Annualized hours

Comparing an exact optimization model with its approximation

M.E. Bouwmeester

TUDelft

Annualized hours

Comparing an exact optimization model with its approximation

by

M.E. Bouwmeester

to obtain the degree of Bachelor of Science
at the Delft University of Technology,
to be defended publicly on Friday June 21, 2019 at 14:00 AM.

Student number:	4323785	
Project duration:	Februari 11, 2019 – June 21, 2019	
Thesis committee:	Dr. ir. J. T. van Essen	TU Delft, supervisor
	Dr. J. L. A. Dubbeldam	TU Delft
	Drs. E. M. van Elderen	TU Delft
	Dr. E. van der Veen	ORTEC

Abstract

In this thesis, we propose two mixed integer linear program formulations for an optimization problem that incorporates annualized hours: an exact one and an approximation. The objective of our model consists of three weighted parts: a part which minimizes the difference between working hours and contract hours for each employee per week, a part which minimizes over and under staffing, and a part which minimizes the difference between contract hours and working hours for each employee over the total planning period. Additionally, the working hours need to be distributed over shifts of a fixed shift duration. We also consider an extension where skills are introduced. In this case, employees can only work on a task for which they are qualified.

To test the proposed formulations, a random data generator is provided by ORTEC. The model should be solvable for a data set up to 100 employees and 52 weeks (and 5 skills). We have tested it on several data sets of that size with varying weights in our objective function. We have compared the run time of our exact model with the run time of the approximate model for different weights. The approximate model gave a relatively quick approximation of the optimal solution when we do not consider skills, and when we do consider skills and vary the weight for the first part of the objective function. For varying the weight on the second part, we used a time limited version of our exact model to approximate the optimal solution. To be able to approximate the optimal solution when varying weight on the third part of the objective function, the approximate model is used with extra weight on the first part, instead of the third.

Preface

This thesis is written to obtain the degree of Bachelor of Science in Applied Mathematics at Delft University of Technology. The project was supervised by Theresia van Essen, who is an assistant professor in the optimization group. The assignment originally came from ORTEC which Egbert van der Veen was the contact person of during the project for answering questions and providing practical insights.

In this project, a multi-objective optimization problem is considered. A nonlinear model is stated, after which a linear exact model is created to find optimal solutions of the problem. Also an approximate model is made, to find approximate solutions for the problem. Their computation time and objectives for multiple instances of a data set are compared. Feel free to contact me about the specific data sets or the outcome of an individual instance.

I would like to take this moment to especially thank Theresia van Essen, for sharing her knowledge, the always constructive meetings we had, her quick responses to any question and her support. In addition, I would like to thank Egbert van der Veen, who welcomed us several times at ORTEC to discuss this research. Also thank you for providing the assignment and the data we used. I would like to thank Wouter Raateland for the mental and technological support in getting used to Python (which I had never used before, but actually started to appreciate pretty much). And last but not least, I would like to thank Johan Dubbeldam and Emiel van Elderen for being part of my thesis committee.

M.E. Bouwmeester
Delft, June 2019

Contents

1 Introduction

“Thought experiment: compare a hospital with Bol.com,” said Taco van der Vaart, Professor of Supply Chain Management at the University of Groningen, recently in *de Volkskrant* [10]. He admits that this is a poor comparison, but makes the clear point that he believes that hospitals can learn a lot from logistics champions like Bol.com. Van der Vaart discusses the three components that, according to him, are the cause of long waiting times in hospitals. The third part starts as follows: “What perhaps intrigues Van der Vaart most, given his supply chain view of the world, is the enormous variability of what a hospital does. In plain Dutch: the work schedules are a mess.”

Fortunately, more and more systems are being introduced to reduce this problem. These are systems that make it easier to respond to the varying demand and absent employees. An example of such a system is the use of annualized hours. In this system, working hours are expressed in working hours per year instead of working hours per week. As a result, within certain bounds, an employee may, for example, work 38 hours in one week, and 42 in the next. The working hours in a contract are also shown in hours per year.

The company ORTEC deals with customers who are struggling with the same issues. ORTEC specializes in developing optimization software with which it is their purpose to improve the world using their passion for mathematics.[1] The assignment for this thesis arose from a request from ORTEC.

In this thesis, we make use of the annualized hours system. Our data consists of a weekly varying demand, a fixed shift duration, and for each employee, the contract hours, minimum and maximum working hours, set of skills and weeks in which the employee is absent. We create a multi-objective optimization model which minimizes the difference between contract hours and working hours for each employee per week, the over and under staffing, and the difference between the contract and working hours over the whole planning period.

An overview of the literature on this topic can be found in Chapter 2. Chapter 3 of this thesis contains the necessary background information, after which our first model is presented in Chapter 4. In this chapter, the original model, a linear version of the exact model and an approximation of that can be found. In Chapter 5, the models get extended. The results of the models are shown and compared in Chapter 6. Finally, the conclusion, discussion and recommendation can be found in Chapter 7.

[1]https://ortec.com/about-us

2

Literature review

In this thesis, we consider the use of annualized hours which might be a great help when the aim is to minimize differences between working hours and contract hours when matching the demand. In this section, we review literature on models that consider annualized hours. To get a clear overview, we look at multiple characteristics of such a model and discuss literature per characteristic.

Definition annualized hours
Before discussing some characteristics, we consider the definition of annualized hours. A formal definition of the term is 'Method of computing working time by year rather than by the week. This method is used sometimes in industries or occupations where there are seasonal variations in demand for services of employees.'[1] This can be translated to our situation; we look for a schedule for a longer planning period, while having a varying demand.

Objective function
One of the characteristics we consider is the focus of the objective function. Corominas et al. [2, 3], Hasan et al. [7] and Van der Veen et al. [11] focus on minimizing the total costs of staffing, while Corominas et al. [2, 3] also focus on minimizing over and under staffing. In our model we focus on the minimization of over and under staffing as well, and next to that our focus is on minimizing the difference between contract and working hours. Note that while the focus seems to be different, the goals could be the same: minimizing the costs induces minimizing overtime, which links to minimizing the difference between contract and working hours. Keim [9] considers the same objective function as the one we consider.
There are researches that use overtime instead of working hours in their objective function, like Azmat et al. [1]. A third way to regulate working hours is by measuring Working Time Accounts (WTAs). In a system with WTAs, an employee is able to work more or less than their contract hours, and thereby, collect working time credits or debits over a certain period in an individual working time account. WTAs are used in e.g. García and Pastor [4], in which the objective is to find a subscription for the hours where the WTA of the employees stays between specified boundaries.

Parameters and constraints
Other characteristics in which an annualized hours model can distinguish itself from other models lay in the way parameters are defined and the various constraints of the given problem.

Fixed flexibility for each employee
In our case, each employee has a specified flexibility. This flexibility is expressed in the fact that the difference in working hours between different weeks for an employee is bounded

[1] www.businessdictionary.com

per week. The aim is that within this flexibility, the working hours over the whole planning period are equal to or as close as possible to the contract hours over the whole period. This characteristic is also used in Van der Veen et al. [11] and Keim [9].

Varying demand
One of the strongest effects of the annualized hours principle is that it helps responding to a varying demand. In our case, the demand is deterministic, just like in Azmat et al. [1] and Hung [8], who use shifts that need to be staffed as their demand, and Van der Veen et al. [11] and Keim [9] whose demand is measured in hours. García et al. [5] consider a stochastic demand by including multiple demand scenarios with a probability of occurrence per case.

Distinction between skills
When people's working hours need to match demand, this demand can be divided over different tasks which require different skills. In this case, every employee gets an extra attribute which represents which skills they have and which they do not. Person X has for example skills 1 and 3, but not skill 2, which makes him qualified to do tasks A and B, but lacks the skill to do task C. In the studies of e.g. Van der Veen et al. [11], Corominas et al. [2] and Grabot and Letouzey [6], the distinction between skills is made.

Usage of shifts
As stated before, some researches express their demand in shifts that need to be filled. In our model, the employees work in shifts of a fixed number of hours, whereas in Keim [9] the shift duration is flexible between minimum and maximum bounds, and constraints are given for the amount of shifts over a week, two weeks, and four weeks.

Computing complexity
While Hung [8] uses special purpose algorithms to solve his annualized hours problem, using mathematical programming is the most common approach. We use it in this thesis, and it is used in e.g. Van der Veen et al. [11], Keim [9] and Azmat et al. [1].
To find an optimal solution to a mathematical program, various solvers can be used. In Keim [9], Gurobi is used which took 6 seconds on average to solve the model with a data set of 100 employees and 52 weeks. Van der Veen et al. [11] use CPLEX. They note that their maximum observed computation time of 10 seconds leads them to not going into detail about this time, since they consider it to be negligible for this kind of analysis and decision making. In García et al. [5] also CPLEX is used, and in this paper finding out whether the model can be solved in short times for instances of a realistic size is one of the two main objectives of their computational experiment. Corominas et al. [2, 3] also use CPLEX, while Hasan et al. [7] shows that it is also possible to use LINGO.

The contribution of this thesis is that, to our knowledge, literature does not consider modeling annualized hours in combination with multi-skill and fixed shift duration, while minimizing over and under staffing and the difference between working hours and contract hours per week and per planning period.

3
Background

In this chapter, the background knowledge required to understand the models in upcoming chapters is discussed.

Mixed Integer Linear Programming (MILP)
To model problems in the field of scheduling, production planning, telecommunication networks or cellular networks a mathematical optimization program is often used. In this section we introduce different forms of integer programming.

Integer programming is a feasibility problem in which the variables are restricted to be integers. Many problems have the addition that the objective function and the constraints are linear, which makes integer programming a bit more specific: integer linear programming (ILP). An ILP in standard form is expressed as:

$$\max \mathbf{c}^T\mathbf{x} \tag{3.1}$$
$$\text{s.t. } \mathbf{A}\mathbf{x} \leq \mathbf{b} \tag{3.2}$$
$$\mathbf{x} \geq \mathbf{0} \tag{3.3}$$
$$\mathbf{x} \in \mathbb{Z}^n \tag{3.4}$$

In this case $\mathbf{c}$ and $\mathbf{b}$ are vectors and $\mathbf{A}$ is a matrix, where all entries are integers.
An example of an ILP problem could look like the following:

$$\max y \tag{3.5}$$
$$\text{s.t. } -x + y \leq 1 \tag{3.6}$$
$$3x + 2y \leq 12 \tag{3.7}$$
$$2x + 3y \leq 12 \tag{3.8}$$
$$x, y \geq 0 \tag{3.9}$$
$$x, y \in \mathbb{Z} \tag{3.10}$$

For this problem, we can plot the feasible solutions, which are made visual in Figure 3.1. The red line represents constraint (3.6), the blue line constraint (3.7) and the orange line represents constraint (3.8). The black and purple lines represent constraint (3.9) and only choosing solutions on the thick lines in the graph makes constraint (3.10) satisfied. So the straight lines in the plot represent the constraints of the problem, while the red dots represent the feasible solutions. There are multiple feasible solutions, and out of all those, we want to obtain the optimal one.

In our case the objective function is simple, $\max y$, so out of this plot we can now easily find our optimum. The optimal solution of this example is $y = 2$, which corresponds to the points $(1, 2)$ and $(2, 2)$. Therefore, in this case, there are two optimal solutions.

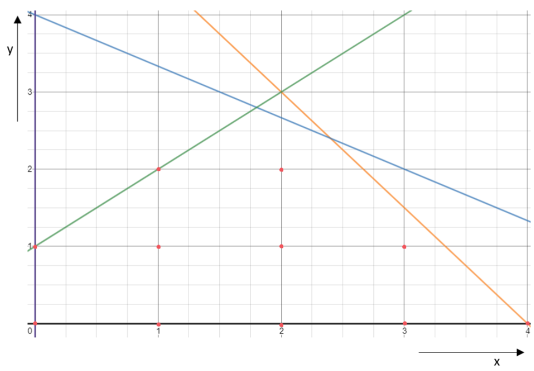

Figure 3.1: Feasible solutions of an ILP example

Mixed Integer Linear Programming (MILP) has an extra characteristic next to the ones of an ILP problem. In a MILP, the variables can be integer as well as non-integer. To be more specific, it involves problems in which only some of the variables, $\mathbf{x_i}$, are constrained to be integers, while others are allowed to be non-integers.

4 Model formulation

We create a Mixed Integer Linear Programming (MILP) model which gives us the number of hours an employee should work each week. That number of hours should represent an outcome in which over and under staffing is minimized, while also minimizing the difference between contract hours and working hours of each employee each week. Next to that, we minimize the difference between contract hours and working hours for each employee over the total planning horizon, for example a year.
We have data that provides us information on how long our planning period is in weeks, the demand of working hours each week, absences of each employee and the minimum, maximum and contract hours of each employee. Next to that, our model should provide a solution which includes that an employee works in shifts of a fixed number of hours.

4.1. Stating the sets, parameters and variables

To formulate a model, we state the following sets, parameters and variables:

Sets	
I	Set of employees
T	Set of time periods (which are weeks)
Parameters	
d_t	Demand (in hours) in week $t \in T$
c_i	Contract hours per week for employee $i \in I$
l_i	Minimum working hours per week for employee $i \in I$
u_i	Maximum working hours per week for employee $i \in I$
$\|T\|$	Length planning horizon in weeks
r	Fixed shift duration
$a_{it} \in \{0,1\}$	0 if employee $i \in I$ is absent in week $t \in T$, 1 otherwise
Variables	
X_{it}	Number of working hours of employee $i \in I$ in week $t \in T$
V_{it}	Number of shifts for employee $i \in I$ in week $t \in T$

The model in this thesis is based on the model formulated by Van der Veen et al. [11] and on the bachelor thesis of Keim [9]. Just like in [9], the significant difference between the model described in [11] and our model is the fact that we do not consider that our employees have different skills. Also, our objective function does not focus on minimizing total costs. The model in [9] and in this thesis are much alike. The biggest difference is that we do not consider a varying duration per shift and a maximum/minimum number of working shifts. What we do take into account is that employees work a fixed amount of hours each shift. This means that when a shift is given to be four hours, an employee with 35 contract hours each week will never get his/her exact contract hours assigned. However, our aim is to get the working hours as close as possible to the contract hours for each employee.

4.2. Initial model

Now we have stated our sets, parameters and variables, we set up our model. Our model consists of a part which is meant to minimize the difference between contract hours and working hours, a part where the over and under staffing is minimized and a part which makes sure that the working hours are as close as possible to the assigned contract hours over the whole planning period. In some cases, it might be interesting to make one part dominant over the other. This is why parameters λ_1, λ_2 and λ_3 are introduced, which determine the importance of each component. A first version of our model is given by:

$$\begin{aligned}\min \Bigg(& \lambda_1 \cdot \sum_{i\in I}\left(\sum_{t\in T}(X_{it}-c_i a_{it})^2\right) \\ & + \lambda_2 \cdot \sum_{t\in T}\left(\sum_{i\in I}X_{it}-d_t\right)^2 \\ & + \lambda_3 \cdot \sum_{i\in I}\left(\sum_{t\in T}(X_{it}+c_i(1-a_{it})) - |T|c_i\right)^2\Bigg) \end{aligned} \tag{4.1}$$

$$\text{s.t. } l_i a_{it} \le X_{it} \le u_i a_{it} \qquad \forall i \in I, \forall t \in T \tag{4.2}$$

$$X_{it} = r \cdot V_{it} \qquad \forall i \in I, \forall t \in T \tag{4.3}$$

$$X_{it}, V_{it} \in \mathbb{N} \qquad \forall i \in I, \forall t \in T \tag{4.4}$$

Our objective function (4.1) represents the main goal of our model, while (4.2), (4.3) and (4.4) represent the constraints. Constraint (4.2) makes sure that the working hours of an employee are between his/her minimum and maximum hours, when that employee is present. To implement that employees work in shifts of a fixed number of hours, constraint (4.3) is introduced. Constraint (4.4) makes sure that the number of working hours X_{it} and the number of shifts an employee works V_{it} are non-negative integers.

4.3. Exact linear model

We aim to make a Mixed Integer Linear Programming (MILP) model. A MILP has as an advantage that there are a lot of solvers available, which can solve the problem relatively quick. To reformulate our initial model (4.1)-(4.4) as a MILP, we need our model, as the name *Mixed Integer Linear Programming* suggests, to be linear. That means we need to lose the squares in our model.

For this reason, we create an alternative model, which makes use of binary variables. This model gives the exact optimal solution to our initial model. To give a clear explanation on how this model is set up, we start with considering only the first part of the objective function.

4.3.1. Reformulating the first part of the initial model

The first part of the objective function of our initial model is the part which minimizes the difference between the working hours and contract hours of each employee per week. The maximum deviation for an employee, b_i, is obtained by subtracting the minimum working hours from the maximum working hours: $b_i = (u_i - l_i), \quad \forall i \in I$. So we know that for each week, this deviation lies between zero and $\max_{i\in I} b_i$ for each employee. We capture these possibilities for deviations in $J = \{0, 1, \ldots, \max_{i\in I} b_i\}$. Furthermore, we define the difference between working hours and contract hours of a certain employee in a certain week to be p_{it}. In other words, $|X_{it} - c_i a_{it}| = p_{it}$. We want our model to be linear, so we need to rewrite the prior such that it does not contain an absolute value. For this, we use the following two constraints:

$$p_{it} \ge X_{it} - c_i a_{it} \qquad \forall i \in I, \forall t \in T \tag{4.5}$$

$$p_{it} \geq c_i a_{it} - X_{it} \qquad \forall i \in I, \forall t \in T \tag{4.6}$$

Minimizing these gives the deviation we are aiming for. We then connect that deviation with a binary variable Y_{ijt}, which indicates whether there is a deviation of $j = p_{it}$ for employee $t \in T$ in week $i \in I$:

$$Y_{ijt} = \begin{cases} 1, & \text{if } j = p_{it}, \text{ where } j, p_{it} \in J \\ 0, & \text{otherwise} \end{cases} \tag{4.7}$$

Summing Y_{ijt} over $j \in J$ should be equal to one, since only one deviation $j \in J$ can be obtained per week for each employee.

$$\sum_{j \in J} Y_{ijt} = 1 \qquad \forall i \in I, \forall t \in T \tag{4.8}$$

With this new way of looking at the problem, and also adding the constraints for the working hours being between the minimum and maximum hours of an employee, and the working hours being a multiple of the fixed shift duration, we get the following model when only considering the first part of the objective function of our initial model:

$$\min \sum_{i \in I} \sum_{t \in T} j^2 \cdot Y_{ijt} \tag{4.9}$$

$$\text{s.t. } l_i a_{it} \leq X_{it} \leq u_i a_{it} \qquad \forall i \in I, \forall t \in T \tag{4.10}$$

$$X_{it} = r \cdot V_{it} \qquad \forall i \in I, \forall t \in T \tag{4.11}$$

$$p_{it} \geq X_{it} - c_i a_{it} \qquad \forall i \in I, \forall t \in T \tag{4.12}$$

$$p_{it} \geq c_i a_{it} - X_{it} \qquad \forall i \in I, \forall t \in T \tag{4.13}$$

$$\sum_{j \in J} j \cdot Y_{ijt} = p_{it} \qquad \forall i \in I, \forall t \in T \tag{4.14}$$

$$\sum_{j \in J} Y_{ijt} = 1 \qquad \forall i \in \forall, t \in T \tag{4.15}$$

$$X_{it}, V_{it} \in \mathbb{N} \qquad \forall i \in I, \forall t \in T \tag{4.16}$$

$$Y_{ijt} \in \{0, 1\} \qquad \forall i \in I, \forall j \in B_i, \forall t \in T \tag{4.17}$$

In this case, our objective function (4.9) contains a value j and a binary variable Y_{ijt}. Constraints (4.10), (4.11) and (4.16) are the same as Constraints (4.2), (4.3) and (4.4) respectively. Constraints (4.12) and (4.13) represent what is stated in Equations (4.5) and (4.6). Constraint (4.14) makes the outcome of the sum over j times the binary variable equal to the deviation p_{it}. And Constraint (4.15) shows what is stated in Equation (4.8). To conclude, Constraint (4.17) gives the binary restriction of Y_{ijt}.

4.3.2. Full exact linear model

The reformulation of our model can be repeated for the second part of the objective function of the initial model, which minimizes the difference between demand and working hours, and the third part which minimizes the difference between contract hours and working hours of an employee over the total planning period. The maximum deviation for the second part is d_t for each week. We capture the possible deviations for this part in the set $H = \{0, 1, \ldots, \max_{t \in T} d_t\}$. For the third part, the maximum deviation is $tc_i = |T| \cdot \max(u_i - c_i, c_i - l_i)$ for each employee. That leads to the set $K = \{0, 1, \ldots, \max_{i \in I} tc_i\}$ for the third part. Now, we add weights λ_1, λ_2 and λ_3 and obtain the following model:

$$\min \lambda_1 \cdot \sum_{i\in I}\sum_{t\in T} j^2 \cdot Y_{ijt} + \lambda_2 \cdot \sum_{t\in T} h^2 \cdot Z_{ht} + \lambda_3 \cdot \sum_{i\in I} k^2 \cdot W_{ik} \tag{4.18}$$

$$\text{s.t. } l_i a_{it} \le X_{it} \le u_i a_{it} \qquad \forall i \in I, \forall t \in T \tag{4.19}$$

$$X_{it} = r \cdot V_{it} \qquad \forall i \in I, \forall t \in T \tag{4.20}$$

$$p_{it} \ge X_{it} - c_i a_{it} \qquad \forall i \in I, \forall t \in T \tag{4.21}$$

$$p_{it} \ge c_i a_{it} - X_{it} \qquad \forall i \in I, \forall t \in T \tag{4.22}$$

$$\sum_{j\in J} j \cdot Y_{ijt} = p_{it} \qquad \forall i \in \forall, t \in T \tag{4.23}$$

$$\sum_{j\in J} Y_{ijt} = 1 \qquad \forall i \in \forall, t \in T \tag{4.24}$$

$$q_t \ge \sum_{i\in I} X_{it} - d_t \qquad \forall t \in T \tag{4.25}$$

$$q_t \ge d_t - \sum_{i\in I} X_{it} \qquad \forall t \in T \tag{4.26}$$

$$\sum_{h\in H} h \cdot Z_{ht} = q_t \qquad \forall t \in T \tag{4.27}$$

$$\sum_{h\in H} Z_{ht} = 1 \qquad \forall t \in T \tag{4.28}$$

$$\tau_i \ge \Big(\sum_{t\in T} X_{it} + c_i(1 - a_{it})\Big) - |T|c_i \qquad \forall i \in I \tag{4.29}$$

$$\tau_i \ge |T|c_i - \Big(\sum_{t\in T} X_{it} + c_i(1 - a_{it})\Big) \qquad \forall i \in I \tag{4.30}$$

$$\sum_{k\in K} k \cdot W_{ik} = \tau_i \qquad \forall i \in I \tag{4.31}$$

$$\sum_{k\in K} W_{ik} = 1 \qquad \forall i \in I \tag{4.32}$$

$$X_{it}, V_{it} \in \mathbb{N} \qquad \forall i \in I, \forall t \in T \tag{4.33}$$

$$Y_{ijt}, Z_{ht}, W_{ik} \in \{0,1\} \qquad \forall h \in H, \forall i \in I, \forall j \in J, \quad \forall k \in K, \forall t \in T \tag{4.34}$$

4.4. Approximate linear model

Next to the reformulation stated in Section 4.3, we could think of other ways to make our initial model linear. In this section, we make a linear model which approximates the solution of our problem.

To reformulate the squares in the objective function of the initial model stated in Section 4.2, we minimize the maximum absolute value of the difference between contract and working hours, the maximum of over and under staffing, and the maximum absolute value of the difference between working hours and contract hours over the entire planning horizon. The objective function turns into the following:

$$\min \lambda_1 \cdot \sum_{i\in I} \max_{t\in T}(|X_{it} - c_i \cdot a_{it}|) \; + \; \lambda_2 \cdot \max_{t\in T}\left(\Big|\sum_{i\in I} X_{it} - d_t\Big|\right) + \lambda_3 \cdot \max_{i\in I}\left(\Big|\sum_{t\in T}(X_{it} + c_i \cdot (1 - a_{it})) - |T| \cdot c_i\Big|\right) \tag{4.35}$$

Our objective function has now lost its quadratic formulation. However, the new maximum functions and absolute values lead still to a non-linear model.
To lose those terms, we reformulate the objective function by using substitute values m_i, ν and σ and the following constraints:

$$m_i \geq X_{it} - c_i a_{it} \qquad \forall i \in I, \forall t \in T \tag{4.36}$$

$$m_i \geq c_i a_{it} - X_{it} \qquad \forall i \in I, \forall t \in T \tag{4.37}$$

$$\nu \geq \sum_{i \in I} X_{it} - d_t \qquad \forall t \in T \tag{4.38}$$

$$\nu \geq d_t - \sum_{i \in I} X_{it} \qquad \forall t \in T \tag{4.39}$$

$$\sigma \geq \sum_{t \in T} (X_{it} + c_i(1 - a_{it})) - |T| c_i \qquad \forall i \in I \tag{4.40}$$

$$\sigma \geq |T| c_i - \sum_{t \in T} (X_{it} + c_i(1 - a_{it})) \qquad \forall i \in I \tag{4.41}$$

Constraints (4.36) and (4.37) make minimizing the maximum difference between contract hours and working hours possible for each employee. Constraint (4.38) and (4.39) are used for minimizing the maximum difference between demand and working hours. Minimizing the maximum difference between total working hours and contract hours is possible because of constraints (4.40) and (4.41). This makes our objective function look like the following:

$$\min \ \lambda_1 \cdot \sum_{i \in I} m_i \ + \ \lambda_2 \cdot \nu + \ \lambda_3 \cdot \sigma \tag{4.42}$$

Now we have these linear parts of the objective function, we are not finished yet. The reason for this is that if we remove the squares from the objective function, we lose grip on the values below the maximum. A quadratic function has the advantage that it distributes the deviations of the differences as well as possible. Through the squares, an option in which two employees deviate by one hour is chosen over an option in which one employee deviates by two hours, and the other by zero. In an attempt to partially copy that characteristic of the squares, we add more substitute values. We add p_{it} with Constraints (4.43) and (4.44) to minimize the total difference between contract and working hours for each week and each employee:

$$p_{it} \geq X_{it} - c_i a_{it} \qquad \forall i \in I, \forall t \in T \tag{4.43}$$

$$p_{it} \geq c_i a_{it} - X_{it} \qquad \forall i \in I, \forall t \in T \tag{4.44}$$

We repeat this for the second and third part of our objective function. We add a part where the weekly over and under staffing is minimized by adding q_t with Constraints (4.45) and (4.46).

$$q_t \geq \sum_{i \in I} X_{it} - d_t \qquad \forall t \in T \tag{4.45}$$

$$q_t \geq d_t - \sum_{i \in I} X_{it} \qquad \forall t \in T \tag{4.46}$$

To obtain the minimum total difference between the total working hours and contract hours for each employee, τ_i is introduced with constraints (4.47) and (4.48).

$$\tau_i \geq \sum_{t \in T} (X_{it} + c_i(1 - a_{it})) - |T| c_i \qquad \forall i \in I \tag{4.47}$$

$$\tau_i \geq |T| c_i - \sum_{t \in T} (X_{it} + c_i(1 - a_{it})) \qquad \forall i \in I \tag{4.48}$$

When adding all these to the objective function, we need to give each term weights. Instead of λ_1-λ_3, we know use κ_1-κ_6 which do have the same ratio. Adding these and combining the introduced variables and constraints, gives the following full model:

$$\min \kappa_1 \cdot \sum_{i\in I} m_i \; + \; \kappa_2 \cdot \nu + \kappa_3 \cdot \sigma + \kappa_4 \cdot \sum_{i\in I}\sum_{t\in T} p_{it} + \kappa_5 \cdot \sum_{t\in T} q_t + \kappa_6 \cdot \sum_{i\in I} \tau_i \tag{4.49}$$

$$\text{s.t. } l_i \cdot a_{it} \le X_{it} \le u_i \cdot a_{it} \qquad \forall i \in I, \forall t \in T \tag{4.50}$$

$$X_{it} = r \cdot V_{it} \qquad \forall i \in I, \forall t \in T \tag{4.51}$$

$$m_i \ge X_{it} - c_i \cdot a_{it} \qquad \forall i \in I, \forall t \in T \tag{4.52}$$

$$m_i \ge c_i \cdot a_{it} - X_{it} \qquad \forall i \in I, \forall t \in T \tag{4.53}$$

$$\nu \ge \sum_{i\in I} X_{it} - d_t \qquad \forall t \in T \tag{4.54}$$

$$\nu \ge d_t - \sum_{i\in I} X_{it} \qquad \forall t \in T \tag{4.55}$$

$$\sigma \ge \sum_{t\in T} \left(X_{it} + c_i \cdot (1 - a_{it})\right) - |T| \cdot c_i \qquad \forall i \in I \tag{4.56}$$

$$\sigma \ge |T| \cdot c_i - \sum_{t\in T} \left(X_{it} + c_i \cdot (1 - a_{it})\right) \qquad \forall i \in I \tag{4.57}$$

$$p_{it} \ge X_{it} - c_i \cdot a_{it} \qquad \forall i \in I, \forall t \in T \tag{4.58}$$

$$p_{it} \ge c_i \cdot a_{it} - X_{it} \qquad \forall i \in I, \forall t \in T \tag{4.59}$$

$$q_t \ge \sum_{i\in I} X_{it} - d_t \qquad \forall t \in T \tag{4.60}$$

$$q_t \ge d_t - \sum_{i\in I} X_{it} \qquad \forall t \in T \tag{4.61}$$

$$\tau_i \ge \sum_{t\in T} \left(X_{it} + c_i \cdot (1 - a_{it})\right) - |T| \cdot c_i \qquad \forall i \in I \tag{4.62}$$

$$\tau_i \ge |T| \cdot c_i - \sum_{t\in T} \left(X_{it} + c_i \cdot (1 - a_{it})\right) \qquad \forall i \in I \tag{4.63}$$

$$X_{it}, V_{it} \in \mathbb{N} \qquad \forall i \in I, \forall t \in T \tag{4.64}$$

5

Model extension

Now we have formulated our model, we make it more interesting to consider the computation time by also considering skills. We start with extending our initial model.

Until this point we assumed that all of the demand needed one skill and that all employees possessed that skill. We now add the distinction between skills to our model. We define S to be the set of skills, and $S_i \subset S$ the set of skills employee i possesses. When extending our model with the presence of skills, the working hours variable gets an additional index: X_{its}. That variable represents the hours employee $i \in I$ works in week $t \in T$ with skill $s \in S_i$. So when we want to know the working hours of a certain employee in a certain week, we use the summation over s, $\sum_{s \in S_i} X_{its}$. Also our demand d_{ts} gets an additional index, because the demand is now specified in hours per week per skill.

5.1. Stating the sets, parameters and variables

With this, the sets, parameters and variables in Section 4.1 get some additions. All together we state the following:

Sets	
I	Set of employees
T	Set of time periods (which are weeks)
S	Set of skills
$S_i \subset S$	Set of skills of employee $i \in I$
Parameters	
d_{ts}	Demand (in hours) in week $t \in T$ for skill $s \in S$
c_i	Contract hours per week for employee $i \in I$
l_i	Minimum working hours per week for employee $i \in I$
u_i	Maximum working hours per week for employee $i \in I$
$\|T\|$	Length planning horizon in weeks
r	Duration of one shift in hours
$a_{it} \in \{0, 1\}$	0 if employee $i \in I$ is absent in week $t \in T$, 1 otherwise
Variables	
X_{its}	Number of working hours of employee $i \in I$ in week $t \in T$ on skill $s \in S$
V_{it}	Number of shifts for employee $i \in I$ in week $t \in T$

While some of the input has changed, note that a lot stays the same as in Section 4.1.

5.2. The extended initial model

Now we have these sets, parameters and variables, we can extend our initial model with the distinction between skills. This gives the following model:

$$\min \lambda_1 \cdot \sum_{i\in I}\left(\sum_{t\in T}\left(\sum_{s\in S_i} X_{its} - c_i a_{it}\right)^2\right) + \lambda_2 \cdot \sum_{t\in T}\sum_{s\in S}\left(\sum_{i\in I} X_{its} - d_{ts}\right)^2$$

$$+ \lambda_3 \cdot \sum_{i\in I}\left(\sum_{t\in T}\left(\sum_{s\in S_i} X_{its} + c_i(1-a_{it})\right) - |T|c_i\right)^2 \tag{5.1}$$

$$\text{s.t. } l_i a_{it} \leq \sum_{s\in S_i} X_{its} \leq u_i a_{it} \qquad \forall i \in I, \forall t \in T \tag{5.2}$$

$$X_{its} = 0 \qquad \forall i \in I, \forall t \in T, \forall s \notin S_i \tag{5.3}$$

$$\sum_{s\in S_i} X_{its} = r \cdot V_{it} \qquad \forall i \in I, \forall t \in T \tag{5.4}$$

$$X_{its}, V_{it} \in \mathbb{N} \qquad \forall i \in I, \forall t \in T \tag{5.5}$$

Objective function (5.1) represents the main goal of our model with the addition of skills. Again, the first part minimizes the difference between contract hours and working hours for each employee each week, the second part minimizes over and under staffing and the third part makes sure that the working hours over the whole planning period are as close to the total assigned contract hours as possible.

Constraint (5.2) makes sure that the working hours of an employee, which is the sum of the working hours over all possessed skills, are between his/her minimum and maximum hours, each week the employee is present. Constraint (5.3) is needed to make sure that an employee does not get hours assigned to work on a skill that the employee does not have; if an employee does not have a certain skill, that employee cannot work on a task that requires that skill. To implement that employees work in shifts of a fixed number of hours, constraint (5.4) is introduced. Note that an employee can work on multiple skills during one shift. That is the reason for requiring the sum of the working hours over all skills to be a multiple of the chosen shift duration r, and not X_{its} itself. Constraint (5.5) makes sure that the number of working hours X_{its} and the number of shifts an employee works V_{it} are non-negative integers.

5.3. The extended exact model

Now we extend the exact model from Section 4.3. Like before, we use sets for the possible deviations per part of the objective function. We use $J = \{0, 1, \dots, \max_{i\in I}(u_i - l_i)\}$ for the first part, and $K = \{0, 1, \dots, \max_{i\in I}(|T|\cdot\max(u_i - c_i, c_i - l_i))\}$ for the third part of the objective function. For the second part, we need to take into account that the demand is now divided in demand per skill per week. This leads us to using $H = \{0, 1, \dots, (\max_{t\in T}(\max_{s\in S} d_{ts}))\}$ for the second part. We get the following extended model:

$$\min \lambda_1 \cdot \sum_{i\in I}\sum_{t\in T} j^2 \cdot Y_{ijt} + \lambda_2 \cdot \sum_{t\in T}\sum_{s\in S} h^2 \cdot Z_{hts} + \lambda_3 \cdot \sum_{i\in I} k^2 \cdot W_{ik} \tag{5.6}$$

$$\text{s.t. } l_i a_{it} \leq \sum_{s\in S_i} X_{its} \leq u_i a_{it} \qquad \forall i \in I, \forall t \in T \tag{5.7}$$

$$X_{its} = 0 \qquad \forall i \in I, \forall t \in T, \forall s \notin S_i \tag{5.8}$$

$$\sum_{s\in S_i} X_{its} = r \cdot V_{it} \qquad \forall i \in I, \forall t \in T \tag{5.9}$$

$$p_{it} \geq \sum_{s\in S_i} X_{its} - c_i a_{it} \qquad \forall i \in I, \forall t \in T \tag{5.10}$$

$$p_{it} \geq c_i a_{it} - \sum_{s\in S_i} X_{its} \qquad \forall i \in I, \forall t \in T \tag{5.11}$$

$$\sum_{j\in J} j \cdot Y_{ijt} = p_{it} \qquad \forall i \in I, \forall t \in T \tag{5.12}$$

$$\sum_{j\in J} Y_{ijt} = 1 \qquad \forall i \in I, \forall t \in T \tag{5.13}$$

$$\rho_{ts} \geq \sum_{i\in I} X_{its} - d_{ts} \qquad \forall t \in T, \forall s \in S \tag{5.14}$$

$$\rho_{ts} \geq d_{ts} - \sum_{i\in I} X_{its} \qquad \forall t \in T, \forall s \in S \tag{5.15}$$

$$\sum_{h\in H} h \cdot Z_{hts} = \rho_{ts} \qquad \forall t \in T, \forall s \in S \tag{5.16}$$

$$\sum_{h\in H} Z_{hts} = 1 \qquad \forall t \in T, \forall s \in S \tag{5.17}$$

$$\tau_i \geq \sum_{t\in T}\left(\sum_{s\in S_i} X_{its} + c_i(1 - a_{it})\right) - |T|c_i \qquad \forall i \in I \tag{5.18}$$

$$\tau_i \geq |T|c_i - \sum_{t\in T}\left(\sum_{s\in S_i} X_{its} + c_i(1 - a_{it})\right) \qquad \forall i \in I \tag{5.19}$$

$$\sum_{k\in K} k \cdot W_{ik} = \tau_i \qquad \forall i \in I \tag{5.20}$$

$$\sum_{k\in K} W_{ik} = 1 \qquad \forall i \in I \tag{5.21}$$

$$X_{its}, V_{it} \in \mathbb{N} \qquad \forall i \in I, \forall t \in T, \forall s \in S \tag{5.22}$$

$$Y_{ijt}, Z_{hts}, W_{ik} \in \{0, 1\} \qquad \forall h \in H, \forall i \in I, \forall j \in J, \quad \forall k \in K, \forall t \in T, \forall s \in S \tag{5.23}$$

5.4. The extended approximate linear model

Like in the initial model, we add the distinction between skills in the approximate linear model.

This for example makes constraints (4.52) and (4.53) turn into:

$$m_i \geq \sum_{s \in S_i} X_{its} - c_i a_{it} \qquad \forall i \in I, \forall t \in T \tag{5.24}$$

$$m_i \geq c_i a_{it} - \sum_{s \in S_i} X_{its} \qquad \forall i \in I, \forall t \in T \tag{5.25}$$

Constraints (4.54) and (4.55) stay the same (while this part of the objective function changes), apart from the new indexes:

$$\nu \geq \sum_{i \in I} X_{its} - d_{ts} \qquad \forall t \in T, \forall s \in S \tag{5.26}$$

$$\nu \geq d_{ts} - \sum_{i \in I} X_{its} \qquad \forall t \in T, \forall s \in S \tag{5.27}$$

And instead of constraints (4.56) and (4.57), we get:

$$\sigma \geq \sum_{t \in T} \left(\sum_{s \in S_i} X_{its} + c_i(1 - a_{it}) \right) - |T| c_i \qquad \forall i \in I \tag{5.28}$$

$$\sigma \geq |T| c_i - \sum_{t \in T} \left(\sum_{s \in S_i} X_{it} + c_i(1 - a_{it}) \right) \qquad \forall i \in I \tag{5.29}$$

The whole model gets extended like that and leads to the following model:

$$\min \kappa_1 \cdot \sum_{i \in I} m_i \ + \ \kappa_2 \cdot \nu + \kappa_3 \cdot \sigma + \kappa_4 \cdot \sum_{i \in I} \sum_{t \in T} p_{it}$$

$$+ \kappa_5 \cdot \sum_{t \in T} q_t + \kappa_6 \cdot \sum_{t \in T} \sum_{s \in S_i} \rho_{ts} + \kappa_7 \cdot \sum_{i \in I} \tau_i \tag{5.30}$$

$$\text{s.t. } l_i a_{it} \leq \sum_{s \in S_i} X_{its} \leq u_i a_{it} \qquad \forall i \in I, \forall t \in T \tag{5.31}$$

$$X_{its} = 0 \qquad \forall i \in I, \forall t \in T, \forall s \notin S_i \tag{5.32}$$

$$\sum_{s \in S_i} X_{its} = r \cdot V_{it} \qquad \forall i \in I, \forall t \in T \tag{5.33}$$

$$m_i \geq \sum_{s \in S_i} X_{its} - c_i a_{it} \qquad \forall i \in I, \forall t \in T \tag{5.34}$$

$$m_i \geq c_i a_{it} - \sum_{s \in S_i} X_{its} \qquad \forall i \in I, \forall t \in T \tag{5.35}$$

$$\nu \geq \sum_{i \in I} X_{its} - d_{ts} \qquad \forall t \in T, \forall s \in S \tag{5.36}$$

$$\nu \geq d_{ts} - \sum_{i \in I} X_{its} \qquad \forall t \in T, \forall s \in S \tag{5.37}$$

$$\sigma \geq \sum_{t \in T} \left(\sum_{s \in S_i} X_{its} + c_i(1 - a_{it}) \right) - |T| c_i \qquad \forall i \in I \tag{5.38}$$

$$\sigma \geq |T|c_i - \sum_{t \in T} \left(\sum_{s \in S_i} X_{its} + c_i(1 - a_{it}) \right) \quad \forall i \in I \quad (5.39)$$

$$p_{it} \geq \sum_{s \in S_i} X_{its} - c_i a_{it} \quad \forall i \in I, \forall t \in T \quad (5.40)$$

$$p_{it} \geq c_i a_{it} - \sum_{s \in S_i} X_{its} \quad \forall i \in I, \forall t \in T \quad (5.41)$$

$$q_t \geq \sum_{i \in I} X_{its} - d_{ts} \quad \forall t \in T, \forall s \in S \quad (5.42)$$

$$q_t \geq d_{ts} - \sum_{i \in I} X_{its} \quad \forall t \in T, \forall s \in S \quad (5.43)$$

$$\rho_{ts} \geq \sum_{i \in I} X_{its} - d_{ts} \quad \forall t \in T, \forall s \in S \quad (5.44)$$

$$\rho_{ts} \geq d_{ts} - \sum_{i \in I} X_{its} \quad \forall t \in T, \forall s \in S \quad (5.45)$$

$$\tau_i \geq \sum_{t \in T} \left(\sum_{s \in S_i} X_{its} + c_i(1 - a_{it}) \right) - |T|c_i \quad \forall i \in I \quad (5.46)$$

$$\tau_i \geq |T|c_i - \sum_{t \in T} \left(\sum_{s \in S_i} X_{its} + c_i(1 - a_{it}) \right) \quad \forall i \in I \quad (5.47)$$

$$X_{its}, V_{it} \in \mathbb{N} \quad \forall i \in I, \forall t \in T, \forall s \in S \quad (5.48)$$

6 Results MILP models

In this chapter, we discuss the results from our previously introduced models.

6.1. Data and parameters

To test our model we need data. This data is obtained from a random instance generator provided by ORTEC. As input, this generator needs the number of employees, weeks, skills and instances we would like it to create. Then, a data set is put together with the following data:

- For each employee is given whether or not the employee possesses a certain skill;
- For each employee, his/her minimum working hours per week, l_i, is given;
- For each employee, his/her contract working hours per week, c_i, is given;
- For each employee, his/her maximum working hours per week, u_i, is given;
- For each employee, the weeks are given in which the employee is absent, this gets translated to a_{it};
- For each week, the demand is given in hours per week (per skill), d_t ($/d_{ts}$)

In the part where the skills are not introduced yet, we only consider one skill, which all employees are qualified for. The demand is then given by d_t instead of d_{ts}.

Next to this data, we choose a value for parameter r, which defines the shift duration. In the tests in this thesis, we defined r to be equal to four hours.

Further, we need to make a choice for our weights. In our initial model and the exact linear model, λ_1, λ_2 and λ_3 are used. Because the three parts of our objective function vary considerably in order of magnitude, we have to adapt our weights to make the parts weigh equally. For doing this, we make use of sets J, H and K introduced in Section 4.3 With their maximum values equal to $\max_{i\in I} b_i = \max_{i\in I}(u_i - l_i)$, $\max_{t\in T} d_t$ / $\max_{t\in T}(\max_{s\in S} d_{ts})$ and $\max_{i\in I} tc_i = \max_{i\in I}(|T| \cdot \max(u_i - c_i, c_i - l_i))$, respectively. We use the halved maximum values to obtain values for the λ_1, λ_2 and λ_3 and standardize the outcome to obtain our weights.

$$\lambda_1^* = \frac{1}{\frac{1}{2}\max_{i\in I} b_i} \qquad \lambda_2^* = \frac{1}{\frac{1}{2}\max_{t\in T} d_t} \qquad \lambda_3^* = \frac{1}{\frac{1}{2}\max_{i\in I} tc_i} \tag{6.1}$$

$$\lambda_{tot}^* = \lambda_1^* + \lambda_2^* + \lambda_3^* \tag{6.2}$$

$$\lambda_1 = \lambda_1^*/\lambda_{tot}^* \qquad \lambda_2 = \lambda_2^*/\lambda_{tot}^* \qquad \lambda_3 = \lambda_3^*/\lambda_{tot}^* \tag{6.3}$$

In the case we consider different skills, λ_2^* changes to $\frac{1}{\frac{1}{2}\max_{t\in T}(\max_{s\in S} d_{ts})}$.

When we would like to make one part of the objective function dominant over the others, we multiply the λ^* of that part by a chosen factor.

We aim to create a model for a planning horizon of 52 weeks and 100 employees, and next to this, in the second time formulating the models, 5 skills. We run our exact linear model and approximate linear model on this data set with varying weights for the three parts of the objective function.

6.2. Results of the models that do not consider skills

First, we do not consider skills. In practice, this means that we consider one skill, which all employees possess. So each employee is able to do each task.

We have set up the two models described in Chapter 4. We implemented them in Python 3.7.2 with the use of PuLP and solved them to optimality with the solver Gurobi (version 8.1).

We have solved the model for 20 instances with 100 employees and 52 weeks. When measuring the run time of the two models, the exact model takes 70.89 seconds on average, while the approximate model is done in 1.83 seconds on average. The run time and objective value for each individual instance and for each model is given in Table 6.1.

	Run time (s)		Objective value		Percentage
Instance	Exact	Approximate	Exact	Approximate	(appr. obj. / exact obj.)
1	49.85	1.39	101,313.0	122,159.8	120.58%
2	53.30	2.41	165,098.0	195,330.1	118.31%
3	168.49	1.22	155,166.0	204,128.8	131.56%
4	47.97	2.68	204,134.0	255,559.5	125.19%
5	76.08	8.46	141,671.0	168,587.2	119.00%
6	65.46	2.09	100,439.0	124,227.1	123.68%
7	79.64	1.26	112,051.0	127,419.0	113.72%
8	55.59	0.86	125,761.0	157,401.6	125.16%
9	68.92	1.47	119,912.0	144,608.3	120.60%
10	28.45	1.24	148,255.0	182,308.5	122.97%
11	73.13	1.11	115,909.0	140,480.6	121.20%
12	46.54	0.95	106,608.0	125,117.2	117.36%
13	62.46	1.11	139,465.0	169,308.6	121.40%
14	41.68	3.31	98,095.0	114,775.2	117.00%
15	61.96	1.21	75,774.7	88,408.7	116.67%
16	64.44	0.98	122,049.0	139,151.2	114.01%
17	60.18	1.35	146,536.0	174,152.0	118.85%
18	68.09	1.87	106,074.0	129,921.2	122.48%
19	125.46	0.30	149,692.0	190,552.1	127.30%
20	120.12	1.36	110,046.0	133,287.3	121.12%
Average	70.89	1.83			120.91%

Table 6.1: Run time (in seconds) and objective values of 20 instances with 100 employees and 52 weeks, the last column represents the percentage that the approximate objective value is of the exact objective value.

From this table, we observe that the run time of the exact model is, in every instance, higher than the run time of the approximate model. Next to this, we notice that the objective value of the approximate model deviates 20.91% from the objective value of the exact model on average. The minimum deviation is 13.72% and the maximum deviation is 31.56%. Note that the two largest deviations in objective values correspond with the two largest deviations in run time when considering the percentages. This is not a clear correlation, since it does not hold for all instances.

6.3. Results of the models that consider skills

We decided to add skills to our model, which is presented in Chapter 5. This introduces additional variables and constraints, which make the run time of our model longer. Again we tested both models on 20 instances with 100 employees, 52 weeks, and now 5 skills. When comparing the run time of the exact model with the approximate model, we use different weights. For this, we use our weights like stated in Equation 6.3 and let one of the weights weigh a chosen factor more than the other two.

In Figure 6.1, the average run times over 20 instances with 100 employees, 52 weeks and 5 skills, for varying weights for the part in which the difference between working hours and contract hours per employee per week is minimized, are presented for the exact and the approximate model. What strikes in this figure is that for the weight of the first part of the objective function weighing as much as the other two parts (when the weight factor is 1), the run time of the exact model is way higher than the other weight factors, and than the approximate model. This might be because it takes time to find a correct balance between the parts. A second remarkable result in this graph is the average run time of the exact model being lower than the run time of the approximate model when the weight is multiplied with a factor ten. For this weight factor, we look at corresponding Table 6.2, which shows the run time and objective value of each instance separately. In this table, we observe that the value for the first part of the objective function - the value we emphasize to minimize in this case - is the same for the exact model and the approximate one. However, the objective value for the exact model is better each instance. We expect this must have to do with the exact model being more appropriate to minimize the parts with a relatively lower weight, than the approximate model is.

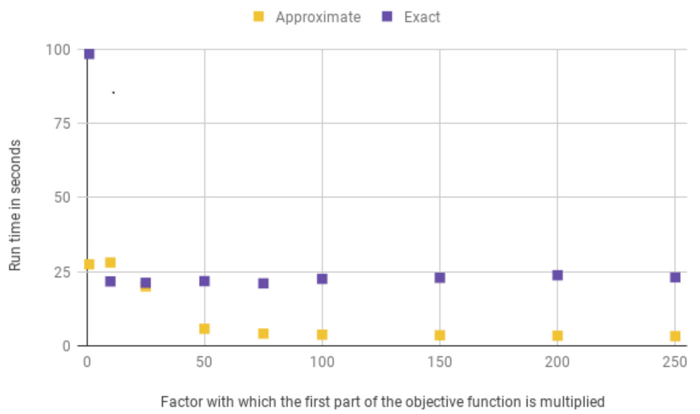

Figure 6.1: Average run time over 20 instances with 100 employees, 52 weeks, 5 skills and a varying factor with which we multiply the first part of the objective function.

After analyzing the effect of varying the weight of the first part of the objective function, we started varying the weight of the second part. The second part of the objective function is the part which is meant to minimize the over and under staffing. We first multiplied the weight with 10, giving us 7.69 seconds as average run time for 20 instances solved with the approximate model, and 189.33 seconds for the 20 instances using the exact model. When multiplying the weight with 25, we got 26.11 seconds on average for the approximate model, and 145.15 for the exact model. When we multiplied the weight with 50, the run time started to vary a lot more per instance, especially for the approximate model. In 16 out of 20 instances, the time needed by the approximate model was longer than the time needed by the exact model. From these results, which are summarized in Table 6.3, we conclude that the approximate model is not suited for getting an approximation of the solution of the objective value of the initial model when the weight for the second part of the objective function is multiplied by 50, 75 or 100. We stopped the computation of the approximate model when it exceeded the run time of the exact model. Therefore it is not possible to compute a representative average run time for the approximate model for some of the weights. In the fourth column we show how many instances exceed the run time of the exact model.

Instead of using the approximate model, we might be able to use the exact model with a time limit. Because the time limit causes that in most instances the optimal value cannot be calculated in this fixed amount of time, this time limited model gives an approximation of

Instance	Run time (s)		Objective value		Value	first part
	Exact	Appr.	Exact	Appr.	Exact	Appr.
1	17.869	23.781	3687.820	3704.618	1316	1316
2	30.134	17.715	4367.283	4391.545	2020	2020
3	12.992	19.681	6515.549	6535.934	196	196
4	38.953	4.398	12065.025	12279.201	1516	1516
5	24.287	5.967	2781.902	2787.609	972	972
6	34.071	56.080	10467.056	10672.391	1308	1308
7	20.225	58.411	3326.457	3339.188	572	572
8	15.851	41.692	4836.339	4890.565	596	596
9	33.281	28.935	6531.890	6597.810	756	756
10	22.024	17.364	2701.313	2711.530	924	924
11	19.379	9.337	3694.619	3694.990	592	592
12	14.733	22.492	4200.169	4250.338	748	748
13	23.948	7.067	6133.590	6169.483	732	732
14	25.202	6.923	4492.524	4514.941	1468	1468
15	14.223	68.018	3523.074	3559.683	576	576
16	27.883	21.944	2896.262	2921.558	552	552
17	10.177	58.686	3768.307	3779.896	780	780
18	17.345	71.541	2651.304	2667.317	560	560
19	11.557	15.057	4081.422	4090.993	168	168
20	21.019	6.947	3862.665	3869.702	756	756
Average	21.758	28.102				

Table 6.2: Run time, objective values and the value for part one of the initial model, for the models for the 20 separate instances with 100 employees, 52 weeks, 5 skills and the weight of the first part of the objective function being a factor 10 bigger than the other two.

Weight	Run time (s)		Times exceeding
	Exact	Approximate	
1	98.471	23.656	0
10	189.326	7.673	0
25	145.145	26.026	0
50	231.774	-	16
75	346.847	-	17
100	309.323	-	15

Table 6.3: Average run time over 20 instances with 100 employees, 52 weeks, 5 skills and a varying factor with which we multiply the second part of the objective function.

the optimal objective value for the initial model, which we then prefer using over the approximate model. We consider time limits of 60 seconds and 120 seconds. The objective values obtained by this are presented in Table 6.4. In this table, the objective values that deviate at most 5% from the objective value of the exact model without time limit are marked green.

For the third part, multiplying the weight by 10 made us obtain average run times of 140.32 seconds and 29.88 seconds for the exact and approximate model, respectively. Multiplying the weight by 25 resulted in average run times of 193.19 seconds for the exact model, but this weight already gave a problem for the approximate model and made the model exceed the run time of the exact model for 14 out of 20 instances. This time we do not use a time limit on our exact model to find another approximation. Namely, there is a correlation between this third part of the objective function and the first part, for which we do have a working approximate model. This correlation is visible in Table 6.5. For the approximate model with extra weight on the first part of the objective function, the value for the third part is the same as or even lower than the exact model with extra weight on the third part. In the table it is clear that the size of weights do not correspond well; we would expect the exact model to give a lower objective value.

Instance	Run time (s) Exact	Objective value Exact	Objective value Exact - TL=60	Objective value Exact TL=120
1	121.677	218268.3	1843910.8	218279.4
2	175.290	267655.2	269092.5	267712.4
3	196.674	544572.4	1263146.0	544629.5
4	16.652	1170432.6	1170432.6	1170432.6
5	202.297	103844.7	104140.3	103865.7
6	58.787	830323.2	83023.2	830323.2
7	222.923	293299.2	838200.1	838200.1
8	220.317	414468.6	416551.4	414537.5
9	49.610	529450.1	529450.1	529450.1
10	276.101	148667.8	10161070.4	10161070.4
11	234.483	311084.6	3950796.5	2131371.1
12	84.194	396956.8	943800.2	396956.8
13	119.341	435610.4	435694.5	435610.4
14	183.023	362195.4	2043692.7	362197.5
15	97.180	293020.0	293057.0	293020.0
16	94.416	232703.5	232854.0	232703.5
17	674.826	339401.1	36791808.4	1109759.3
18	253.698	122572.0	3945076.1	2033049.6
19	230.733	376477.1	1808776.8	376518.8
20	176.662	297209.8	2185658.5	297248.3

Table 6.4: Objective values for 20 separate instances with 100 employees, 52 weeks, 5 skills and the weight of the second part of the objective function being a factor 50 bigger than the other two, for the exact model (for which also run time in seconds is given) and the exact model with a time limit (TL) on the run time of 60 and 120 seconds.

We expect that putting a higher weight on the third part of the objective function of the exact model will give us values for the third part as good as the approximate model with extra weight on the first part. We chose to multiply the weight of the third part by 100, for which the results are shown in Table 6.6. In this case we did indeed get the same values for the third part of the objective function as the approximate model. However, what is remarkable is that the overall objective value is higher than of the approximate model. We expect this having to do with the weights of the exact and approximate model not corresponding one on one, and the weight on the first part of the objective model not corresponding to the weight of the third part.

Instance	Run time (s) Exact - w3=10	Appr. w1=10	Objective value Exact - w3=10	Appr. w1=10	Value third part Exact - w3=10	Appr. w1=10
1	119.819	23.781	18829.1	3704.6	12	12
2	238.553	17.715	19244.8	4391.5	636	12
3	119.904	19.681	45080.6	6535.9	996	4
4	127.995	4.398	80088.6	12279.2	3660	12
5	170.032	5.967	14292.3	2787.6	4	4
6	157.842	56.080	69433.5	10672.4	4012	12
7	209.701	58.411	22232.5	3339.2	508	12
8	75.972	41.692	32168.9	4890.6	1028	4
9	213.515	28.935	43921.6	6597.8	1036	12
10	65.641	17.364	13983.6	2711.5	12	12
11	107.352	9.337	23583.4	3695.0	8	8
12	153.604	22.492	28030.5	4250.3	84	4
13	136.633	7.067	40467.4	6169.5	972	12
14	95.009	6.923	23617.6	4514.9	20	20
15	59.720	68.018	21533.9	3559.7	24	8
16	126.767	21.944	19029.9	2921.6	40	8
17	138.108	58.686	22496.0	3779.9	916	4
18	118.564	71.541	16131.2	2667.3	8	8
19	130.124	15.057	29269.4	4091.0	1056	0
20	241.463	6.947	23901.0	3869.7	956	12

Table 6.5: Run time in seconds, objective value and value of the third part of the initial objective function for 20 separate instances with 100 employees, 52 weeks and 5 skills, for the exact model with the weight of the third part of the objective function being a factor 10 bigger than the other two, and the approximate model with the weight of the first part of the objective function being a factor 10 bigger than the other two.

Instance	Run time Exact - w3=100	Objective value Exact - w3=100	Value third Exact - w3=100
1	113.066	5964.3	12
2	1057.344	6105.6	12
3	724.653	13101.5	4
4	135.234	25742.1	12
5	370.585	4531.9	4
6	211.544	22293.8	12
7	209.489	7906.1	12
8	69.031	10242.0	4
9	68.999	14015.7	12
10	108.512	4429.8	12
11	479.769	7461.2	8
12	36.114	9959.0	4
13	118.447	12963.2	12
14	271.467	7483.2	20
15	80.325	6204.9	8
16	152.815	6763.8	8
17	122.752	7165.9	4
18	811.912	5155.0	8
19	175.131	9351.2	0
20	359.431	7582.4	12

Table 6.6: Run time in seconds, objective value and the value for the third part of the initial model for the 20 separate instances with 100 employees, 52 weeks, 5 skills and the weight of the third part of the objective function being a factor 100 bigger than the other two.

7 Conclusion, discussion and recommendation

In this thesis, we have created two models which give a working schedule for 100 employees with 5 skills for 52 weeks. From the exact linear model, we obtain the schedule for which the difference between contract hours and working hours for each employee each week, over and under staffing and the difference between contract hours and working hours for each employee over the total planning horizon, is minimized. Next to this model, we have made an approximate model, which is meant to approximate the solution of the problem within a shorter amount of time.

We used Gurobi to solve our implemented problem. We compared the run time for our two models first for a version in which we do not distinct between skills. The average run time of 20 instances of these models for 100 employees, 52 weeks and equal weights for each part of the objective function was 17.26 seconds and 0.99 seconds for the exact and the approximate model, respectively.

After that, we extend the models with a distinction between skills. For these models, we evaluated the effect of giving different weights to parts of the objective function. For varying the weight on the first part of the objective function, we are able to use the extended version of the approximate model. When the wish is to emphasize the second part of the objective function, the approximate model did not manage to find an approximate solution of the problem within a shorter amount of time when a weight is used that is 50 times bigger than the other two. If this is the case, we can use a time limited version of our exact model, which has a deviation of at most 5% from the exact model without time limit for 15 out of 20 instances, when using a time limit of 120 seconds. It would be interesting for future research to analyze this with higher weights and more instances. The approximate model also encountered problems when changing the third weight. This made us use the approximate model with using the weight on the first part of the objective function. The weights do not exactly correspond, so the outcome for the third part of the initial objective function is not the same, but the third part does get minimized by the approximate model. It would be interesting for future research to analyze more precisely why the approximate model stops being able to solve the problem in a shorter amount of time when varying the weight on the second or the third part of the model. If possible it would be interesting to create one approximate model which is able to provide a quick approximation for the optimal solution for any weights.

The overall outcome does not seem that strange to us, since there are way more options in interchanging the hours of the employees to fit the demand, and to make an employees hours sum up to the needed amount of contract hours over the whole planning period, than to interchange hours to give an employee its contract hours each week. This, however, is an unsubstantiated statement, so we would recommend to investigate this in future research.

It could also be interesting to do more research on (the relation between) the weights. We have now chosen our weights based on the maximum values of each part of the objective function, but there might be a better way to do this.

Bibliography

[1] Carlos S. Azmat, Tony Hürlimann, and Marino Widmer. Mixed integer programming to schedule a single-shift workforce under annualized hours. *Annals of Operations Research*, 128(1):199–215, 2004.

[2] A. Corominas, A. Lusa, and R. Pastor. Using MILP to plan annualised working hours. *Journal of the Operational Research Society*, 53(10):1101–1108, 2002.

[3] Albert Corominas, Amaia Lusa, and Rafael Pastor. Planning annualised hours with a finite set of weekly working hours and joint holidays. *Annals of Operations Research*, 128(1):217–233, 2004.

[4] Amaia García and Rafael Pastor. Planning working time accounts under demand uncertainty. *Computers Operations Research*, 38:517–524, 2011.

[5] Amaia García, Albert Corominas, and Norberto Muñoz. A scenario optimisation procedure to plan annualised working hours under demand uncertainty. *International Journal of Production Economics*, 113, 2008.

[6] B. Grabot and A. Letouzey. Short-term manpower management in manufacturing systems: new requirements and DSS prototyping. *Computers in Industry*, 43:11–29, 2000.

[7] Gulzar Hasan, Irfan Ali, and S S Hasan. Annualized hours planning with fuzzy demand constraint. *ProbStat Forum*, 9:50–56, 2016.

[8] Rudy Hung. A multiple-shift workforce scheduling model under annualized hours. *Naval Research Logistics (NRL)*, 46(6):726–736.

[9] J.H. Keim. Annualised hours: Optimisation with heuristic algorithm. Bachelor thesis, TU Delft, 2018. URL `http://resolver.tudelft.nl/uuid:cff60809-75e8-4908-967f-1b93419b1079`.

[10] M. van der Geest. Waarom u zo lang moet wachten in een ziekenhuis – en wat daaraan te doen is. Interview met Taco van der vaart. *Volkskrant*, 23-05-2019.

[11] Egbert Van der Veen, Erwin W. Hans, Bart Veltman, Leo M. Berrevoets, and Hubert J.J.M. Berden. A case study of cost-efficient staffing under annualized hours. *Health Care Management Science*, 18(3):279–288, 2015.