Preferred starting time**: Activities with an earlier preferred starting time first, i.e. First Come First Serve (FCFS).
2. **Activity size**: Group activities first, because it is preferable to keep a single resident waiting instead of an entire group.
3. **Qualification level**: Activities with a higher qualification level are first, because they have less care workers to choose from.
4. **Duration**: Shorter activities first, since this will result in a lower waiting time. This is called Shortest Job First (SJF).

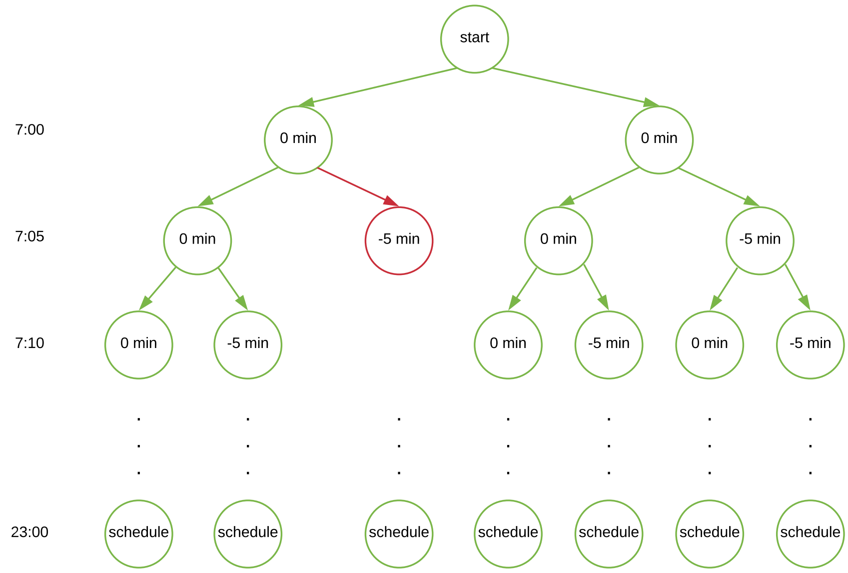
Order	Preferred starting time	Activity size	Qualification level	Expected duration (minutes)
1	8:00	individual	4	20
2	8:00	individual	3	20
3	8:00	individual	3	30
4	8:10	group	3	10
5	8:10	individual	3	10
5	8:10	individual	3	15

Table 3: Example of sorting activities

In addition to this ordering of activities, the algorithm always tries to match an activity to a care worker with the lowest qualification level possible. This is will leave care workers with higher qualification idle more often, which saves them for activities in the future that do need a care worker with a high qualification level. This rule is not the same as sorting criteria number 3.

Sometimes it might be useful to let some activities start before their preferred starting time, to remove some pressure from busy periods. Therefore, the parameter *maximumEarliness* is used to determine what the maximum earliness is for starting individual activities. If *maximumEarliness* equals 5 minutes, it means that activities can start 5 minutes before their preferred starting time. However, allowing activities to start sooner is not always desirable. Therefore, a tree structure is used that tries different scenarios. At each point in time, two scenarios will be considered, namely *maximumEarliness* equal to 0 minutes and *maximumEarliness* equal to 5 minutes. This process is shown in Figure 3, where the times are shown on the left.

Figure 3: Tree Heuristic



The figure shows that each node has two splits, resulting in the nodes "0 min" and "-5 min". If there is a node "-5 min" in which it is not possible to start at least one activity 5 minutes earlier, that node will be exactly the same as the node on its left ("0 min"). To prevent the algorithm from executing scenarios that are exactly the same, such nodes will be pruned. An example of such a node is the red node in Figure 3. That node will not be investigated any further, because it is the same as its sibling node on the right.

After the entire tree is build, there are many possible task schedules to choose from. In Figure 3 these are the "schedule" nodes at the bottom, which contain possible task schedules. For each task schedule, the average absolute waiting time is calculated and used to select the best task schedule. In this model, the best task schedule is defined as the task schedule with the lowest average waiting time.

Creating an entire tree is computationally expensive. In the worst-case scenario each node has two child nodes. Fortunately this will not happen in practice because of the relatively low amount of activities in the data. Because of this low amount, there will not be that many cases where nodes have a "-5 min" split to the right. The depth of the tree is equal to $n = 193$, because the algorithm goes from 7:00 to 23:00 with time steps of 5 minutes. Theoretically, the number of nodes in each layer of the three could double compared to the layer above it. If that would happen, the number of nodes in the bottom of the tree equals 2^{n-1}. This is the number of schedules that is created.

In summary, the heuristic algorithm loops through time. At each point in time it starts activities using a specific way of sorting. On top of this, many different scenarios (task schedules) are considered by varying the *maximumEarliness* parameter using a tree structure. Eventually, the task schedule with the lowest average waiting time is chosen.

4 Experiments and Results

The model presented in the previous section is a general model. Its results depend on the input data and on additional constraints. To get more insights into the quality of the model, this section presents several experiments. First, the current shift schedule of Siza is considered, because it can serve as a benchmark. This is done with and without the heuristic task allocation. Then, the ILP model and the heuristic are used to create a new staffing and task schedule for Siza. This will be compared to the original staffing schedule. To make the comparison more realistic, another experiment is conducted where additional constraints are added to the ILP model. The last experiment considers unscheduled demand.

4.1 Experiment 1: Siza's shifts as a benchmark

The first experiment uses the existing shift schedule of Siza to be able to use it as a benchmark. This means that the ILP model is not used for shift scheduling. Unfortunately, the task schedule from Siza is not known, so assumptions about this need to be made. Two scenarios are investigated. The first one assumes Siza uses First Come First Serve (FCFS) and the three structure, but no additional sorting of activities. The second scenario uses the heuristic task allocation algorithm.

FCFS

Using FCFS and the tree structure, a task schedule can be created. Activities are randomly sorted, because the order of activities influences the task schedule. To get reliable results, the activities are randomly sorted ten times. Each time a shift schedule is created using FCFS. The average waiting time of the ten schedules is 5.36 minutes per activity. The lowest and highest average waiting times found are 4.92 and 5.76 minutes respectively.

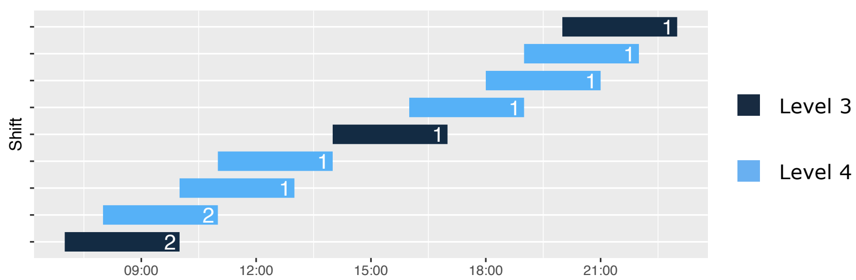
Task allocation heuristic

Using the task allocation heuristic with Siza's shift schedule, a task schedule is created. Doing this results in 393,216 different task schedules, where the best task schedule has an average waiting time of 4.77 minutes. This experiment shows that applying the task allocation heuristic has a positive effect on the average waiting time. The average waiting time is reduced from 5.36 to 4.77 minutes, and even the lowest average waiting time found using FCFS is higher than the one found using the task allocation heuristic.

4.2 Experiment 2: Model without additional constraints

The second experiment uses the ILP model to create a staffing schedule and the heuristic algorithm to create a task schedule. The resulting staffing schedule is shown in Figure 4. All shifts have a length of 3 hours, because choosing shorter shifts results in an optimal staffing schedule. However, in reality the employees of a long-term care facility do not like many short shifts. There are also many level 4 shifts (the light blue bars), because the model does not favor one level over the other.

Figure 4: Experiment 2 staffing schedule



The staffing schedule is used as input for the heuristic algorithm to determine staffing levels. The algorithm has generated 327,680 possible schedules. The best schedule has an average waiting time of 2.65 minutes per activity. The worst schedule as an average waiting time of 4.77 minutes per activity. This means that using the ILP model for shift scheduling results in lower waiting times than the shift schedule Siza is currently using.

4.3 Experiment 3: Model with additional constraints

The current shifts of Siza have lengths 4, 6, 7 (2x) and 9 hours. To make the comparison between the model in this research and the current shift planning fairer, additional constraints are added to the ILP model. One of these constraints allows only one shift of length 4 or shorter. This constraint is formulated as

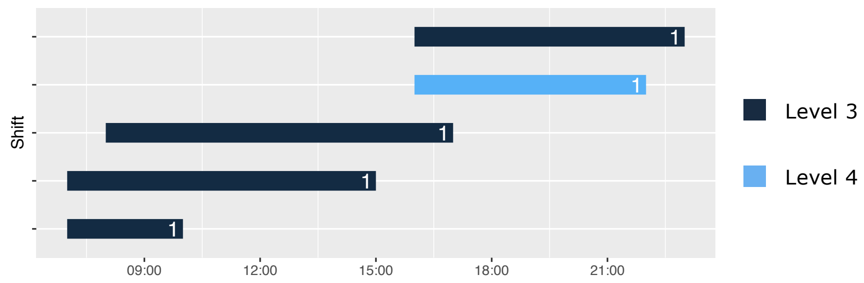
$$\sum_{t \in T} \sum_{i \in I} \sum_{j \in \{3hr, 4hr\}} x_{i,j,t} \leq 1 \tag{19}$$

The second constraint allows only one shift with qualification level 4. The constraint is formulated as

$$\sum_{t \in T} \sum_{j \in J} x_{lvl4,j,t} \leq 1 \tag{20}$$

The resulting staffing schedule is shown in Figure 5. It can be seen that these shifts are much more similar to the original shifts of Siza (see Figure 2 in Section 2.2). This is caused by the additional constraints.

Figure 5: Experiment 3 staffing schedule



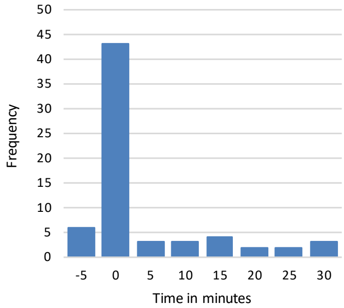
The task schedule produced by the heuristic algorithm is shown in Figure 7c. The algorithm created 393,216 task schedules. The worst task schedule has an average waiting time of 5.00 minutes and the best task schedule has an average waiting time of 3.33 minutes.

4.4 Comparing Experiments 1-3

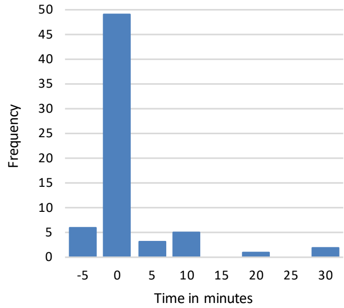
This section will compare Experiments 1-3 from the previous sections. From Experiment 1, only the schedule that was created using heuristic task allocation is used for comparison, because the FCFS task schedule might not be used by Siza in practice. The experiments can be compared in several ways. The shift schedules have already been compared in the previous section. This section will compare the waiting times and task schedules resulting from the experiments.

The waiting time for each task in a task schedule can be calculated. This is done assuming that there is no unscheduled demand and that the tasks are all executed exactly according to the task schedule. These waiting times are shown in Figure 6. Waiting times equal to or close to 0 minutes are preferred over waiting times that lie further away from 0 minutes. It can be seen that Experiment 2 has the most waiting times equal to 0, and its other waiting times lie very close to zero. Experiment 3 has some higher waiting times than Experiment 2, due to the additional constraints for the shift schedule. Experiment 1, that uses the current Shift schedule of Siza, has the most waiting times that deviate from 0 minutes.

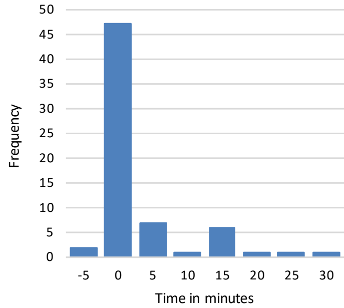
Figure 6: Waiting times of Experiments 1-3



(a) Experiment 1: Siza's shifts with heuristic task allocation

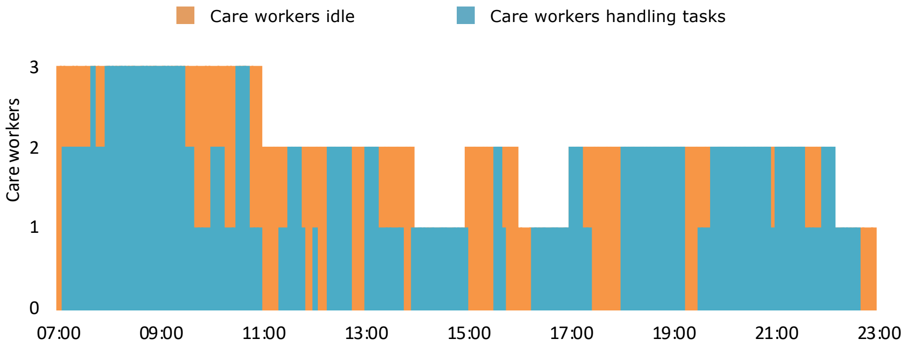


(b) Experiment 2: No additional constraints

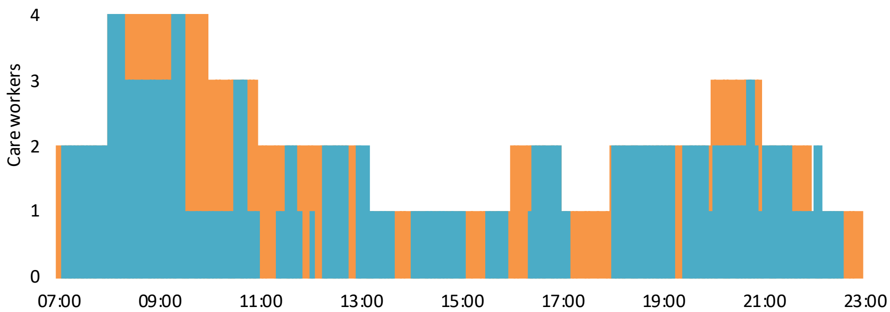


(c) Experiment 3: Additional constraints

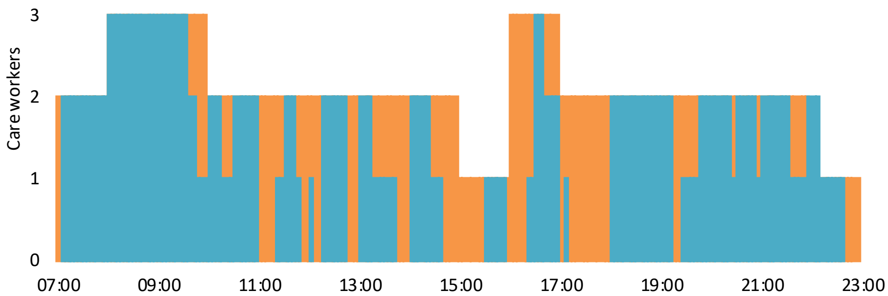
Figure 7: The task schedule and staffing levels of Experiments 1-3



(a) Experiment 1: Siza's shifts with heuristic task allocation



(b) Experiment 2: No additional constraints



(c) Experiment 3: Additional constraints

Another way to compare the experiments is by looking at the task schedules. These are visualized in Figure 7. The figure shows the number of idle care workers (orange) and the number of care workers handling tasks (blue) at each point in time. The total staffing level at each point is time is calculated by adding these two values together. When a care facility has unscheduled demand, it is desirable that there are some idle moments throughout the day. Unscheduled demand can then be handled at those idle moments.

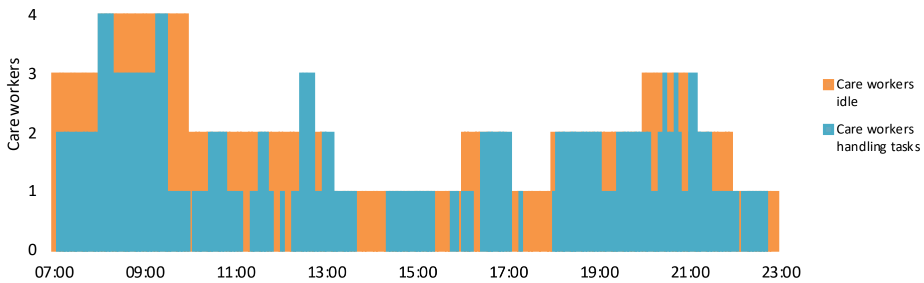
4.5 Experiment 4: Unscheduled demand

The main focus of this research is on scheduled demand because unscheduled demand is random and can therefore not be planned exactly. Still, there is some data about the frequency and duration of the unscheduled demand.

One way to incorporate the unscheduled demand into the shift scheduling model is by adding "average" unscheduled demand to the data. This is done as follows. When, for example, a resident has approximately 8 times unscheduled demand of approximately 5 minutes, 8 blocks of 5 minutes are added to the activities data. This is done by evenly spreading the blocks across the day. Shift scheduling and task allocation are then done based on this increased amount of activities. But after task scheduling is done, the unscheduled activities are removed from the schedule, because the actual times and occurrences of unscheduled activities are unknown.

The experiment described above is performed using the model described Experiment 2. This model uses the ILP model for shift scheduling and the heuristic algorithm for task allocation, and it does not use additional constraints. The resulting task schedule is visualized in Figure 8. As mentioned before, the task schedule does not include the unscheduled demand. The orange blocks in the figure are moments when care workers are idle. These idle moments can be used to handle possible unscheduled demand. The figure can be compared to Figure 7b, where no unscheduled demand was taken into account. It can be seen that the current task schedule spreads the idle moments more evenly throughout the day.

Figure 8: The task schedule and staffing levels of Experiment 4



There will be no focus on the resulting waiting times, because this requires simulation of unscheduled demand. That is beyond the scope of this research paper.

5 Conclusion and Advice

The goal of this research was to build a model that minimizes absolute waiting times in long-term care facilities. This is done using an ILP model for shift scheduling and a heuristic algorithm for task allocation. The model is applied to the long-term care facility Siza, using different experiments that were described in Section 4. The resulting average absolute waiting times are summarized in Table 4.

		Average absolute waiting time (in minutes)	Description
Experiment 1	a	4.92	Siza shift schedule + FCFS
	b	4.77	Siza shift schedule + Task allocation heuristic
Experiment 2		2.65	Model
Experiment 3		3.33	Model + additional constraints

Table 4: Summary of results

It can be concluded that the ILP for shift scheduling gives lower waiting times than the current shift schedule of Siza. When adding more constraints to the model, the waiting times increase but are still lower than when the current shift schedule of Siza is used. In Experiment 1, the task scheduling heuristic is compared with First Come First Serve (FCFS). Using the heuristic instead of FCFS resulted in a lower average waiting time. Experiment 4 included unscheduled demand in the model. This resulted in idle moments being spread more evenly throughout the day.

The advice for Siza is to consider using the ILP model for shift scheduling because it can lower the waiting times of their residents. They should first test the shift schedule to make sure it is successful in practice. The task allocation algorithm has also shown to be successful, but cannot be implemented yet. Before the task schedule is used, Siza should make decisions about how unscheduled demand will be taken into account in the task schedule. Care workers could handle unscheduled demand on a FCFS basis or use a more advanced set of rules to handle it.

6 Discussion

The time span of this research is four weeks. Therefore, it was required to limit the scope. If there would be more time, two interesting improvements to the model could be made. First, the task allocation heuristic could be improved. There could be more splits made in the tree structure ("-10 minutes" and "-15 minutes") to search through more possible task schedules. Increasing the number of splits will make the running time of the algorithm explode. Therefore, more advanced pruning criteria need to be added to the heuristic.

Another interesting improvement of the model would be to create interaction between shift scheduling and task allocation. This could be done as follows. When a shift schedule and task schedule are determined, the waiting time of each activity can be calculated. When there seem to be periods with very high waiting times, the task schedule could pass this on to the shift scheduling model. Then a new shift schedule could be created using this new information. This is an iterative procedure.

References

1. Rijksinstituut voor Volksgezondheid en Milieu (2018). Levensloop en zorgkosten https://www.rivm.nl/bibliotheek/rapporten/270082001.pdf

2. Centraal Bureau Statistiek (2018). Zorguitgaven stijgen in 2017 met 2,1 procent https://www.cbs.nl/nl-nl/nieuws/2018/22/zorguitgaven-stijgen-in-2017-met-2-1-procent

3. Bekker, R., Moeke, D., & Schmidt, B. (2018). Keeping pace with the ebbs and flows in daily nursing home operations. *Health Care Management Science*, 2018, 1-14. https://doi.org/10.1007/s10729-018-9442-x

4. A. Lieder, D. Moeke, G. Koole, R. Stolletz, Task scheduling in long-term care facilities: A client-centered approach, *Operations Research for Health Care* (2015), http://dx.doi.org/10.1016/j.orhc.2015.06.001

5. Mankowska, D.S., Meisel, F. & Bierwirth, C., The home health care routing and scheduling problem with interdependent services, *Health Care Management Science* (2014) 17: 15. https://doi.org/10.1007/s10729-013-9243-1