August 28, 2018

MASTER THESIS

AUTOMATICALLY SCHEDULING TRANSPLANT SURGEONS AT THE LUMC WHILE EVENLY DISTRIBUTING SURGICAL EXPERIENCE

Nicky P. E. Schuermans

Faculty of Electrical Engineering, Mathematics and Computer Science (EEMCS)
Stochastic Operations Research

Supervisors:
Dr. Ir. A. Braaksma
Prof. Dr. R. J. Boucherie
Drs. N. E. Papadopoulos

LUMC Leiden University Medical Center

UNIVERSITY OF TWENTE.

Summary

The transplant department of the Leiden University Medical Center (LUMC) experiences difficulties with scheduling the transplant surgeons. This is because the transplant surgeons have more shifts than other departments. Therefore, many surgeries are unexpected and therefore it is not known when these surgeries will take place. The schedules of the different tasks are made per quarter and are currently made by one of the surgeons by hand. The aim of this research is to schedule the transplant surgeons such that the number of surgeries done per year is divided over the surgeons. This is currently not part of their planning environment as equally as possible. To do this, first an integer linear program (ILP) is developed for automatically making the quarter schedules. The number of surgeries performed is not taken into account here. By using this ILP optimal schedules, given the availability of the surgeons, that fulfill all constraints can be made automatically. However, when we look at the fourth quarter of 2018 shortages occur during different tasks in this schedule. Also at the organ removal shift, which is the most important task to schedule. Therefore, the model is adjusted to schedule the organ removal shift by another structure.

Then, we do take into account the number of surgeries performed. A stochastic dynamic program (SDP) is developed to do this for a more generic case, where persons must be scheduled over different tasks and during these tasks orders can arrive. The goal here is to equally divide the number of completed orders over the persons at the end of the horizon. We solve this SDP for different small instances with the goal to find a structure in the optimal actions that can be extended to bigger instances. Most of the times the optimal action is such that the expected number of surgeries at the end of the schedule is divided as equally as possible. However, there are some exceptions. This SDP is adjusted to the LUMC case, but we cannot solve this due to the curse of dimensionality. Therefore, we approximate the optimal solution by using the ILP model for scheduling the different tasks over the surgeons and add the condition that we want to divide the expected number of surgeries at the end of the schedule as equally as possible over the surgeons. By using this model, the LUMC can include the number of surgeries performed per surgeon in their planning environment. the LUMC to use the model for scheduling the transplant surgeons and take into account the number of surgeries performed per surgeon. However, to do this a user friendly application should be developed. By doing this optimal schedules, given the availability of the surgeons, can be made that fulfill all constraints, the program is impartial and the surgeons do not have to spend time at scheduling the different tasks over the surgeons.

Preface

This report is the result of my graduation project which is a part of the master program in Applied Mathematics of the University of Twente. The research is performed at the transplant department of the Leiden University Medical Center from January 2018 until August 2018.

First, I would like to thank my supervisors Aleida Braaksma and Richard Boucherie for providing me advice, new ideas and feedback when needed. Also the CHOIR group for the help and support.

I would like to thank Niki Papadopoulos for this opportunity. I would also like to thank the transplant surgeons for their enthusiasm about my project. Especially Dorottya de Vries and Ian Alwayn who where always willing to answer my questions about the schedules.

Finally, I would like to thank my family and friends for supporting me during this research. Especially, Jasmijn Manders who was always willing to listen and think along when I needed it.

Nicky Schuermans

Enschede, August 2018

Contents

Chapter 1

Introduction

In the Netherlands, only academic hospitals can perform transplants of organs at patients who need a transplant organ. Each academic hospital can perform kidney transplants and the government decides if a hospital can also perform transplants of other organs. The Leiden University Medical Center (LUMC) performs transplants of kidneys, pancreas and livers [Nederlandse Transplantatie Stichting, 2018]. The transplant surgeons in the LUMC do not only perform the transplants of organs at patients who need a transplant organ, but also the organ removals are performed by them. In the Netherlands five hospitals have a team that can perform these organ removals. A team that can perform an organ removal consists of two surgeons, two operating room assistants and an anaesthesiologist. The only thing they need in the hospital where the organ removal takes place is a free operating room. They bring all other requirements in their specially equipped bus [van der Hoeven, 2013].

The transplant department of the LUMC experiences difficulties with scheduling the transplant surgeons. The problem is that transplant surgeons have more shifts than other departments because in comparison with other departments the transplant department has more emergency surgeries. With emergency surgeries, we mean the surgeries that cannot be scheduled during the operating room (OR) days. These emergency surgeries are organ transplants and organ removals. For the emergency transplants transplant shifts must be scheduled and for the organ removals the organ removal shift must be scheduled. Not all transplants are emergency transplants. For example, a kidney donation with a living donor can be scheduled during the OR. One example of an emergency transplant is a organ transplant with a deceased donor. All organ removals are emergency surgeries, although this removal can be known a day in advance. This is because the treatment of a patient is stopped and this patient is an organ donor. However, the time of the removal is not known exactly and the removal can be cancelled if the organs of the patient are not good enough.

Not only these shifts are done by transplant surgeons, also the inpatients should be visited, the OR should be scheduled for transplants that can be scheduled and the outpatient clinic should be scheduled. Making a schedule for these surgeons is, therefore, a difficult job, performed by one of the transplant surgeons by hand. Every quarter, this surgeon spends one day on making the schedules and after that about fifteen minutes per week for the changes in the schedule. The different tasks must fulfill all kind of constraints when these tasks are scheduled. For example, always a senior surgeon should be scheduled at the transplant shift and always at least one multi-organ donation (MOD) certified surgeon should be scheduled at the organ removal shift when this shift should be scheduled. Furthermore, it is desirable to distribute the number of surgeries done by the end of the year as equally as possible over the surgeons. In this way, every surgeon can gain about the same amount of experience during a year. This is currently not a part of their planning environment.

The goal of this research is to schedule the transplant surgeons in such a way that the number of surgeries done per year is divided over the surgeons as equally as possible. Therefore this research aims to investigate:

In which way can the schedules of the transplant surgeons be made to improve the schedules and to equally divide the number of surgeries done over the transplant surgeons per year?

To answer this question, we will first develop a model for making the quarter scheduled without taking into account the number of surgeries done per surgeon. For this, an integer linear program (ILP) is developed and the following questions will be answered.

– *Which tasks are done by transplant surgeons?*

- *When should these tasks be scheduled?*
- *When can these tasks be scheduled for another surgeon at the next day?*
- *What are the constraints for the schedules?*

When this is done, we also want to consider the number of surgeries done. This is done by developing a Stochastic Dynamic Program (SDP). First the SDP will be developed for a more generic case, where we have persons who can be scheduled at different tasks and orders can arrive during tasks. The goal is to equally divide the number of completed orders over the persons at the end of the horizon. The following questions will be answered for the generic case

- *What is the distribution of the transition probabilities?*
- *Is a deterministic Markovian policy optimal in this case?*
- *Can we find a structure in the optimal actions?*
- *How can these results be extended to the LUMC context?*

This report is organized as follows. In Chapter 2 we give an overview of some related literature. In Chapter 3 we describe the tasks done by transplant surgeons and when these tasks should be scheduled. Chapter 4 describes the developed ILP model for making the quarter schedule. Chapter 5 shows the results from the ILP. Chapter 6 describes the model that includes the number of surgeries done per surgeon. In Chapter 7 results of this model are shown. Chapter 8 we will adjust the generic model for equally dividing the number of orders over the persons to the LUMC case and use an ILP to approximate the optimal schedule. Finally, in Chapters 9 and 10 we will respectively discuss and conclude our research.

Chapter 2

Literature review

A transplant is the replacement of a poor or completely non-functional organ of a patient. This is done by transplant surgeons. The transplants can be done at every moment of the day because it must be done as soon as possible when a donor organ is available. Therefore, always a transplant surgeon should be available for transplant. But also transplant surgeons should be available for taking the donor organ. Therefore, different tasks must be scheduled over the transplant surgeons. To schedule these tasks some constraints must be met. It is preferred that the number of surgeries is equally divided over the transplant surgeons per year. This is currently not included in the planning environment of the LUMC.
To schedule in this way, first, a model is made for scheduling the different tasks over the surgeons without considering the number of surgeries done per surgeon. Some literature about this part is described in Section 2.1. Later, this first part is used to schedule the transplant surgeons with considering the number of surgeries done per surgeon. Some literature about this part is described in Section 2.2.

2.1 Scheduling tasks

A lot of research has been done about employee scheduling [Bruecker et al., 2015, den Bergh et al., 2013]. Both articles are a literature review about employee scheduling. den Bergh et al. [2013] uses four classification fields: the personal characteristics, the constraints, the solution method and uncertainty incorporation and the application area and applicability of research. For the first field, the contract can be considered, as is done in Akbari et al. [2013]. In our case all surgeons are working full-time, so we do not take this into account. We are considering the skills of the different surgeons, as is done in Akbari et al. [2013]. Not all surgeons can be scheduled at all tasks. Often hierarchical skills are used, employees with a lower skill level can do less than employees with a higher-level skill. In this research, no hierarchical skills are used because different surgeons can do different tasks. We also have a different experience of surgeons. When the transplant shifts are scheduled always one surgeon must be a senior surgeon. This is a senior with more experience. The distinction in experience is also made in Majozi and Zhu [2005]. Further, a distinction is made between multi-organ donation (MOD) certified surgeons and not MOD certified surgeons. When the organ removal shifts should be scheduled at least one of the surgeons should be certified. In this research we schedule tasks for individuals. It is also possible to schedule tasks for teams, as is done in Dück et al. [2012]. Different solution methods can be used for employee scheduling. Most of the times mathematical programming is used to solve the scheduling problem, with sometimes a heuristic when the problem is big. In this research also, mathematical programming will be used. To schedule the tasks over the surgeons integer linear programming (ILP) is used. More about ILP can be found in Section 4.1.

2.2 Include the number of surgeries done per surgeon

When we include the number of surgeries done per surgeon, the goal is to equally divide this number over the surgeons per year. Every quarter we make a schedule for the different tasks over the surgeons such that all constraints hold. Every task has a probability distribution of the number of surgeries done during this task. Stochastic Dynamic Programming (SDP) is used for this. More about SDP can be found in Section 6.1. An SDP is used for problems with a finite horizon. When the horizon is infinite Markov Decision Process (MDP) can be used.

MDP/SDP is used in healthcare on various subjects. For example, Chow et al. [2012] and van Arendonk et al. [2015] use MDP for kidney transplants. Chow et al. [2012] use an MDP to allow patients to see their estimated survival probability after accepting versus declining an infectious risk donor offer over a five-year horizon. van Arendonk et al. [2015] use an MDP to compare the outcomes of the different options for a given pediatric kidney transplant candidate with only one living donor available. This candidate must choose between immediate primary living donor kidney transplant, followed if necessary by deceased donor retransplantation versus waiting to undergo primary deceased donor kidney tranplantation, followed if necessary by living donor retransplantation (if the living donor is still able to donate) or deceased donor retransplantation (if the living donor is no longer able to donate).

In our research we are using an SDP, because we want to divide the number of surgeries done per year as equally as possible. Therefore, we have a finite horizon. An SDP can be used when the target is fixed as is done in Sever et al. [2018] and we want to know the optimal route to this target. In our research, the target is the line where all surgeons have done an equal number of surgeries per year and we want to approach this line as good as possible at the end of the year. Tsai [2017] has done something similar by scheduling operating room (OR) and outpatient department (OD) sessions for orthopedic surgeons. Each surgeon has a budgeted amount of OR and OD sessions per year. Therefore, the number of OR and OD sessions scheduled per year is fixed. The goal is to schedule the OR and OD sessions in such a way that the expected number of patients in the OR queue at the beginning of the next week is minimized. The state of the system is the number of patients that is referred to radiology, screening and the number of patients in the OR queue. In our research the tasks that must be scheduled are fixed. However, these tasks can be done by multiple surgeons. The target at the end of the year is also not known, but we want to distribute the number of surgeries as equally as possible over the surgeons. Therefore, we want to approach the line where all surgeons have done the same number of surgeries as well as possible. Tsai [2017] used an upper boundary to guarantee that each patient has the possibility to be scheduled within a certain amount of weeks with a high enough probability and solves the problem by using an exact stochastic dynamic program (SDP), but to make the SDP numerically tractable approximations for the transition probabilities are used. In our research no boundaries are needed to guarantee things about the goal.

Our research has similarities with that of Hulshof et al. [2016]. This article looks at a generalized situation of the situation of Tsai [2017], a network of queues. First, a mixed integer linear program (MILP) is developed in Hulshof et al. [2013] and in Hulshof et al. [2016] an approximate dynamic program (ADP) is developed for the same system extended with stochastic aspects, such as random patient arrivals and patient transitions between queues. On every decision epoch is decided how many patients are treated who already have been waiting a certain amount of time. The action space is restricted by the number of resources available. The state of the system is equal to the number of patients in the different queues with certain waiting time. Therefore, in this article the target is not known. They used value iteration with a cost function as a function of the waiting time per individual queue. Post-decision states are used for a single approximation for the outcome state, the basis function approach is used to approximate the value function and an ILP is used to overcome the large decision space for large problem instances. Solving a Dynamic Program is typically intractable due to the curse of dimensionality, as the problem size increase. Therefore, Hulshof et al. [2016] first develops an SDP to solve small instances and uses ADP for bigger instances. More about ADP can be found in Powell [2011].

To the best of our knowledge no research has been done on an SDP with in which the target equals a line. In Zhang et al. [2018] an SDP is used to compute the optimal trajectory of an underwater vehicle subject to some mission objectives. This SDP is applied to a submarine whose goal is to detect one or several targets, or/and to minimize its own detection range perceived by the other targets. To do this, discretisation grids that are dynamic and concentrated at most probable positions of the targets are used. The costs only depend on the submarine and target relative positions and some fixed parameters. In this article, the target is not equal to a line. However, in our research the costs depend on the relative position of the state of the system from the line where all surgeries are equally divided over the surgeons.

Chapter 3

Context analysis

Different tasks are done by transplant surgeons. Section 3.1 describes which tasks must be scheduled. These tasks are scheduled for a surgeon when a switching moment occurs. The switching moments are described in Section 3.2. Last, the fixed structure of scheduling the organ removal shift is discussed in Section 3.3.

3.1 Tasks

The tasks done by transplant surgeons are shown below. At each shift, one surgeon should be scheduled.

- **T1**: This is the transplant shift from 18:00 till 8:00 o'clock on weekdays and is done by the same surgeon the whole week. During this shift, the transplant surgeon works in the night and is home during the day. The weekend $T1$ shift is from Saturday 8:00 o'clock till Monday 8:00 o'clock. During this shift in the morning, the surgeon visits inpatients and the surgeon is available for emergency cases. After this shift, on Monday no task can be scheduled because it is not known how busy the shift was.
- **T2**: This is the reserve transplant shift. Some surgeries must be done with two surgeons, so also two surgeons should be available to do this operation at every moment in time. It is also possible that the surgeon of the $T1$ shift is busy and the surgeon of $T2$ must fill in. This shift is from 22:00 o'clock till the next day 22:00 o'clock. One surgeon is scheduled at this shift for the whole weekend. So, this surgeon starts at Friday 22:00 o'clock and is scheduled at the $T2$ shift till Sunday 22:00 o'clock. By scheduling shifts, we want to minimize the number of switching moments between surgeons, because information can be lost by switching from surgeon. However, this shift is a 24h shift and therefore this shift should not be scheduled too long for the same surgeon. Therefore, this shift is scheduled two or three days for the same surgeon. It is also possible to schedule this shift four days for the same surgeon, but this is not preferred.
- **Td**: This is the transplant shift during the day, from 8:00 o'clock till 18:00 o'clock. The continuity of this shift is not important, so it can be done by another surgeon every day. Supervising the intensive care is also part of the shift. This shift is only scheduled during weekdays.
- **M1**: This is the organ removal shift. This shift is done together with the transplant team of the Erasmus MC. One week this shift is done by the transplant team of the Erasmus MC and the other week this shift is done by the transplant team of the LUMC. The $M1$ and $M2$ shift is at the same time done by the same transplant team. So, one week $M1$ and $M2$ are done by the transplant team of the Erasmus MC and the other week these shifts are done by the transplant team of the LUMC. The shift starts at Friday 12:00 o'clock and ends the next Friday 12:00 o'clock. The switch between surgeons of the LUMC is at 22:00 o'clock. The start and switch moments of this shift are fixed. When the organ removal shift should be scheduled, one surgeon starts with the $M1$ shift on Friday and this surgeon is scheduled at this task till Sunday. The next surgeon starts on Sunday evening and is scheduled till Tuesday. The last surgeon starts on Tuesday evening and is scheduled till Friday at the $M1$ shift. One of the surgeons of the $M1$ or $M2$ shift should be multi-organ donation (MOD) certified.
- **M2**: This shift is the same as the $M1$ shift.

The following tasks should also be scheduled. At each of these tasks, one surgeon should be scheduled. These tasks should be scheduled on weekdays.

- **Hpb**: This is the chef Hpb. Hpb is a department for the liver, pancreas, bile duct and gallbladder. During this task phone calls are answered on these subjects and inpatients are visited. This shift is not considered heavy and is regularly combined with the $T2$ shift.
- **Ch**: chef ward. This is the same as the chef Hpb, but for the transplant department.

While the schedule of the tasks described above is made per quarter, the next tasks are not scheduled until three weeks before they take place.This is because the shifts are more important. But these tasks could be scheduled at the same time as the shifts.

- **Outpatient clinic (OC)**: This is at Monday, Tuesday, and Wednesday. Every Monday two surgeons should be scheduled at the outpatient clinic and the surgeon with the Hpb task also has outpatient clinic. At Tuesday and Wednesday, the surgeon with Ch task has outpatient clinic and no other surgeons are scheduled at the outpatient clinic.
- **Operating room (OR)**: This is at Wednesday, Thursday, and Friday. During these days two surgeons should be scheduled at the operating room. Per two weeks one day is OR reduction. So, during this day no operations can be scheduled. This OR reduction day is not fixed and is known a few weeks in advance. When the OR task is scheduled during the quarter schedule, the OR task should be scheduled an Wednesday, Thursday and Friday, because we do not know when the OR reduction days are.

The following tasks are scheduled for the whole surgery department. So, a transplant surgeon can be scheduled at this task, but also another surgeon can do this task. These tasks are scheduled after the other tasks are scheduled. So, these tasks are not considered when the other tasks for the transplant surgeons are scheduled.

- **CVD**: This is the surgeon of the day. This task can be combined with other tasks. During this task, the surgeon answers phone calls.
- **Front guard surgery**: This shift is from 18:15 till 23:00 o'clock. When this task is done by a transplant surgeon, the surgeon with the T1 shift has this task.

3.2 Switching moments

A switching moment of a task is the moment another surgeon is scheduled for this task. The switching moments have fixed times. These moments are shown in Figures 3.1 – 3.8. These figures show when the tasks should be scheduled. This is the case when a block is filled. When the pattern of a block changes another transplant surgeon can be scheduled. The figures are from the first Friday the organ removal shifts should be scheduled (Figure 3.1) till the last Friday of an organ removal shift week (Figure 3.8). We will now look at the shifts during these days in more detail.

- **First Friday organ removal week**
  We start with the first Friday of an organ removal shift week shown in Figure 3.1. We see that the outpatient clinic is not scheduled during this day and the organ removal shift starts at 12:00 o'clock. No other tasks than the Hpb task can be scheduled for the surgeon that starts with the organ removal shift. This is because of the compensation of the organ removal shift. The pattern of the $T2$ shift changes at 22:00 o'clock. Therefore, at Friday 22:00 o'clock is a switching moment of the $T2$ shift. This does not mean that another surgeon must be scheduled, but it is possible to schedule another surgeon.

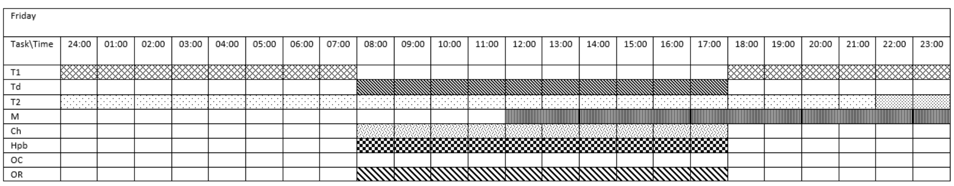

Figure 3.1: The shifts on Friday when the organ removal shifts start.

- **Saturday**
In Figure 3.2 the shifts scheduled during Saturday are shown. We see that during the weekends only the $T1$, $T2$ and M shifts are scheduled. One switching moment occurs during Saturday. This switching moment is at 8:00 o'clock at the $T1$ shift.

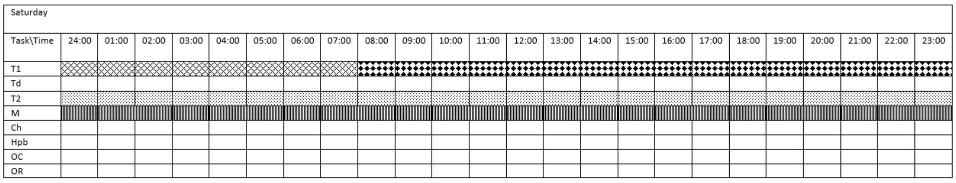

Figure 3.2: The shifts on Saturday when the organ removal shifts must be scheduled.

- **Sunday**
In Figure 3.3 the shifts scheduled on Sunday are shown. We see that the same shifts are scheduled as on Saturday. We can also see that a shift is scheduled for the whole weekend if a shift is scheduled at Saturday, because no switching moments occur between Saturday morning and Sunday evening. On Sunday evening the $T2$ shift has a switching moment at 22:00 o'clock.

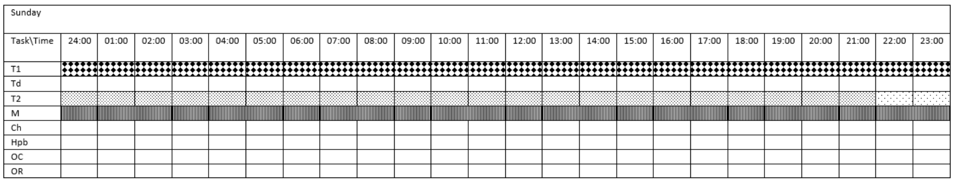

Figure 3.3: The shifts on Sunday when the organ removal shifts must be scheduled.

- **Monday**
In Figure 3.4 the shifts scheduled on Monday are shown. We see that the operating room is not scheduled on Monday, but all other tasks are. We can also see that the weekend $T1$ shift ends at Monday 8:00 o'clock and at 18:00 o'clock another surgeon is scheduled at the $T1$ shift. The M shift has a switching moment at 12:00 o'clock and the $T2$ shift also has a switching moment on Monday. This switching moment is at 22:00 o'clock. While the weekend M shifts ends at Monday 12:00 o'clock, the surgeon cannot be scheduled at another task this day. This is because this surgeon has compensation of this shift during the rest of the day. Also, the surgeon that starts on 12:00 o'clock with the M shift cannot be scheduled at another task on this day, because of the compensation for this shift.

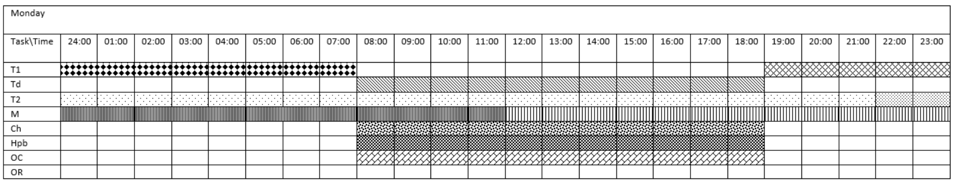

Figure 3.4: The shifts on Monday when the organ removal shifts must be scheduled.

- **Tuesday**
In Figure 3.5 the shifts scheduled on Tuesday are shown. We see that the operating room is not scheduled on Tuesday. The surgeon with the Ch task has also outpatient clinic. Therefore, the outpatient clinic has the same pattern in the blocks on Tuesday as the Ch task. Another surgeon can be scheduled at the Td shift on Tuesday than on Monday. The same thing holds for the Hpb and Ch tasks. The $T2$ shift has a switching moment on Tuesday 22:00 o'clock.

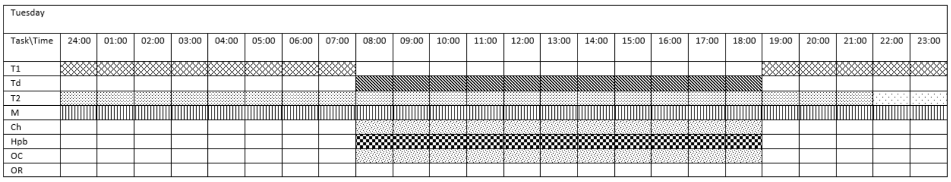

Figure 3.5: The shifts on Tuesday when the organ removal shifts must be scheduled.

- **Wednesday**
  In Figure 3.6 the shifts scheduled on Wednesday are shown. The surgeon with the Ch task has outpatient clinic. Therefore, the pattern in the blocks of the outpatient clinic and the Ch task are the same. The Td shift, Ch task, and Hpb tasks can be done at Wednesday by another surgeon than on Tuesday. Also, the M shift has a switching moment at 12:00 o'clock and the $T2$ shift has a switching moment on Wednesday at 22:00 o'clock. While the M shift that starts at Monday ends at Wednesday 12:00 o'clock, the surgeon cannot be scheduled at another task on Monday. This surgeon has compensation of this shift during the rest of the day. Also, the surgeon that starts on 12:00 o'clock with the M shift cannot be scheduled at another task on this day, because of the compensation for this shift.

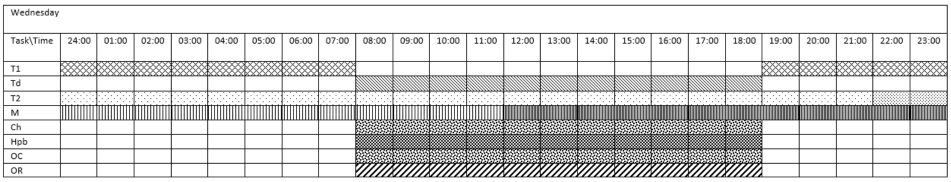

Figure 3.6: The shifts on Wednesday when the organ removal shifts must be scheduled.

- **Thursday**
  In Figure 3.7 the shifts scheduled on Thursday are shown. We see that the outpatient clinic is not scheduled on Thursday. The Td shift, Ch task, and Hpb tasks can be done at Thursday by another surgeon than on Wednesday. Also, the $T2$ shift has a switching moment on Thursday at 22:00 o'clock.

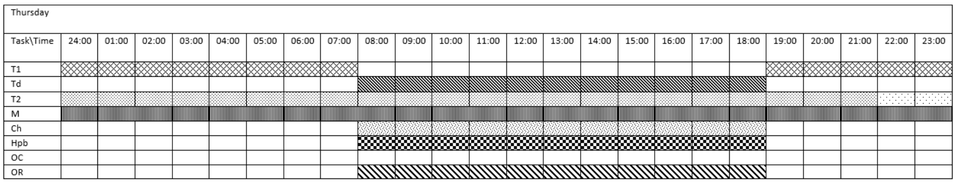

Figure 3.7: The shifts on Thursday when the organ removal shifts must be scheduled.

- **Last Friday organ removal week**
  In Figure 3.8 the shifts scheduled at the last Friday of the organ removal week are shown. We see that the outpatient clinic is not scheduled on Friday. The Td shift, Ch task, and Hpb tasks can be done at Friday by another surgeon than on Thursday. The organ removal shift ends at 12:00 o'clock and the $T2$ shift has a switching moment on Friday at 22:00 o'clock. The surgeon scheduled at the organ removals shift cannot be scheduled at another task during this day, because of the compensation of the organ removal shift.

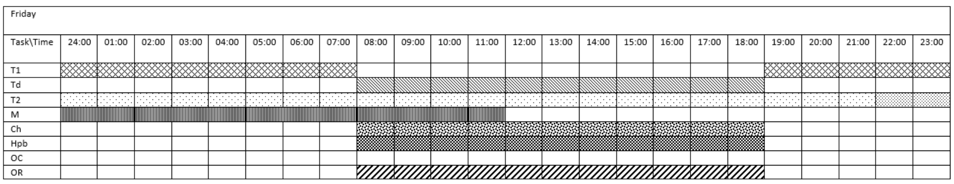

Figure 3.8: The shifts on the last Friday of the the organ removal shift week.

- **Combine $T1$ and Td shift**
  During a vacation period, it can be decided to combine the $T1$ and Td shifts during weekdays. The surgeon scheduled at this combination must be available from 22:00 o'clock till 22:00 o'clock the next day. Therefore, the switching moments change when these shifts are combined. The switching moments, in this case, are the same as the switching moments of the $T2$ shift described above. Therefore, the $T1$ shift is not scheduled for the same surgeon the whole week and we do not have this continuity. Nevertheless, we have no transfer moment between the $T1$ and Td shift. Therefore, we do have more continuity during a day.

3.3 Scheduling organ removal shift

The organ removal shift has a fixed structure as shown in Section 3.2. This structure is new and is made by the zut ('zelfstandige uitname teams'). Therefore, this structure must be used in the Netherlands. This structure is made, because surgeons where not always fit during the organ removal shift. Before this structure, no rules were used for scheduling the organ removal shift. Therefore, some surgeons where scheduled a whole week at the organ removal shift, while this shift is a 24h shift. The goal of this structure is to have fit surgeons scheduled at the organ removal shift.

However, this structure leads to challenges with scheduling the organ removal shift. This is shown in Table 3.1. This table shows the number of surgeons that should be scheduled at the organ removal shift per day during two weeks. We see that the differences between the number of surgeons scheduled during a day during week one or two is at least equal to two, except for Friday. The Friday is the only day without differences in the number of surgeons that must be scheduled. The difference of two surgeons cannot be reduced by using another structure of scheduling the organ removal shift, when this shift should be scheduled every other week and two surgeons should be scheduled. However, the difference in the number of surgeons that must be scheduled on Monday and Wednesday during an organ removal shift week and no organ removal shift week is four surgeons. This difference is a result of the structure, because two surgeons are scheduled during the morning on these days and have compensation during the rest of the day and two surgeons are scheduled from 12:00 o'clock and have compensation the rest of the day.

Also, the fixed consecutive days that must be scheduled can lead to more shortages on the organ removal shift. The surgeons have a fixed day of the week a non-clinical day (no tasks can be scheduled during this day) and the organ removal shift block that contains this day cannot be scheduled for this surgeon. While without this structure, the organ removal shift can only not be scheduled during the non-clinical day.

Table 3.1: The number of surgeons scheduled at the organ removal shift per day during two weeks. Week one is the week where no organ removal shift is scheduled and during week two the organ removal shift is scheduled.

Day	Number of surgeons scheduled	
	Week one	Week two
Monday	0	4
Tuesday	0	2
Wednesday	0	4
Thursday	0	2
Friday	2	2
Saturday	2	0
Sunday	2	0

Chapter 4

Model for scheduling tasks

This chapter formulates a model for scheduling the different tasks over the transplant surgeons as described in Chapter 3. These tasks are scheduled per day over the surgeons during a quarter. The schedule must fulfill various conditions. For example, not every surgeon can be scheduled at every task, some tasks are scheduled for multiple days at once etcetera. The schedule is made in such a way that the shortages are minimized. The shortages are the tasks that are not scheduled, but should be scheduled. Not only the shortages are minimized, also the number of times one surgeon is scheduled at multiple tasks, the difference between the number of tasks scheduled per surgeon etcetera. This is done by formulating an integer linear program. First, Section 4.1 describes integer linear programming. Then, Section 4.2 shows the notation used. Section 4.3 shows the conditions for the variables; this is when the variables are created and when not. Section 4.4 shows the constraints and Section 4.5 the objective function. The complete model is shown in Appendix A. Section 4.6 describes how we get a schedule from the solution of the ILP and Section 4.7 describes three models to schedule the organ removal shift differently.

4.1 Integer Linear Programming (ILP)

A linear program (LP) is a mathematical optimization problem of the following form: [Bertsimas and Tsitsiklis, 1998]

$$\begin{aligned} \min \quad & C^T \cdot \boldsymbol{x} \\ \text{s.t.} \quad & A \cdot \boldsymbol{x} \leq B \\ & x_j \geq 0 \qquad \forall j \end{aligned} \tag{4.1}$$

The matrix A and vectors B, C are given and $x_1, \ldots, x_n$ are decision variables. The LP is an integer linear program (ILP) if $\boldsymbol{x} \in \mathbb{Z}^d$. An $\boldsymbol{x}$ is called a feasible solution of (4.1) if it fulfills all constraints and it is called an optimal solution if it minimizes $C^T \cdot \boldsymbol{x}$ among all feasible solutions. $C^T \cdot \boldsymbol{x}$ is called the objective function. In the remainder of this chapter, we will formulate an ILP for scheduling the different tasks over the surgeons.

4.2 Notation

This section describes the notation used for the model for scheduling the tasks over the transplant surgeons. First, the basic notation is described. The notation for the different shifts and other tasks is shown below. The two organ removal shifts are combined in the M shift. The two shifts are the same, so this can also be done by scheduling two surgeons at the M shift.

Notation	**Task**
$T1$	Transplant shift
$T2$	Transplant shift backup
Td	Transplant shift during the day
M	Organ removal shift
Ch	Chef department
Hpb	Chef hpb
OC	Outpatient clinic
OR	Operating room

Also, some notation is used for the days during a week. Each day of the week has a number as shown below. This is done because some constraints only hold for particular days.

Notation	**Day of the week**
0	Sunday
1	Monday
2	Tuesday
3	Wednesday
4	Thursday
5	Friday
6	Saturday

Now that the basic notation is introduced, the sets, variables and parameters can be described. This is done by respectively Section 4.2.1, 4.2.2 and 4.2.3.

4.2.1 Sets

The sets used are shown below. First, the set of all tasks that should be scheduled is shown, then the set of the surgeons who can do the tasks or a part of the tasks. Last, the set of the week numbers and the set of the day numbers is shown. These sets contain numbers corresponding to weeks/days for which tasks should be scheduled. The length of the set depends on the horizon of the schedule. So, when we want to schedule tasks during a quarter (13 weeks), the set of week numbers consist of the numbers 1 till 13 and the set of the day numbers consist of the numbers 1 till 91.

Set	**Description**	**Index**
$T = \{T1, T2, Td, M, Ch, Hpb, OC, OR\}$	Tasks	t
$S = \{s_1, \ldots, s_{\lvert S\rvert}\}$	Surgeons	s
$W = \{w_1, \ldots, w_{\lvert W\rvert}\}$	Week numbers	w
$D = \{d_1, \ldots, d_{\lvert W\rvert \cdot 7}\}$	Day numbers	d

4.2.2 Variables

This section describes the variables used in the model for scheduling the different tasks. First, the decision variables are described and then the state variables. The decision variables are the variables corresponding to the decisions that must be made, and the state variables are the variables following from the decision variables.

Decision variables

The decisions that must be made are which surgeon is scheduled at which task during a day. This is done with the variables shown below. This variable indicates when a particular task is scheduled for a particular surgeon during a day. When a task is scheduled for some surgeon during a day, the variable is equal to one. When the task is not scheduled for the surgeon during the day, the variable is equal to zero. Some tasks are scheduled for more days at once. For example the organ removal shift is scheduled from Friday till Monday. In this case we only schedule the start moment of this task (Friday) and we know that this shift also is scheduled during Saturday till Sunday. By doing this, fewer decisions have to be made by the model.

$$x_{s,t,d} = \begin{cases} 1 & \text{if surgeon } s \text{ starts with task } t \text{ at day } d \\ 0 & \text{otherwise} \end{cases}$$

State variables

The state variables are shown below. To indicate when two tasks are combined the first four variables are used. By combining tasks we mean that one surgeon is scheduled at both tasks. When tasks are combined the number of shortages can decrease. However, we want to minimize the number of combinations. Therefore, we have to know when tasks are combined. Some combinations are less desirable than other combinations and therefore we also want to know which tasks are combined. This is done with the first variable for the chef Hpb and the $T2$ shift.

The second one indicates when the Hpb task is combined with the M shift. The third one indicates when the Hpb task is combined with the Ch task and the fourth one when the $T2$ shift is combined with the operating room task. The fifth variable indicates if other combinations are made. By other combinations we mean other combinations than described above from the chef tasks, the $T2$ shift and the Td. These combinations are less desirable than the combinations described above, but are possible to reduce the number of shortages. When necessary these tasks can be scheduled at one surgeon, but also a part of these tasks can be scheduled at one surgeon. The last variable indicates if these other combinations are made during a day. This is because we also want to minimize the number of days other combinations are needed to reduce the shortages.

$$y_d = \begin{cases} 1 & \text{if one surgeon is scheduled at the tasks } Hpb \text{ and } T2 \text{ shift at day } d \\ 0 & \text{otherwise} \end{cases}$$

$$w_d = \begin{cases} 1 & \text{if one surgeon is scheduled at the tasks } Hpb \text{ and } M \text{ shift at day } d \\ 0 & \text{otherwise} \end{cases}$$

$$u_d = \begin{cases} 1 & \text{if one surgeon is scheduled at the tasks } Hpb \text{ and } Ch \text{ at day } d \\ 0 & \text{otherwise} \end{cases}$$

$$z_d = \begin{cases} 1 & \text{if one surgeon is scheduled at the } T2 \text{ shift and } OR \text{ at day } d \\ 0 & \text{otherwise} \end{cases}$$

$$mc_{d,t,t'} = \begin{cases} 1 & \text{if one surgeon is scheduled at the } t \text{ task and the } t' \text{ task at day } d \\ & t, t' \in \{Ch, Hpb, T2, Td\} \\ 0 & \text{otherwise} \end{cases}$$

$$mcd_d = \begin{cases} 1 & \text{if } mc_{d,t,t'} > 0 \text{ for some } t, t' \\ 0 & \text{otherwise} \end{cases}$$

The following two variables indicate if a surgeon is scheduled at the Hpb/Ch task. This is used to minimize the number of different surgeons scheduled at these tasks. For the continuity of a department it is undesired that five different surgeons are scheduled on the Ch task during one week. The same thing holds for the Hpb task. Therefore, the number of different surgeons that are scheduled on the Ch task during a week is minimized. The same thing is done for the Hpb task.

$$ih_{s,w} = \begin{cases} 1 & \text{if surgeon } s \text{ is scheduled at the } Hpb \text{ task at week } w \\ 0 & \text{otherwise} \end{cases}$$

$$ic_{s,w} = \begin{cases} 1 & \text{if surgeon } s \text{ is scheduled at the } Ch \text{ task at week } w \\ 0 & \text{otherwise} \end{cases}$$

When many surgeons are on holiday during a week, the organ removal shift can be scheduled from Friday till Monday and from Monday till Friday. Therefore, the switching moment on Wednesday expires. The next variable indicates when this happens. This variable is used, because we want to minimize the number of times this is happening.

$$lm_w = \begin{cases} 1 & \text{if the } M \text{ shift is scheduled from Monday till Friday at week } w \\ 0 & \text{otherwise} \end{cases}$$

A weekend $T2$ shift can be scheduled before a week $T1$ shift, but preferably not because the weekend $T2$ shift is a 24h shift and the $T1$ shift is a week night shift. The variable shown below indicates if the $T2$ weekend shift is scheduled before a week $T1$ shift during a week.

$$wk_w = \begin{cases} 1 & \text{if the } T2 \text{ weekend shift is scheduled before a week } T1 \text{ shift at week } w \\ 0 & \text{otherwise} \end{cases}$$

The $T2$ shift can be scheduled for four days consecutively at the same surgeon, but at most three days consecutive $T2$ shift is preferred. This is because the $T2$ shift is a 24h shift. The variable shown below is used to indicate when the $T2$ shift is scheduled four days consecutively. By using this variable we can minimize the number of times that the $T2$ shift is scheduled for four days consecutive for the same surgeon.

$$fd_d = \begin{cases} 1 & \text{if the } T2 \text{ shift is scheduled for four days consecutively at the same surgeon,} \\ & \text{starting at day } d \\ 0 & \text{otherwise} \end{cases}$$

Also, at the boundary (the beginning of the schedule) must be indicated when four days consecutive $T2$ shift is scheduled. This is done with the two variables shown below.

$$fdm = \begin{cases} 1 & \text{if on the first Monday the fourth consecutive day } T2 \text{ shift is scheduled for some surgeon} \\ 0 & \text{otherwise} \end{cases}$$

$$fdtu = \begin{cases} 1 & \text{if on the first Tuesday the fourth consecutive day } T2 \text{ shift is scheduled for some surgeon} \\ 0 & \text{otherwise} \end{cases}$$

The variables shown below are the maximum number of days a surgeon is scheduled at a certain task and maximum total number. The first variable is used to equally divide the number of days a surgeon is scheduled at some task over the surgeons. The second variable is used to equally divide workload. By doing this not always the same surgeon has multiple tasks at one day.

$$\begin{aligned} mds_t &\in \mathbb{R}^+ \cup \{0\} && \text{maximum number of days scheduled at task } t \\ md &\in \mathbb{R}^+ \cup \{0\} && \text{maximum total number of tasks scheduled} \end{aligned}$$

To ensure a feasible solution exists even when some tasks cannot be scheduled, a dummy variable is used. When the dummy variable is equal to one for some task and day, one surgeon fewer than needed is scheduled at this tasks at this day. The maximum value the dummy variable can take is the number of surgeons that should be scheduled at the task. When enough surgeons are scheduled, the dummy variable has value zero.

$$d_{t,d} \in [0, MAX_t] \qquad \text{number of surgeons fewer than needed scheduled}$$

Also, a dummy variable is used for the constraint that always an MOD certified surgeon is scheduled when the organ removal shift must be scheduled. The dummy variable used for this is shown below.

$$dm_d = \begin{cases} 1 & \text{if no MOD certified surgeon can be scheduled during day } d \text{ and} \\ & \text{the } M \text{ shift should be scheduled} \\ 0 & \text{otherwise} \end{cases}$$

For the constraints that always a senior surgeon is available at the transplant shifts and every week an hpb surgeon is scheduled at the operating room also dummy variables are used. These are shown below.

$$ds_d = \begin{cases} 1 & \text{if no senior surgeon can be scheduled during day } d \text{ at} \\ & \text{a transplant shift} \\ 0 & \text{otherwise} \end{cases}$$

$$dhpb_w = \begin{cases} 1 & \text{if no hpb surgeon can be scheduled at the operating room during week } w \\ 0 & \text{otherwise} \end{cases}$$

4.2.3 Parameters

This section describes the parameters used in the model for scheduling the tasks. The parameters contain information that is known before the schedule is made. For example, the days off for the surgeons. This information is needed for the constraints.

The first parameter is used to indicate if a surgeon is MOD certified. One of the surgeons scheduled for the organ removal shifts should be MOD certified every day this task should be scheduled. This is done by using the parameters shown below. The first parameter indicates if the surgeon is MOD certified during the whole schedule. If the surgeon is not MOD certified during the whole schedule, the second parameter can be used to indicate which days the surgeon is MOD certified.

$$MOD_s = \begin{cases} 1 & \text{if surgeon } s \text{ is MOD certified} \\ 0 & \text{otherwise} \end{cases}$$

$$MODC_{s,d} = \begin{cases} 1 & \text{if surgeon } s \text{ is MOD certified during day } d \\ 0 & \text{otherwise} \end{cases}$$

Also for scheduling the transplant surgeons we need information about the surgeons. A surgeon can be a junior surgeon. It is not possible that both surgeons who are scheduled at the same time at the transplant shift are junior surgeons. Therefore, the parameter shown below is used. This parameter indicates if a surgeon is a junior surgeon.

$$JR_s \quad = \quad \begin{cases} 1 & \text{if surgeon } s \text{ is a junior} \\ 0 & \text{otherwise} \end{cases}$$

For scheduling the operating room we also need information about the surgeons. One day per week hpb surgeries are scheduled. For these hpb surgeries hbp surgeons must be scheduled. Therefore, every week at least one hpb surgeon must be scheduled at the operating room. To indicate which surgeons are hpb surgeons the parameter shown below is used.

$$HPB_s \quad = \quad \begin{cases} 1 & \text{if surgeon } s \text{ is an hpb surgeon} \\ 0 & \text{otherwise} \end{cases}$$

Some surgeons cannot handle a whole week of night shifts (T1 shift) well. Therefore, these surgeons can do more weekend $T1$ shifts. During a weekend $T1$ shift, the shift is not only during the night but also during the day. The weekend shift starts at Saturday morning and ends at Monday morning, where the week night shift is five consecutive days. The next parameter indicates if a surgeon prefers weekend shifts over week shifts, because the surgeon cannot handle a whole week of night shifts well.

$$FWT1_s \quad = \quad \begin{cases} 1 & \text{if surgeon } s \text{ cannot handle well a whole week (Monday till Friday) night shifts} \\ 0 & \text{otherwise} \end{cases}$$

Every surgeon has a non-clinical day during the week. During the non-clinical day no tasks are scheduled for this surgeon. This non-clinical day is usually on the same day of the week. The parameter shown below indicates the standard non-clinical day of a surgeon.

$$NCD_s \quad = \quad \begin{cases} 1 & \text{if surgeon } s \text{ has a non-clinical day on Monday} \\ 2 & \text{if surgeon } s \text{ has a non-clinical day on Tuesday} \\ 3 & \text{if surgeon } s \text{ has a non-clinical day on Wednesday} \\ 4 & \text{if surgeon } s \text{ has a non-clinical day on Thursday} \\ 5 & \text{if surgeon } s \text{ has a non-clinical day on Friday} \\ -1 & \text{if it is not important when surgeon } s \text{ has a non-clinical day} \end{cases}$$

The non-clinical days can also be changed. For a new non-clinical day, the parameter shown below is used.

$$ENCD_{s,d} \quad = \quad \begin{cases} 1 & \text{if surgeon } s \text{ has an extra non-clinical day at day number } d \\ 0 & \text{otherwise} \end{cases}$$

When the standard non-clinical day is not a non-clinical day at a day, the parameter shown below is used.

$$NNCD_{s,d} \quad = \quad \begin{cases} 1 & \text{if surgeon } s \text{ has no non-clinical day at day number } d \\ 0 & \text{otherwise} \end{cases}$$

The organ removal shift is the first and last day half a day. Therefore, some surgeons do not have a problem with the fact that this is scheduled during their non-clinical day. This is indicated with the parameters shown below. The first one is for a morning organ removal shift (end of the organ removal shift) and the second one for an afternoon organ removal shift (start of the organ removal shift).

$$MONCD_{s,d} \quad = \quad \begin{cases} 1 & \text{if surgeon } s \text{ does not have a problem with a morning organ removal shift during} \\ & \text{his/her non-clinical day during day } d \\ 0 & \text{otherwise} \end{cases}$$

$$MNCD_{s,d} \quad = \quad \begin{cases} 1 & \text{if surgeon } s \text{ does not have a problem with an afternoon organ removal shift} \\ & \text{during his/her non-clinical day during day } d \\ 0 & \text{otherwise} \end{cases}$$

We also have to take into account where the different surgeons can be scheduled. Not all surgeons can be scheduled at all tasks. The parameter shown below indicates which tasks can be scheduled for each surgeon.

$$TS_{s,t} \quad = \quad \begin{cases} 1 & \text{if surgeon } s \text{ can be scheduled at task } t \\ 0 & \text{otherwise} \end{cases}$$

Surgeons can go to a congress or go on holiday. During these days no tasks can be scheduled. This is done with the parameter shown below.

$$V_{s,d} = \begin{cases} 1 & \text{if surgeon } s \text{ is on holiday or congress at day } d \\ 0 & \text{otherwise} \end{cases}$$

Sometimes a surgeon can do the start or the end of the organ removal shift during the day off. This is indicated with the parameters shown below. The first one for the start of the organ removal shift and the second one for the end of the organ removal shift.

$$MV_{s,d} = \begin{cases} 1 & \text{if surgeon } s \text{ is on holiday or congress at day } d \text{ and can start the organ removal} \\ & \text{shift during this day} \\ 0 & \text{otherwise} \end{cases}$$

$$MOV_{s,d} = \begin{cases} 1 & \text{if surgeon } s \text{ is on holiday or congress at day } d \text{ and can end the organ removal} \\ & \text{shift during this day} \\ 0 & \text{otherwise} \end{cases}$$

It is also possible that a surgeon does not want to do shifts during a day. During this day, only tasks during office hours can be scheduled. Also, an organ removal shift can be scheduled during a day when the surgeon does not want to do shifts, but only the morning can be scheduled. When a surgeon does not want to do shift on Friday, this surgeon can be scheduled at the organ removal shift from Wednesday till Friday. This is done with the parameter shown below.

$$G_{s,d} = \begin{cases} 1 & \text{if surgeon } s \text{ does not want to do shifts at day } d \\ 0 & \text{otherwise} \end{cases}$$

During a vacation period, it can be decided that the $T1$ and Td shifts are combined. In this case, one surgeon is scheduled for both tasks. By doing this, one surgeon fewer is needed per day. To indicate when these shifts are combined, the parameter shown below is used.

$$COMB_w = \begin{cases} 1 & \text{if the } T1 \text{ and } Td \text{ shifts are combined during week } w \\ 0 & \text{otherwise} \end{cases}$$

The organ removal shifts are scheduled every other week. So, the weeks that this shift must be scheduled must be indicated. This is done with the parameter shown below. This parameter is equal to one if the whole first week the organ removal shift should be scheduled and zero if the whole second week the organ removal shift should be scheduled.

$$MW = \begin{cases} 1 & \text{if the first Monday of the schedule the organ removal shifts must be scheduled} \\ 0 & \text{if the second Monday of the schedule the organ removal shifts must be scheduled} \end{cases}$$

When some tasks are already fixed, this can be filled in the following parameter. This can be done at the start moments of the tasks. Tasks are for example fixed for surgeons who only are doing organ removal shifts and have another speciality than transplantation.

$$X_{s,t,d} = \begin{cases} 1 & \text{if surgeon } s \text{ must be scheduled at task } t \text{ at day number } d \\ 0 & \text{otherwise} \end{cases}$$

The public holidays can be indicated with the parameter shown below. During public holidays only the shifts that are done during weekends must be scheduled.

$$PH_d = \begin{cases} 1 & \text{if day } d \text{ is public holiday} \\ 0 & \text{otherwise} \end{cases}$$

When the public holiday is Christmas, this is indicated with the parameter shown below. During Christmas no surgeon can be scheduled both days at the same task.

$$CHRISTMAS_d = \begin{cases} 1 & \text{if day } d \text{ is Christmas} \\ 0 & \text{otherwise} \end{cases}$$

The maximum value of the dummy variable is the number of surgeons needed for a task. The maximum value of the dummy variable per task is shown below.

$$MAX_t = \begin{cases} 1 & \text{if } t \in \{T1, T2, Td, Ch, Hpb\} \\ 2 & \text{if } t \in \{M, OC, OR\} \end{cases}$$

To distribute the tasks over the surgeons such that the workload is equally divided, the number of days a surgeon has done a certain task must be included. In this case, not only the tasks are equally divided during the schedule, but it considers the period before the start date of the schedule. This is done with the parameter shown below. This is the number of days a surgeon already has done a certain task.

$$DT_{s,t} \in \mathbb{N} \qquad \text{number of days surgeon } s \text{ has done task } t$$

To determine the importance factor of equally dividing tasks over surgeons, we need to know how many days the different tasks are scheduled during the schedule. This is done with the following parameter. We say that the tasks is scheduled two days when a task is scheduled for two surgeons at one day.

$$DTS_t \in \mathbb{N} \qquad \text{number of days task } t \text{ must be scheduled}$$

Some parameters are used to ensure the conditions also hold at the boundary (the beginning of the schedule). These parameters contain information of the schedule before the start date of the new schedule. The parameter shown below indicates if the surgeon has done a shift at the last weekend before the start date.

$$WB_s = \begin{cases} 1 & \text{if surgeon } s \text{ has done a shift in the last weekend before this schedule} \\ 0 & \text{otherwise} \end{cases}$$

The next parameter indicates if the surgeon has done a shift in the second-last weekend before the start date.

$$WBT_s = \begin{cases} 1 & \text{if surgeon } s \text{ has done a shift in the second-last weekend before this schedule} \\ 0 & \text{otherwise} \end{cases}$$

The parameter shown below indicates if the surgeon has done a $T1$ shift in the last week before the start date.

$$TWB_s = \begin{cases} 1 & \text{if surgeon } s \text{ has done } T1 \text{ shift in the last week before this schedule} \\ 0 & \text{otherwise} \end{cases}$$

The parameter shown below indicates if the surgeon has done an $T1$ shift in the last weekend before the start date.

$$TW_s = \begin{cases} 1 & \text{if surgeon } s \text{ has done the } T1 \text{ shift in the last weekend before this schedule} \\ 0 & \text{otherwise} \end{cases}$$

The parameter shown below indicates if the surgeon has done an M shift in the last weekend before the start date.

$$WBM_s = \begin{cases} 1 & \text{if surgeon } s \text{ has done the } M \text{ shift in the last weekend before this schedule} \\ 0 & \text{otherwise} \end{cases}$$

To ensure not more than four days consecutive $T2$ is scheduled at the boundary, the parameter shown below is used for the number of consecutive days a surgeon has done $T2$ shift before the schedule.

$$ND_s \in \{0,1,2,3,4\} \qquad \text{number of consecutive days } T2 \text{ before schedule}$$

When the $T1$ shift is combined with the Td shift, not more than three consecutive days $T1$ shift can be scheduled for the same surgeon. To ensure this holds at the boundary, the parameter shown below is used for the number of consecutive days a surgeon has done $T1$ shift before the schedule.

$$NDT_s \in \{0,1,2,3\} \qquad \text{number of consecutive days } T1 \text{ before schedule}$$

4.3 Conditions for creating variables

Not all variables are used for all combinations of the indices. The variables are not created for the combinations that are not possible. By doing this, the model does not have to take into account these variables and the solving time is smaller. For example, when a surgeon has a day off this surgeon cannot be scheduled. Therefore, we do not create the decision variables to schedule this surgeon during his/her day off. Also, when a surgeon cannot be scheduled at the Ch task, we do not have to look if this surgeon is scheduled during a week at the Ch task, because we know that this is not possible. Section 4.3.1 shows the constraints for the variable that indicates if a surgeon is scheduled at the Hpb/Ch task during a week, Section 4.3.2 shows the conditions for the variable that shows the maximum number of days a surgeon is scheduled at a task, Section 4.3.3 for the variables for combining different tasks, Section 4.3.4 for the variables for scheduling the different tasks and Section 4.3.5 for the dummy variable.

4.3.1 Surgeon scheduled at Hpb/Ch

A surgeon can only be scheduled at the Hpb task during a week if a surgeon can do this task. Therefore, if a surgeon cannot be scheduled at the Hpb task, this surgeon is not taken into account in indicating how many different surgeons are scheduled at this task during a week. This is done with constraint (4.2). The same thing holds for the Ch task. This is done with constraint (4.3).

$$ih_{s,w} = 0 \qquad \forall s, w \quad \text{st} \quad TS_{s,Hpb} = 0 \tag{4.2}$$
$$ic_{s,w} = 0 \qquad \forall s, w \quad \text{st} \quad TS_{s,Ch} = 0 \tag{4.3}$$

4.3.2 Maximum number of days at a task

Only the tasks $T1$, $T2$, M and OR must be distributed equally over the surgeons, all other tasks not. Therefore, we do not have to determine the maximum number of days a surgeons is scheduled at the other tasks. This is done with constraints (4.4).

$$mds_t = 0 \qquad \forall t \in \{Td, Hpb, Ch, OC\} \tag{4.4}$$

4.3.3 Variables for combining tasks

The Hpb task and $T2$ shift can only be combined during weekdays. Therefore, the variable that indicates if these tasks are combined is equal to zero during the other days. This is done with constraint (4.5).

$$y_d = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{0, 6\} \tag{4.5}$$

The Hpb task and the M shift can only be combined when the M shift starts at 12:00 o'clock (first Friday of organ removal shift, Monday of organ removal shift and Wednesday of organ removal shift). Therefore, the variable that indicates if these tasks are combined is equal to zero during the other days. This is done with constraint (4.6).

$$w_d = 0 \qquad \forall d \quad \text{st} \quad d \bmod 14 \notin \{5 + 7 \cdot MW, 1 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW)\} \tag{4.6}$$

The OR task and $T2$ shift can only be combined on Wednesday, Thursday, and Friday. Therefore, the variable that indicates if these tasks are combined is equal to zero during the other days. This is done with constraint (4.7).

$$z_d = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{0, 1, 2, 6\} \tag{4.7}$$

The Ch and Hpb tasks can be combined on Monday till Friday. Therefore, the variable that indicates if these tasks are combined is equal to zero during the weekend. This is done with constraint (4.8).

$$u_d = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{0, 6\} \tag{4.8}$$

The day tasks and the $T2$ shift can only be combined during weekdays. This is done with constraint (4.9). Also, only the $T2$ shift, the chef tasks and the Td shift can be combined. Therefore, the variable is zero for all other combinations. This is done with constraints (4.10) and (4.11). We only take into account the tasks when the t task is earlier in the set of the different tasks than the t' task. This is done because otherwise we would create two variables for each combination while we need only one. This is done with constraint (4.12). Last, for the combinations which were already possible no extra variable has to be created. This is done with constraint (4.13).

$$mc_{d,t,t'} = 0 \qquad \forall d, t, t' \quad \text{st} \quad d \bmod 7 \in \{0, 6\} \tag{4.9}$$
$$mc_{d,t,t'} = 0 \qquad \forall d, t, t' \quad \text{st} \quad t \in \{T1, M, OC, OR\} \tag{4.10}$$
$$mc_{d,t,t'} = 0 \qquad \forall d, t, t' \quad \text{st} \quad t' \in \{T1, M, OC, OR\} \tag{4.11}$$
$$mc_{d,t,t'} = 0 \qquad \forall d, t, t' \quad \text{st} \quad t \prec t' \tag{4.12}$$
$$mc_{d,t,t'} = 0 \qquad \forall d, t, t' \quad \text{st} \quad t' = Hpb, t \in \{T2, Ch\} \tag{4.13}$$

The combinations that can be made by variable mc_d, t, t' can only be done during weekdays. Therefore, the variable that indicates which days these tasks are combined is equal to zero during the weekend. This is done with constraint (4.14)

$$mcd_d = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{0, 6\} \tag{4.14}$$

4.3.4 Variables for scheduling tasks

The variables for scheduling tasks do not have to be created for all combinations of indices for several reasons. These are explained below.

Start moments

Only the start moments of the different tasks should be scheduled. When a shift starts at $22:00$ o'clock, we say that the starting moment is the next day. When for example the $T2$ shift starts on Tuesday 22:00 o'clock, the starting moment is on Wednesday. When a shift does not start at a day, no surgeon has to be scheduled for this task at this day. So, no variable must be created for this day and this task.

This is done with constraints (4.15) and (4.16) for the $T1$ shift. The $T1$ shift must be scheduled every day. However, the whole weekend this shift is done by the same surgeon. Therefore, we say that the weekend shift has as starting moment Saturday and we do not schedule the Sunday, because we know that the surgeon scheduled on Saturday at the $T1$ task also is scheduled on Sunday at this task. This is done with constraint (4.15). When the $T1$ shift is not combined with the Td shift, this shift is done by the same surgeon from Monday till Friday. Therefore, we only schedule this shift on Monday in this case. This is done with constraint (4.16). When the $T1$ shift is combined with the Td shift, every weekday we have a switching moment. Therefore, all variables for scheduling the $T1$ shift during the week are created.

$$x_{s,T1,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 0 \tag{4.15}$$

$$x_{s,T1,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{2,3,4,5\}, COMB_{\lceil \frac{d}{7} \rceil} = 0 \tag{4.16}$$

The $T2$ shift must always be scheduled. However, the $T2$ shift is the whole weekend done by the same surgeon. Therefore, the surgeon that is scheduled for the $T2$ shift on Saturday is also scheduled on Sunday. This shift is not scheduled on Sunday, because we know that the surgeon who is scheduled on Saturday is also scheduled on Sunday. This is done with constraint (4.17)

$$x_{s,T2,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 0 \tag{4.17}$$

The M shift only must be scheduled every other week. During a week that the M shift should be scheduled, the first Friday, Monday, and Wednesday are the start moments. On Friday the shift starts at 12:00 o'clock and ends on Monday 12:00 o'clock. When a surgeon is scheduled on Monday on the M shift, this surgeon starts at 12:00 o'clock and is scheduled till Wednesday 12:00 o'clock. When a surgeon is scheduled on Wednesday on the M shift, the surgeon starts at 12:00 o'clock and is scheduled till Friday 12:00 o'clock. Therefore, the M shift only is scheduled at these fixed switch moments. This is done with constraint (4.18).

$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 14 \notin \{1 + 7\cdot(1-MW), 3 + 7\cdot(1-MW), 5 + 7\cdot MW\} \tag{4.18}$$

The Td, Ch and Hpb only must be scheduled during weekdays and not during the weekend. This is done with constraint (4.19). The outpatient clinic only must be scheduled on Monday. This is done with constraint (4.20). Last, the operating room must only be scheduled on Wednesday, Thursday, and Friday. This is done with constraint (4.21).

$$x_{s,t,d} = 0 \qquad \forall s,t,d \quad \text{st} \quad t \in \{Td, Ch, Hpb\}, d \bmod 7 \in \{0,6\} \tag{4.19}$$

$$x_{s,OC,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{0,2,3,4,5,6\} \tag{4.20}$$

$$x_{s,OR,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{0,1,2,6\} \tag{4.21}$$

During non-clinical day no task during the day

During the non-clinical day, no task, except the $T1$ shift, should be scheduled. Sometimes the start of end of the organ removal shift can be scheduled during the non-clinical day if the surgeon does not have problems with this. The non-clinical day is a fixed day of the week per surgeon. It is possible the non-clinical day is changed, this can be done by the parameters $ENCD_{s,d}$ and $NNCD_{s,d}$. During the non-clinical day no other tasks than the $T1$ or M shift can be scheduled. This is done with constraint (4.22) for the fixed non-clinical day per week and with constraint (4.23) for the non-clinical days that are scheduled at another day.

$$x_{s,t,d} = 0 \qquad \forall s,t,d \quad \text{st} \quad t \notin \{T1, M\}, d \bmod 7 = NCD_s, NNCD_{s,d} = 0 \tag{4.22}$$

$$x_{s,t,d} = 0 \qquad \forall s,t,d \quad \text{st} \quad t \notin \{T1, M\}, ENCD_{s,d} = 1 \tag{4.23}$$

When the non-clinical day of a surgeon is during a day the whole day M shift is scheduled at the same surgeon, this M shift cannot be scheduled for this surgeon. This is done with constraints (4.24) and (4.25) for the second day of the M shift. So, a non-clinical day on Tuesday and the M shift that starts on Monday and a non-clinical day on Thursday and the M shift that starts on Wednesday. Constraint (4.24) is for the standard non-clinical day and constraint (4.25) for the extra non-clinical days.

$$\begin{aligned} x_{s,M,d} = 0 \quad & \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1+7\cdot(1-MW), \\ & 3+7\cdot(1-MW)\}, (d+1) \bmod 7 = NCD_s, NNCD_{s,d+1} = 0 \qquad (4.24) \\ x_{s,M,d} = 0 \quad & \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1+7\cdot(1-MW), 3+7\cdot(1-MW)\}, \\ & ENCD_{s,d+1} = 1 \qquad (4.25) \end{aligned}$$

The M shift that starts on Monday can be scheduled when a surgeon has the standard non-clinical day on Wednesday and has no problems with a Morning M shift during his/her non-clinical day. This is done with constraint (4.26). When a surgeon has an extra non-clinical day on Wednesday, the M shift that starts on Monday cannot be scheduled, unless the surgeon has no problems with a Morning M shift. This is done with constraint (4.27). The M shift that starts on Wednesday can be scheduled when a surgeon has the standard non-clinical day on Friday and has no problems with a morning M shift during his/her non-clinical day. This is also done with constraint (4.26). Constraint (4.27) does the same for an extra non-clinical day.

$$\begin{aligned} x_{s,M,d} = 0 \quad & \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1+7\cdot(1-MW), 3+7\cdot(1-MW)\}, \\ & (d+2) \bmod 7 = NCD_s, NNCD_{s,d+2} = 0, MONCD_{s,d+2} = 0 \qquad (4.26) \\ x_{s,M,d} = 0 \quad & \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1+7\cdot(1-MW), 3+7\cdot(1-MW)\}, \\ & ENCD_{s,d+2} = 1, MONCD_{s,d+2} = 0 \qquad (4.27) \end{aligned}$$

The M shift that starts on Friday ends on Monday. Therefore, this shift can only be scheduled if the surgeon has no problems with a morning M shift during a non-clinical day on Monday or when the surgeon does not have a non-clinical day on Monday. This is done with constraint (4.28) for the fixed non-clinical day and with constraint (4.29) for the extra non-clinical day.

$$\begin{aligned} x_{s,M,d} = 0 \quad & \forall s,d \quad \text{st} \quad d \bmod 14 = 5+7\cdot MW, (d+3) \bmod 7 = NCD_s, \\ & NNCD_{s,d+3} = 0, MONCD_{s,d+3} = 0 \qquad (4.28) \\ x_{s,M,d} = 0 \quad & \forall s,d \quad \text{st} \quad d \bmod 14 = 5+7\cdot MW, ENCD_{s,d+3} = 1, MONCD_{s,d+3} = 0 \qquad (4.29) \end{aligned}$$

The M shift can be scheduled when the surgeon does not have a non-clinical day during the start day of this shift or this surgeon does not have problems with an afternoon M shift during his/her non-clinical day. This is done with constraint (4.30) for the fixed non-clinical day and with constraint (4.31) for the extra non-clinical day.

$$\begin{aligned} x_{s,M,d} = 0 \quad & \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1+7\cdot(1-MW), 3+7\cdot(1-MW), 5+7\cdot MW\}, \\ & d \bmod 7 = NCD_s, NNCD_{s,d} = 0, MNCD_{s,d} = 0 \qquad (4.30) \\ x_{s,M,d} = 0 \quad & \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1+7\cdot(1-MW), 3+7\cdot(1-MW), 5+7\cdot MW\}, \\ & ENCD_{s,d} = 1, MNCD_{s,d} = 0 \qquad (4.31) \end{aligned}$$

Shifts that can be done per surgeon

When a surgeon cannot do a task, this surgeon-task combination can never be scheduled. This is done with constraint (4.32).

$$x_{s,t,d} = 0 \qquad \forall s,t,d \quad \text{st} \quad TS_{s,t} = 0 \qquad (4.32)$$

Holiday or congress

When a surgeon is on holiday or at congress, the surgeon cannot be scheduled at tasks during this day. First, this is done for weekdays. Then for the weekend days.

- **Weekdays**
  The start moment of the organ removal shift cannot be scheduled if a surgeon has a day off during the days belonging to this start moment. However, the first or last day of the organ removal shift can be scheduled if the surgeon has indicated that this is possible. This is done with constraints (4.33) – (4.37).

On Friday, Monday and Wednesday the organ removal shift cannot be scheduled if the surgeon has a day off and the surgeon cannot be scheduled for an afternoon at the organ removal shift during the start day. This is done with constraint (4.33).

$$x_{s,M,d} = 0 \qquad \forall s, d \quad \text{st} \quad d \bmod 14 \in \{5 + 7 \cdot MW, 1 + 7 \cdot (1 - MW), \\ 3 + 7 \cdot (1 - MW)\}, V_{s,d} = 1, MV_{s,d} = 0 \tag{4.33}$$

The organ removal shift can also not be scheduled on Friday, Monday or Wednesday if the surgeon has a day off the next day. This is done with constraint (4.34).

$$x_{s,M,d} = 0 \qquad \forall s, d \quad \text{st} \quad d \bmod 14 \in \{5 + 7 \cdot MW, 1 + 7 \cdot (1 - MW), \\ 3 + 7 \cdot (1 - MW)\}, V_{s,d+1} = 1 \tag{4.34}$$

When the surgeon has a day off on Sunday, the organ removal shift that starts on Friday cannot be scheduled for this surgeon. This is done with constraint (4.35).

$$x_{s,M,d} = 0 \qquad \forall s, d \quad \text{st} \quad d \bmod 14 = 5 + 7 \cdot MW, V_{s,d+2} = 1 \tag{4.35}$$

When the surgeon has a day off on Wednesday, the organ removal shift that starts on Monday can only be scheduled if the surgeon has indicated that the surgeon can be scheduled for the organ removal shift on Wednesday morning. The same thing holds for scheduling the organ removal shift on Wednesday when the surgeon has a day of on Friday. This is done with constraint (4.36).

$$x_{s,M,d} = 0 \qquad \forall s, d \quad \text{st} \quad d \bmod 14 \in \{1 + 7 \cdot (1 - MW), \\ 3 + 7 \cdot (1 - MW)\}, V_{s,d+2} = 1, MOV_{s,d+2} = 0 \tag{4.36}$$

The organ removal shift that starts on Friday ends on Monday. Therefore, this organ removal shift cannot be scheduled if a surgeon has a day of on Monday and cannot be scheduled on Monday morning at the organ removal shift. This is done with constraint (4.37).

$$x_{s,M,d} = 0 \qquad \forall s, d \quad \text{st} \quad d \bmod 14 = 5 + 7 \cdot MW, V_{s,d+3} = 1, MOV_{s,d+3} = 0 \tag{4.37}$$

Tasks can also not be scheduled when a surgeon has a day off during the start moment for other tasks than the organ removal shift. This is done with constraint (4.38).

$$x_{s,t,d} = 0 \qquad \forall s, t, d \quad \text{st} \quad t \neq M, d \bmod 7 \in \{1, 2, 3, 4, 5\}, V_{s,d} = 1 \tag{4.38}$$

The $T1$ shift is scheduled from Monday till Friday for the same surgeon if the $T1$ shift and Td shift are not combined. Therefore, this shift cannot be scheduled if the surgeon has a day off during a week. This is done with constraints (4.39) (day off on Tuesday)– (4.42) (day off on Friday).

$$x_{s,T1,d} = 0 \qquad \forall s, d \quad \text{st} \quad d \bmod 7 = 1, V_{s,d+1} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \tag{4.39}$$

$$x_{s,T1,d} = 0 \qquad \forall s, d \quad \text{st} \quad d \bmod 7 = 1, V_{s,d+2} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \tag{4.40}$$

$$x_{s,T1,d} = 0 \qquad \forall s, d \quad \text{st} \quad d \bmod 7 = 1, V_{s,d+3} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \tag{4.41}$$

$$x_{s,T1,d} = 0 \qquad \forall s, d \quad \text{st} \quad d \bmod 7 = 1, V_{s,d+4} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \tag{4.42}$$

- **Weekend days**
  When a surgeon has a day off during the weekend, no weekend shifts can be scheduled. This is done with constraints (4.43).

$$x_{s,t,d} = 0 \qquad \forall s, t, d \quad \text{st} \quad t \in \{T1, T2\}, d \bmod 7 = 6, V_{s,d} + V_{s,d+1} \geq 1 \tag{4.43}$$

Public holiday

During public holidays only the shifts that are done during weekends must be scheduled (M, $T1$ and $T2$). All other tasks are not scheduled. This is done with constraint (4.44).

$$x_{s,t,d} = 0 \qquad \forall d, t \quad \text{st} \quad t \in \{Td, Ch, Hpb, OC, OR\}, d \bmod 7 \in \{1, 2, 3, 4, 5\}, PH_d = 1 \tag{4.44}$$

No shifts

When a surgeon does not want to do shifts during a day, no shifts should be scheduled. First, this is done for the weekdays. Then for the weekend days.

- **Weekdays**
When a surgeon does not want to do shifts during a day, the $T1$, $T2$ and M shift cannot be scheduled for this surgeon. This is done with constraint (4.45).

$$x_{s,t,d} = 0 \qquad \forall s,t,d \quad \text{st} \quad t \in \{T1, T2, M\}, d \bmod 7 \in \{1,2,3,4,5\}, G_{s,d} = 1 \tag{4.45}$$

The organ removal shift is scheduled for multiple days at once. Therefore, this shift can only be scheduled if the surgeon can be scheduled at shifts during these days. The organ removal shift cannot be scheduled at Friday, Monday or Wednesday if the surgeon does not want to do shifts on the next day. This is done with constraint (4.46). On Friday the organ removal shift can also not be scheduled if the surgeon does not want to do shifts on Sunday. This is done with constraint (4.47). The organ removal shift can be scheduled if the surgeon does not want to do shifts during the last day. This is because the shift ends at 12:00 o'clock. When a surgeon does not want to do shifts, this means the surgeon cannot be scheduled after office hours during this day.

$$\begin{aligned} x_{s,M,d} = 0 \qquad & \forall s,d \quad \text{st} \quad d \bmod 14 \in \{5 + 7 \cdot MW, 1 + 7 \cdot (1 - MW), \\ & 3 + 7 \cdot (1 - MW)\}, G_{s,d+1} = 1 \qquad (4.46) \\ x_{s,M,d} = 0 \qquad & \forall s,d \quad \text{st} \quad d \bmod 14 = 5 + 7 \cdot MW, G_{s,d+2} = 1 \qquad (4.47) \end{aligned}$$

The $T1$ shift is scheduled from Monday till Friday for the same surgeon if the $T1$ shift and Td shift are not combined. Therefore, this shift cannot be scheduled during a week if the surgeon does not want to do shifts during a day of this week. This is done with Constraints (4.48) (no shift on Tuesday) – (4.51) (no shift on Friday).

$$x_{s,T1,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 1, G_{s,d+1} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \tag{4.48}$$
$$x_{s,T1,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 1, G_{s,d+2} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \tag{4.49}$$
$$x_{s,T1,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 1, G_{s,d+3} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \tag{4.50}$$
$$x_{s,T1,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 1, G_{s,d+4} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \tag{4.51}$$

- **Weekend days**
When a surgeon cannot be scheduled on a shift one day during the weekend, no weekend shifts can be scheduled. This is done with constraint (4.52).

$$x_{s,t,d} = 0 \qquad \forall s,t,d \quad \text{st} \quad t \in \{T1, T2\}, d \bmod 7 = 6, G_{s,d} + G_{s,d+1} \geq 1 \tag{4.52}$$

Only M, OC and OR tasks or only M shifts
When a surgeon can only be scheduled at the M, OC and OR, it is input when the M task is scheduled. This is because this M task is also scheduled during days the surgeon is normally not available. Therefore, for all other days no variables are created to schedule the M shift for that surgeon. The variables for scheduling the M task is only created when it is known that the M shift must be scheduled. The same thing holds for surgeons who can only be scheduled at M shifts, because the surgeon is only available at the moments the M shift must be scheduled. This is done with constraint (4.53). When the surgeon can also be scheduled at the OC and OR tasks, the variables for scheduling these tasks are created when the surgeon is available. Therefore, no extra constraints are needed for these tasks.

$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad \sum_t TS_{s,t} = TS_{s,M} + TS_{s,OC} + TS_{s,OR}, X_{s,M,d} = 0 \tag{4.53}$$

4.3.5 Dummy variable

Shortages can only occur when the task must be scheduled. So, the dummy variable is only needed when the tasks should be scheduled. This is done with constraints (4.54) – (4.59). The $T1$ and $T2$ shift are scheduled on Saturday for the whole weekend. Therefore no shortages can occur during Sunday. This is done with constraint (4.54).

$$d_{t,d} = 0 \qquad \forall t,d \quad \text{st} \quad t \in \{T1, T2\}, d \bmod 7 = 0 \tag{4.54}$$

When the $T1$ shift is not combined with the Td shift, the $T1$ shift is scheduled on Monday for the whole week. Therefore, no shortages can occur during Tuesday till Friday in this case. This is done with constraint (4.55).

$$d_{T1,d} = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{2,3,4,5\}, COMB_{\lceil \frac{d}{7} \rceil} = 0 \tag{4.55}$$

The organ removal shift is scheduled during the fixed switching moments on Friday, Monday and Wednesday every other week. Therefore, no shortages can occur during the other days. This is done with constraint (4.56).

$$d_{M,d} = 0 \qquad \forall d \quad \text{st} \quad d \bmod 14 \notin \{1 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW), 5 + 7 \cdot MW\} \tag{4.56}$$

The Td, Ch and Hpb tasks are scheduled during weekdays. Therefore, no shortages can occur during the weekends. This is done with constraint (4.57).

$$d_{t,d} = 0 \qquad \forall t, d \quad \text{st} \quad t \in \{Td, Ch, Hpb\}, d \bmod 7 \in \{0, 6\} \tag{4.57}$$

The outpatient clinic is scheduled on Monday. During all other days no shortage can occur at the outpatient clinic. This is done with constraint (4.58).

$$d_{OC,d} = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{0, 2, 3, 4, 5, 6\} \tag{4.58}$$

The operating room is scheduled on Wednesday, Thursday and Friday. During all other days no shortage can occur. This is done with constraint (4.59).

$$d_{OR,d} = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{0, 1, 2, 6\} \tag{4.59}$$

During a public holiday only the $T1$, $T2$ and M shifts must be scheduled, all other tasks not. Therefore, no shortages can occur at tasks not equal to $T1$, $T2$ or M during a public holiday. This is done with constraint (4.60).

$$d_{t,d} = 0 \qquad \forall t, d \quad \text{st} \quad t \in \{Td, Ch, Hpb, OC, OR\}, PH_d = 1 \tag{4.60}$$

The dummy variable used for the constraint that one of the surgeons scheduled for the M shift must be MOD certified is only used during starting moments of the M shift. For all other days, this dummy variable is equal to zero. This is done with constraint (4.61).

$$dm_d = 0 \qquad \forall d \quad \text{st} \quad d \bmod 14 \notin \{5 + 7 \cdot MW, 1 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW)\} \tag{4.61}$$

4.3.6 Extend duration

Some tasks can be scheduled longer for the same surgeon, but preferably not. The variables that indicate if the task is longer scheduled at the same surgeon cannot always be equal to one. This is explained below.

Monday till Friday M shift

The start moments of the M shift on Monday and Wednesday can only be scheduled at the same surgeon if the M shift must be scheduled during that week. When this is not the case, the variable that indicates if both start moments are scheduled at the same surgeon is equal to zero. This is done with constraint (4.62).

$$lm_w = 0 \qquad \forall w \quad \text{st} \quad w \bmod 2 = 1 - MW \tag{4.62}$$

Four days $T2$ shift

On Wednesday no block of four consecutive days $T2$ can be scheduled because the whole weekend is scheduled when the $T2$ shift is scheduled on Saturday. Also, the $T2$ shift cannot start on Sunday because of this. Therefore, no block of four consecutive days $T2$ shift can start on Sunday and Wednesday. This is done with constraint (4.63).

$$fd_d = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{0, 3\} \tag{4.63}$$

4.4 Constraints

This section shows the constraints of the model. First, Section 4.4.1 shows the generic constraints. Then, Section 4.4.2 shows the boundary constraints.

4.4.1 Generic constraints

In this section the generic constraints are described. This is done per condition and we will start with the MOD certification.

MOD certified
One of the two surgeons scheduled at the organ removal shifts at the same time must be MOD certified. A dummy variable is added to ensure the model is feasible, also when it is not possible to schedule an MOD certified surgeon. This is done with constraint (4.64).

$$\sum_s (MOD_s + MODC_{s,d}) \cdot x_{s,M,d} \geq 1 - dm_d \qquad \forall d \quad \text{st} \quad d \bmod 14 \in \{7 \cdot MW + 5, 7 \cdot (1 - MW) + 1, 7 \cdot (1 - MW) + 3\} \tag{4.64}$$

Junior surgeon
At most one of the surgeons scheduled at a transplant shifts at the same time is a junior surgeon. This is done with the constraints shown below. We will first look at the overlap between the $T1$ and $T2$ shift during weekends, then the overlap between the $T1$ shift and the $T2$ shift during weekdays. Last, we will look at the overlap between the $T2$ shift and the Td shift. There are no other overlaps between the transplant shifts.

- **Overlap $T1$ and $T2$ during weekend**
When the $T1$ shift is not combined with the Td shift, the weekend $T1$ shift starts on Saturday 8:00 0'clock. The weekend $T2$ shift starts on Friday 22:00 o'clock. Therefore, both weekend shifts overlap. This is done with constraint (4.65). Also, the weekend $T2$ shift overlaps with the week $T1$ shift. This is done with constraint (4.66). The weekend $T1$ shift overlaps with the Monday $T2$ shift. The weekend $T1$ shift ends on Monday 8:00 0'clock and the Monday $T2$ shift starts on Sunday 22:00 o'clock. This is done with constraint (4.67).

When the $T1$ shift is combined with the Td shift during the week before the weekend shift, the $T1$ weekend shift start on Friday 22:00 o'clock. Therefore, the weekend $T1$ and $T2$ shift overlap also in this case. This is done with constraint (4.65). However, the weekend $T2$ shift does not overlap with the week $T1$ shift in this case. Therefore, constraint (4.66) must only hold when the $T1$ shift is not combined with the Td shift during the week before the schedule. When the $T1$ shift is combined with the Td shift the week after the weekend shift, the weekend $T1$ shift ends at Sunday 22:00 o'clock. Therefore, the weekend $T1$ shift does not overlap with the Monday $T2$ shift in this case and constraint (4.67) must only hold when the $T1$ and Td shift are not combined the week after the weekend shift.

$$\sum_s (1 - JR_s) \cdot (x_{s,T1,d} + x_{s,T2,d}) \geq 1 - ds_d \qquad \forall d \quad \text{st} \quad d \bmod 7 = 6 \tag{4.65}$$

$$\sum_s (1 - JR_s) \cdot (x_{s,T1,d-5} + x_{s,T2,d}) \geq 1 - ds_d \qquad \forall d \quad \text{st} \quad d \bmod 7 = 6, \; COMB_{\lceil \frac{d}{7} \rceil} = 0 \tag{4.66}$$

$$\sum_s (1 - JR_s) \cdot (x_{s,T1,d} + x_{s,T2,d+2}) \geq 1 - ds_{d+2} \qquad \forall d \quad \text{st} \quad d \bmod 7 = 6 \; COMB_{\lceil \frac{d+2}{7} \rceil} = 0 \tag{4.67}$$

- **Overlap $T1$ and $T2$ during weekdays**
When the $T1$ shift is not combined with the Td shift, the whole week the same surgeon is scheduled at the $T1$ shift. The whole week $T1$ shift overlaps with the $T2$ shift on Monday, Tuesday, Wednesday, Thursday and Friday. This is done with constraint (4.68).

$$\sum_s (1 - JR_s) \cdot (x_{s,T1,d} + x_{s,T2,d'}) \geq 1 - ds_{d'} \qquad \forall d, d' \quad \text{st} \quad d \bmod 7 = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0, d' \in \{d, d+1, d+2, d+3, d+4\} \tag{4.68}$$

When the $T1$ shift is combined with the Td shift, the $T1$ shift has the same switching moments during weekdays as the $T2$ shift. Therefore, the $T1$ shift overlaps the whole shift with the $T2$ shift scheduled at the same day. This is done with constraint (4.69).

$$\sum_s (1 - JR_s) \cdot (x_{s,T1,d} + x_{s,T2,d}) \geq 1 - ds_d \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{1, 2, 3, 4, 5\}, \; COMB_{\lceil \frac{d}{7} \rceil} = 1 \tag{4.69}$$

- **Overlap $T2$ and Td during weekdays**
  Also, the $T2$ and Td have an overlap during weekdays. Also, in this case, at most one of the surgeons is a junior surgeon. The $T2$ shift overlaps with the Td shift scheduled at the same day. This is done with constraint (4.70). This constraint must only hold when the $T1$ is not combined with the Td shift, otherwise this constraint is already included in the overlap between the $T1$ and $T2$ shift.

$$\sum_s (1 - JR_s) \cdot (x_{s,Td,d} + x_{s,T2,d}) \geq 1 - ds_d \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5\}, \quad COMB_{\lceil \frac{d}{7} \rceil} = 0 \tag{4.70}$$

Hpb surgeon

Every week at least one hpb surgeon must be scheduled at the operating room. This is done with constraint (4.128). When it is not possible to make a schedule by fulfilling this constrain, we still want to make a schedule. Therefore, a dummy variable is used.

$$\sum_{d'=d}^{d+2} \sum_s HPB_s \cdot x_{s,OR,d'} \geq 1 - dhpb_{\lceil \frac{d}{7} \rceil} \qquad \forall d \quad \text{st} \quad d \bmod 7 = 3 \tag{4.71}$$

Schedule tasks

We want that the tasks are scheduled when this is needed. Therefore constraints (4.72) — (4.78) are used. Constraint (4.72) is used to ensure one surgeon is scheduled during week days at the tasks Td, Ch and Hpb. When it is not possible to schedule these tasks, a dummy variable is used. These tasks should only be scheduled when the weekday is no public holiday. The same thing is done with constraints (4.73) and (4.74) for the $T1$ shift. Constraint (4.74) must only hold when the $T1$ shift is combined with the Td shift. This is because the $T1$ shift is scheduled for the whole week at the same surgeon when the $T1$ shift is not combined with the Td shift. Constraint (4.75) is for the $T2$ shift. At this shift always one surgeon should be scheduled, except for the Sunday. Constraint (4.76) is used for the M shift. At the M shift always two surgeons must be scheduled at the starting moments. Constraint (4.77) is used for the outpatient clinic and constraint (4.78) for the operating room. For the outpatient clinic and operating room two surgeons should be scheduled when this task should be scheduled.

$$\sum_s x_{s,t,d} + d_{t,d} = 1 \qquad \forall t,d \quad \text{st} \quad t \in \{Td, Ch, Hpb\}, d \bmod 7 \in \{1,2,3,4,5\}, \quad PH_d = 0 \tag{4.72}$$

$$\sum_s x_{s,T1,d} + d_{T1,d} = 1 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{1,6\} \tag{4.73}$$

$$\sum_s x_{s,T1,d} + d_{T1,d} = 1 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{2,3,4,5\}, COMB_{\lceil \frac{d}{7} \rceil} = 1 \tag{4.74}$$

$$\sum_s x_{s,T2,d} + d_{T2,d} = 1 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5,6\} \tag{4.75}$$

$$\sum_s x_{s,M,d} + d_{M,d} = 2 \qquad \forall d \quad \text{st} \quad d \bmod 14 \in \{5 + 7 \cdot MW, 7 \cdot (1 - MW) + 1, 7 \cdot (1 - MW) + 3\} \tag{4.76}$$

$$\sum_s x_{s,OC,d} + d_{OC,d} = 2 \qquad \forall d \quad \text{st} \quad d \bmod 7 = 1, PH_d = 0 \tag{4.77}$$

$$\sum_s x_{s,OR,d} + d_{OR,d} = 2 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{3,4,5\}, PH_d = 0 \tag{4.78}$$

$T1$ shift

When the $T1$ shift is not combined with the Td shift, the $T1$ shift has a fixed structure. This fixed structure is created with the constraints shown below.

- **No other tasks scheduled**
  When the $T1$ shift is scheduled no other tasks can be scheduled during that day. This is done with constraint (4.79).

$$x_{s,t,d'} \leq 1 - x_{s,T1,d} \qquad \forall s,t,d,d' \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 0, t \neq T1, d \bmod 7 = 1, \quad d' \in \{d, d+1, d+2, d+3, d+4\} \tag{4.79}$$

- **Weekend shift before week $T1$ shift**
When a surgeon is scheduled at the week $T1$ shift, the weekend before this week no weekend $T1$ shift can be scheduled. This is done with constraint (4.82). It is possible to schedule a weekend $T2$ shift before the week $T1$ shift, but preferably not. This is done with constraint (4.80).

$$x_{s,T2,d} \leq 1 - x_{s,T1,d+2} + wk_{\lceil \frac{d+2}{7} \rceil} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d+2}{7} \rceil} = 0, \; d \bmod 7 = 6 \tag{4.80}$$

- **No weekend shift after week $T1$ shift**
When a surgeon has a week $T1$ shift, the weekend after this week $T1$ no shifts can be scheduled. The weekend M shift is not scheduled, because this shift starts on Friday. For the $T1$ and $T2$ shift during the weekend constraint (4.81) is used.

$$\sum_{t \in \{T1,T2\}} x_{s,t,d+5} + x_{s,T1,d} \leq 1 \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 0, d \bmod 7 = 1 \tag{4.81}$$

- **No task on Monday after weekend shift $T1$**
When a surgeon has a weekend shift $T1$, the Monday after this shift no task can be scheduled. This is done with constraint (4.82).

$$\sum_{t} x_{s,t,d+2} + 5 \cdot x_{s,T1,d} \leq 5 \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d+2}{7} \rceil} = 0, d \bmod 7 = 6 \tag{4.82}$$

- **No two weeks consecutive $T1$ shift**
No two weeks consecutive $T1$ shift can be scheduled. Therefore, only one of two consecutive weeks the $T1$ shift can be scheduled at the same surgeon. This is done with constraint (4.83).

$$x_{s,T1,d} + x_{s,T1,d+7} \leq 1 \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 0, COMB_{\lceil \frac{d+7}{7} \rceil} = 0, d \bmod 7 = 1 \tag{4.83}$$

No three weeks consecutive weekend shift

A surgeon should not be scheduled at a weekend shift for three weekends consecutively. This is done with constraint (4.84).

$$x_{s,M,d-1} + x_{s,M,d+6} + x_{s,M,d+13} + \sum_{t \in \{T1,T2\}} (x_{s,t,d} + x_{s,t,d+7} + x_{s,t,d+14}) \leq 2 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 6 \tag{4.84}$$

Combine tasks

It is desirable that each surgeon has at most one task during a day, but it is possible to combine some of the tasks. Not all tasks can be combined. Constraint (4.85) ensures no tasks can be combined with the outpatient clinic tasks and constraint (4.86) ensures that the operating room tasks can only can be combined with the $T2$ shift. Other constraints ensure the $T1$ shift is only combined with the Td shift during the vacation period (constraints (4.110) and (4.111)). Also, other constraints ensure that the organ removal shift is not combined with other tasks except for the Hpb task on Friday when the organ removal week starts (constraints (4.103), (4.106) and (4.109)).

$$x_{s,t,d} \leq 1 - x_{s,OC,d} \qquad \forall s,t,d \quad \text{st} \quad d \bmod 7 = 1, t \neq OC \tag{4.85}$$

$$x_{s,t,d} \leq 1 - x_{s,OR,d} \qquad \forall s,t,d \quad \text{st} \quad d \bmod 7 \in \{3,4,5\}, t \notin \{T2, OR\} \tag{4.86}$$

The other tasks can be combined, but it is not preferred. Therefore, we want to know when the tasks are combined. With constraint (4.87) the Hpb and $T2$ tasks can be combined, with constraint (4.88) the M and Hpb task can be combined, with constraint (4.90) one surgeon can do both OR and $T2$, and with constraint (4.91) one surgeon can do both Ch and Hpb. The Hpb task only can be combined with the M shift on Wednesday if the switching moment on Wednesday is used. This is done with constraint (4.89). Constraints (4.92) and (4.93) ensure only one of the combination possibilities is used per day, except for the combinations OR and $T2$ for one surgeon and Ch and Hpb for another surgeon and the combinations Hpb and M for one surgeon and $T2$ and OR for another surgeon. Constraint (4.94) ensures that the chef tasks, Td shift and $T2$ shift can be combined to decrease the number of shortages, but not the combination Hpb with Ch or with $T2$.

These combinations can be done with constraints (4.87) and (4.91).

$$x_{s,Hpb,d} + x_{s,T2,d} \leq 1 + y_d \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5\} \tag{4.87}$$

$$x_{s,Hpb,d} + x_{s,M,d} \leq 1 + w_d \qquad \forall s,d \quad \text{st} \quad d \bmod 14 \in \{7 \cdot MW + 5, 7 \cdot (1 - MW) + 1, 7 \cdot (1 - MW) + 3\} \tag{4.88}$$

$$w_d \leq 1 - lm_{\lceil \frac{d}{7} \rceil} \qquad \forall d \quad \text{st} \quad d \bmod 14 = 7 \cdot (1 - MW) + 3 \tag{4.89}$$

$$x_{s,OR,d} + x_{s,T2,d} \leq 1 + z_d \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{3,4,5\} \tag{4.90}$$

$$x_{s,Ch,d} + x_{s,Hpb,d} \leq 1 + u_d \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5\} \tag{4.91}$$

$$y_d + z_d \leq 1 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{3,4,5\} \tag{4.92}$$

$$y_d + w_d + u_d \leq 1 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5\} \tag{4.93}$$

$$x_{s,t,d} + x_{s,t',d} \leq 1 + mc_{d,t,t'} \qquad \forall s,t,t',d \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5\}, t,t' \in \{T2, Td, Ch, Hpb\}, t' \prec t, \neg(t \in \{T2, Ch\}, t' = Hpb) \tag{4.94}$$

Initial conditions

When some of the tasks are already scheduled before the rest of the schedule is made, constraint (4.95) ensures these tasks are scheduled at these moments for the corresponding surgeon.

$$x_{s,t,d} = X_{s,t,d} \qquad \forall s,t,d \quad \text{st} \quad X_{s,t,d} = 1, d \bmod 7 \neq 0 \tag{4.95}$$

Minimal number of days $T2$ shift

The $T2$ shift must be scheduled for at least two consecutive days. This is done with constraint (4.96) when a surgeon starts with the $T2$ shift on Monday and with constraint (4.97) when a surgeon starts with the $T2$ shift at another day. When a surgeon starts on Saturday, the surgeon is scheduled the whole weekend on the $T2$ shift. Therefore, no constraint is needed for the start on Saturday.

$$x_{s,T2,d+1} \geq x_{s,T2,d} - x_{s,T2,d-2} \qquad \forall d \quad \text{st} \quad d \bmod 7 = 1 \tag{4.96}$$

$$x_{s,T2,d+1} \geq x_{s,T2,d} - x_{s,T2,d-1} \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{2,3,4,5\} \tag{4.97}$$

Maximum number of days $T2$ shift

The $T2$ shift can be scheduled at most four days consecutively. We want to indicate when the $T2$ shift is scheduled four days consecutively, because we want to minimize the number of times the $T2$ shift is scheduled four days consecutively. This is done with constraints (4.98) – (4.101). Four constraints are needed because on Saturday the shift is scheduled for the whole weekend. When the $T2$ shift is scheduled for the first day on Monday or Tuesday, constraint (4.98) is used to indicate when four consecutive days $T2$ are scheduled. When the shift is scheduled at Wednesday, the shifts can be scheduled till Saturday and not on Sunday. Therefore, the whole weekend shift cannot be scheduled, and it is not possible to schedule the $T2$ shift four days consecutively starting on Wednesday. This is done with constraint (4.99). When the $T2$ shift starts at Thursday, Friday or Saturday, the Sunday is also a day the shift can be scheduled. Therefore, constraints (4.100) and (4.101) are needed. Constraint (4.102) ensures that after four consecutive days $T2$ shift the next day no $T2$ shift is scheduled for this surgeon.

$$\sum_{d'=d}^{d+3} x_{s,T2,d'} \leq 3 + fd_d \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,2\} \tag{4.98}$$

$$\sum_{d'=d}^{d+3} x_{s,T2,d'} \leq 3 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 3 \tag{4.99}$$

$$\sum_{d'=d}^{d+2} x_{s,T2,d'} \leq 2 + fd_d \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 4 \tag{4.100}$$

$$\sum_{d'=d}^{d+3} x_{s,T2,d'} \leq 2 + fd_d \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{5,6\} \tag{4.101}$$

$$x_{s,T2,d+4} \leq 2 - fd_d - x_{s,T2,d} \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,2,4,5,6\} \tag{4.102}$$

Scheduling M shift

The structure of the M shift is fixed. Every other week two surgeons start on Friday and are scheduled for this shift till Monday 12:00 o'clock, on Monday 12:00 o'clock two surgeons start and are scheduled till Wednesday 12:00 o'clock and on Wednesday two surgeons start and are scheduled till Friday 12:00 o'clock.

- **Start on Friday**
When a surgeon starts with the M shift on Friday, no other tasks can be scheduled during this Friday, except for the Hpb task. This is done with constraint (4.103). The Hpb task can be combined with the M shift on Friday. This is done with constraint (4.88).

$$x_{s,t,d} \leq 1 - x_{s,M,d} \qquad \forall s,t,d \quad \text{st} \quad t \in \{T2, Td, OR, Ch\}, \\ d \bmod 14 = 5 + 7 \cdot MW \tag{4.103}$$

When a surgeon starts with the M shift on Friday, this surgeon is also scheduled for the M shift during the weekend. Therefore, no other tasks can be scheduled during the weekend. This is done with constraint (4.104).

$$x_{s,t,d+1} \leq 1 - x_{s,M,d} \qquad \forall s,t,d \quad \text{st} \quad t \in \{T1, T2\}, d \bmod 14 = 5 + 7 \cdot MW \tag{4.104}$$

No task can be scheduled on Monday when the surgeon starts on Friday before with the M shift. This is because the M shift is scheduled till Monday 12:00 o'clock in this case and the rest of the Monday is compensation for the M shift. This is done with constraint (4.105).

$$x_{s,t,d+3} \leq 1 - x_{s,M,d} \qquad \forall s,t,d \quad \text{st} \quad d \bmod 14 = 5 + 7 \cdot MW \tag{4.105}$$

- **Start on Monday**
When a surgeon starts with the M shift on Monday, no other tasks can be scheduled on Monday and Tuesday. This is done with constraint (4.106).

$$x_{s,t,d'} \leq 1 - x_{s,M,d} \qquad \forall s,t,d,d' \quad \text{st} \quad t \in \{T1, T2, Td, OC, Ch, Hpb\}, \\ d \bmod 14 = 1 + 7 \cdot (1 - MW), d' \in \{d, d+1\} \tag{4.106}$$

No task, except the M shift, can be scheduled on Wednesday when a surgeon starts with the M shift on Monday. The surgeon is scheduled on the M shift till 12:00 o'clock and the rest of the day is compensation for the M shift. This is done with constraint (4.107).

$$x_{s,t,d+2} \leq 1 - x_{s,M,d} \qquad \forall s,t,d \quad \text{st} \quad t \neq M, d \bmod 14 = 1 + 7 \cdot (1 - MW) \tag{4.107}$$

The M shift can be scheduled on Wednesday, but preferably not. This is done with constraint (4.108). By doing this, only two surgeons need to be scheduled on Wednesday on the M shift. However, the surgeons who are scheduled are scheduled form Monday till Friday. Therefore, these surgeons are scheduled four days on a 24h shift.

$$x_{s,M,d+2} \leq 1 - x_{s,M,d} + lm_{\lceil \frac{d}{7} \rceil} \qquad \forall s,d \quad \text{st} \quad d \bmod 14 = 1 + 7 \cdot (1 - MW) \tag{4.108}$$

- **Start on Wednesday**
When a surgeon starts with the M shift on Wednesday, no other tasks can be scheduled on Wednesday, Thursday, and Friday. This is done with constraint (4.109).

$$x_{s,t,d'} \leq 1 - x_{s,M,d} \qquad \forall s,t,d,d' \quad \text{st} \quad t \in \{T2, Td, OR, Ch, Hpb\}, \\ d \bmod 14 = 3 + 7 \cdot (1 - MW), d' \in \{d, d+1, d+2\} \tag{4.109}$$

Combine $T1$ and Td shift

During the vacation period, it can be decided to combine the $T1$ and Td shift. When the two shifts are combined, this combination is scheduled in the same way as the $T2$ shift. The switch moments are the same, it is scheduled for at least two consecutive days and for at most three consecutive days. This is one day shorter than the $T2$ shift, because the $T1$ and Td shift combined is busier than the $T2$ shift. The constraints used for this combination are shown below.

- **Schedule $T1$ and Td and no other tasks**
When we combine the $T1$ and Td shift, the Td shift is also scheduled when the $T1$ shift is scheduled during weekdays. This is done with constraint (4.110). This constraint only holds when the day is not a public holiday, because during a public holiday only the $T1$ shift is scheduled and not the Td shift. When the $T1$ shift is scheduled no other tasks than the Td shift can be scheduled. This is done with constraint (4.111).

$$x_{s,Td,d} = x_{s,T1,d} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 1, PH_d = 0, d \bmod 7 \neq 6 \tag{4.110}$$

$$x_{s,t,d} \leq 1 - x_{s,T1,d} \qquad \forall s,t,d \quad \text{st} \quad t \notin \{T1, Td\}, COMB_{\lceil \frac{d}{7} \rceil} = 1 \tag{4.111}$$

- **Not scheduled after $T2$ shift**
  The $T1$, Td combination cannot be scheduled the day after the $T2$ shift ends. When this was possible, a surgeon could be scheduled for seven consecutive days on a 24h shift. This is done with constraint (4.112) for all days except Monday and with constraint (4.113) for Monday.

$$x_{s,T1,d+1} \leq 1 - x_{s,T2,d} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d+1}{7} \rceil} = 1, \quad (d+1) \bmod 7 \neq 1 \tag{4.112}$$

$$x_{s,T1,d+2} \leq 1 - x_{s,T2,d} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d+2}{7} \rceil} = 1, \quad (d+2) \bmod 7 = 1 \tag{4.113}$$

- **No $T2$ shift after the combination**
  The day after the combination $T1$, Td is scheduled, the $T2$ shift cannot be scheduled. When this was possible, a surgeon could be scheduled for seven consecutive days on a 24h shift. This is done with constraint (4.114) for all days except for Monday and Saturday and with constraint (4.115) for Monday. When the shift is scheduled on Saturday, also the Sunday is scheduled. Therefore, two days are scheduled.

$$x_{s,T2,d+1} \leq 1 - x_{s,T1,d} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 1, \quad (d+1) \bmod 7 \neq 1 \tag{4.114}$$

$$x_{s,T2,d+2} \leq 1 - x_{s,T1,d} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 1, \quad (d+2) \bmod 7 = 1 \tag{4.115}$$

- **At least two days**
  When the $T1$, Td combination is scheduled for some surgeon, this is scheduled for at least two consecutive days. This is done with constraint (4.116) for all days except for Monday and with constraint (4.117) for Monday. On Saturday the shift is scheduled for the whole weekend. Therefore, no constraint is needed for the Saturday.

$$x_{s,T1,d+1} \geq x_{s,T1,d} - x_{s,T1,d-1} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 1, \quad d \bmod 7 \in \{2,3,4,5\} \tag{4.116}$$

$$x_{s,T1,d+1} \geq x_{s,T1,d} - x_{s,T1,d-2} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 1, \quad d \bmod 7 = 1 \tag{4.117}$$

- **At most three days**
  When the $T1$, Td combination is scheduled, the combination is scheduled for at most three consecutive days. This is done with constraints (4.118), (4.119) and (4.120). Different constraints are needed because on Saturday the shift is scheduled for the whole weekend. When the $T1$, Td combination is scheduled for the first day on Monday, Tuesday or Wednesday constraint (4.118) is used. When the $T1$, Td combination starts at Thursday, Friday or Saturday, the Sunday is also a day the shift can be scheduled. Therefore, the constraints (4.119) and (4.120) are needed.

$$\sum_{d'=d}^{d+3} x_{s,T1,d'} \leq 3 \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 1, d \bmod 7 \in \{1,2,3\} \tag{4.118}$$

$$\sum_{d'=d}^{d+2} x_{s,T1,d'} \leq 2 \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 1, d \bmod 7 = 4 \tag{4.119}$$

$$\sum_{d'=d}^{d+3} x_{s,T1,d'} \leq 2 \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d+3}{7} \rceil} = 1, d \bmod 7 \in \{5,6\} \tag{4.120}$$

Determine maximum number of days scheduled at a task

The differences between the number of days scheduled at a task per surgeon are minimized, not for all tasks but for the $T1$, $T2$, M and OR tasks, To do this the maximum number of days a surgeon is scheduled at a task is minimized. With constraint (4.121) this value is determined for the M shift. When a surgeon starts on Friday with the M shift, this surgeon is scheduled for three days at the M shift. When a surgeon is scheduled on Monday or Wednesday on the M shift, this surgeon is scheduled at the M shift for two days.

For the M shift only the surgeons who can be scheduled at the transplant shifts are taken into account. This is done, because the other surgeons can do more organ removal shifts to compensate the fact that they cannot be scheduled at the transplant shifts.

$$\begin{aligned}&3 \cdot \sum_{d: d \bmod 14 = 5+7\cdot MW} x_{s,M,d} + 2 \cdot \sum_{d: d \bmod 14 = 1+7\cdot(1-MW)} x_{s,M,d} + \\ &2 \cdot \sum_{d: d \bmod 14 = 3+7\cdot(1-MW)} x_{s,M,d} + DT_{s,M} \leq mds_M \qquad \forall s \quad \text{st} \sum_{t \in \{T1,T2,M\}} TS_{t,s} \neq TS_{M,s} \quad (4.121)\end{aligned}$$

The same thing is done with constraint (4.122) for the $T2$ shift. When the $T2$ shift is scheduled at some day, except the Saturday, this shift is scheduled for one day at this surgeon. When a surgeon is scheduled at Saturday on the $T2$ shift, this surgeon is scheduled for two days on the $T2$ shift.

$$\sum_{d: d \bmod 7 \neq 6} x_{s,T2,d} + 2 \cdot \sum_{d: d \bmod 7 = 6} x_{s,T2,d} + DT_{s,T2} \leq mds_{T2} \qquad \forall s \qquad (4.122)$$

The maximum number of days a surgeon is scheduled at the $T1$ shift is determined with constraint (4.123). When the surgeon is scheduled at Tuesday, Wednesday or Thursday at the $T1$ shift, this surgeon is scheduled for one day at this shift. When the surgeon is scheduled on Monday on the $T1$ shift and the $T1$ shift is not combined with the Td shift, this surgeon is scheduled for five days at this shift. When the $T1$ shift is combined with the Td shift, the surgeon is scheduled for one day at the $T1$ shift if the surgeon is scheduled on Monday at the $T1$ shift. When the surgeon is scheduled on Saturday on the $T1$ shift, this surgeon is scheduled the whole weekend at this shift. Therefore, a surgeon is scheduled two days at the $T1$ shift when a surgeon is scheduled on Saturday at the $T1$ shift.

$$\begin{aligned}&\sum_{d: d \bmod 7 \notin \{1,6\}} x_{s,T1,d} + 5 \cdot \sum_{d: d \bmod 7 = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0} x_{s,T1,d} + \sum_{d: d \bmod 7 = 1, COMB_{\lceil \frac{d}{7} \rceil} = 1} x_{s,T1,d} \\ &+ 2 \cdot \sum_{d: d \bmod 7 = 6} x_{s,T1,d} + DT_{s,T1} \leq mds_{T1} \qquad \forall s \qquad (4.123)\end{aligned}$$

The same thing is done with constraint (4.124) for the OR task. When a surgeon is scheduled at the OR task, this surgeon is always scheduled for one day.

$$\sum_{d} x_{s,OR,d} + DT_{s,OR} \leq mds_{OR} \qquad \forall s \qquad (4.124)$$

Determine maximum number of days scheduled

The scheduled days must be equally divided over the surgeons. Therefore, every surgeon is approximately scheduled at a task the same number of days, but also the number of times more tasks are scheduled at the same day for a surgeon. To do this the maximum number of tasks scheduled for one surgeon is determined. We also take into account when a task is scheduled for more days at once. Therefore, a tasks scheduled at Saturday counts for two tasks scheduled etcetera. This is done with constraint (4.125). By doing this, the workload is equally divided over the surgeons.

$$\begin{aligned}&3 \cdot \sum_{d: d \bmod 14 = 5+7\cdot MW} x_{s,M,d} + 2 \cdot \sum_{d: d \bmod 14 = 1+7\cdot(1-MW)} x_{s,M,d} + 2 \cdot \sum_{d: d \bmod 14 = 3+7\cdot(1-MW)} x_{s,M,d} \\ &+ 2 \cdot \sum_{d: d \bmod 7 = 6} \sum_{t \in \{T1,T2\}} x_{s,t,d} + \sum_{d: d \bmod 7 \neq 6} \sum_{t \notin \{M,T1\}} x_{s,t,d} + \sum_{d: d \bmod 7 \notin \{1,6\}} x_{s,T1,d} \\ &+ 5 \cdot \sum_{d: d \bmod 7 = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0} x_{s,T1,d} + \sum_{d: d \bmod 7 = 1, COMB_{\lceil \frac{d}{7} \rceil} = 1} x_{s,T1,d} + 2 \cdot \sum_{d: d \bmod 7 = 6} x_{s,T1,d} \\ &+ \sum_{t} DT_{s,t} \leq md \qquad \forall s \qquad (4.125)\end{aligned}$$

Only M, OC and OR tasks or only M shift

When a surgeon can only be scheduled at M, OC and OR tasks, the organ removal shift can only be scheduled when the parameter $X_{s,M,d}$ is equal to one. The same thing holds for surgeons who can only be scheduled at the M shift. This is done with constraint (4.126).

$$\begin{aligned}x_{s,M,d} = X_{s,M,d} \qquad \forall s,d \quad \text{st} \sum_{t} TS_{t,s} = \sum_{t \in \{M,OR,OC\}} TS_{t,s}, d \bmod 14 \in \{5 + 7 \cdot WM, \\ 1 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW)\} \qquad (4.126)\end{aligned}$$

No transplant shifts

Some surgeons cannot be scheduled at the transplant shifts. When no transplant shifts can be scheduled, at most one-fifth of the days an organ removal shift can be scheduled per quarter. This is done with constraint (4.127).

$$\begin{aligned} &3 \cdot \sum_{d: d \bmod 14 = 5+7\cdot MW} x_{s,M,d} + 2 \cdot \sum_{d: d \bmod 14 = 1+7\cdot(1-MW)} x_{s,M,d} + \\ &2 \cdot \sum_{d: d \bmod 14 = 3+7\cdot(1-MW)} x_{s,M,d} \leq \frac{|D|}{5} \qquad \forall s \quad \text{st} \quad \sum_{t \in \{T1,T2,M\}} TS_{t,s} = TS_{M,s} \end{aligned} \tag{4.127}$$

Scheduling Hpb task

The number of different surgeons scheduled at the Hpb task during a week is minimized. To determine the number of surgeons scheduled at the Hpb task during a week, constraint (4.128) is used.

$$\sum_{d=7\cdot(w-1)+1}^{7\cdot w} x_{s,Hpb,d} \leq 5 \cdot ih_{s,w} \qquad \forall s,w \quad \text{st} \quad TS_{s,Hpb} = 1 \tag{4.128}$$

Scheduling Ch task

The number of different surgeons scheduled at the Ch task during a week is minimized. To determine the number of surgeons scheduled at the Ch task during a week constraint (4.129) is used.

$$\sum_{d=7\cdot(w-1)+1}^{7\cdot w} x_{s,Ch,d} \leq 5 \cdot ic_{s,w} \qquad \forall s,w \quad \text{st} \quad TS_{s,Ch} = 1 \tag{4.129}$$

More combinations

When more combinations are made than the combinations Ch/Hpb, M/Hpb, $T2/Hpb$ or $T2/OR$ we want to know this, because extra costs belong to this case. This case is indicated with constraint (4.130)

$$\begin{aligned} mcd_d \geq mc_{d,t,t'} \qquad &\forall d,t,t' \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5\}, t \prec t' \\ &t,t' \in \{T2, Td, Ch, Hpb\} \end{aligned} \tag{4.130}$$

Week $T1$ shift

Some surgeons cannot handle the week $T1$ shift well, because this is the night shift for the whole week. Therefore, these surgeons have to do fewer night shifts during the week and are scheduled for at most one week the whole week at the $T1$ shift. This is done with constraint (4.131). However, we want to distribute the shifts equally over the surgeons. Therefore, these surgeons have to do more weekend $T1$ shifts.

$$\sum_{d: d \bmod 7 = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0} x_{s,T1,d} \leq 1 \qquad \forall s \quad \text{st} \quad FWT1_s = 1 \tag{4.131}$$

Shifts during Christmas

During Christmas only shifts that are done during weekends are scheduled. However, during Christmas we have extra constraints. The $T1$ or $T2$ shift cannot be done by the same surgeon the two Christmas days. This is done with constraint (4.132) for the $T2$ shift. For the $T1$ shift holds that during Christmas the whole week $T1$ and Td are combined. Constraint (4.133) ensures that two different surgeons are scheduled at the $T1$ shift during Christmas.

$$x_{s,T2,d} + x_{s,T2,d+1} \leq 1 \qquad \forall s,d \quad \text{st} \quad CHRISTMAS_d = 1, CHRISTMAS_{d+1} = 1 \tag{4.132}$$

$$x_{s,T1,d} + x_{s,T1,d+1} \leq 1 \qquad \forall s,d \quad \text{st} \quad CHRISTMAS_d = 1, CHRISTMAS_{d+1} = 1 \tag{4.133}$$

4.4.2 Boundary conditions

To ensure the conditions also hold at the boundary, the beginning of the schedule, the constraints shown below are used. For this, some extra parameters are used. These parameters contain information of the shifts done one week or two weeks before the start date of the new schedule.

No task on Monday after weekend $T1$ shift

When the last weekend before the new schedule a $T1$ shift is done, the first Monday of the new schedule no tasks can be scheduled when the $T1$ and Td shifts are not combined. This is done with constraint (4.134).

$$\sum_t x_{s,t,1} \leq 5 - 5 \cdot TW_s \qquad \forall s \quad \text{st} \quad COMB_1 = 0 \tag{4.134}$$

No two weeks consecutive $T1$ shift

When the last week before the new schedule a $T1$ shift is done by a surgeon the whole week, the first week of the new schedule the $T1$ shift cannot be scheduled if the $T1$ and Td shifts are not combined. This is done with constraint (4.135).

$$x_{s,T1,1} \leq 1 - TWB_s \qquad \forall s \quad \text{st} \quad COMB_1 = 0 \tag{4.135}$$

No three weeks consecutive weekend shift

To ensure no three weeks consecutive weekend shifts are scheduled at the beginning of the schedule, the weekend before the start date of the schedule must be considered. Constraint (4.136) considers two weekends before the start date of the schedule and constraint (4.137) only the last weekend before the start date.

$$x_{s,M,5} + \sum_{t \in \{T1,T2\}} x_{s,t,6} \leq 2 - WB_s - WB_s \cdot WBT_s \qquad \forall s \tag{4.136}$$

$$x_{s,M,5} + \sum_{t \in \{T1,T2\}} x_{s,t,6} + x_{s,M,12} + \sum_{t \in \{T1,T2\}} x_{s,t,13} \leq 2 - WB_s \qquad \forall s \tag{4.137}$$

Maximum number of consecutive days $T2$ shift

The $T2$ shift can be scheduled at most four consecutive days for the same surgeon. This must also hold at the boundary. When we know how many consecutive days a surgeon is scheduled at the $T2$ shift before the first day of the new scheduled, we know when four consecutive days are scheduled. When four consecutive days $T2$ shift are scheduled, the first day of the new schedule no $T2$ shift can be scheduled for this surgeon. This is done with constraint (4.138). When three consecutive days $T2$ shift are scheduled, the first day of the schedule the $T2$ shift can be scheduled, but preferably not. This is done with constraint (4.139). When two consecutive days $T2$ shift are scheduled before the first day of the schedule, the second day $T2$ shift can be scheduled. When also the first day of the schedule the $T2$ shift is scheduled for this surgeon, four consecutive days are scheduled. This is done with constraint (4.140).

$$x_{s,T2,1} = 0 \qquad \forall s \quad \text{st} \quad ND_s = 4 \tag{4.138}$$

$$x_{s,T2,1} \leq fdm \qquad \forall s \quad \text{st} \quad ND_s = 3 \tag{4.139}$$

$$x_{s,T2,2} \leq 1 - x_{s,T2,1} + fdtu \qquad \forall s \quad \text{st} \quad ND_s = 2 \tag{4.140}$$

Maximum number of consecutive days $T1$ shift

When the $T1$ shift is combined with the Td shift, this combination can be scheduled maximally three consecutive days. This is done with constraints (4.141) and (4.142).

$$x_{s,T1,1} = 0 \qquad \forall s \quad \text{st} \quad COMB_1 = 1, NDT_s = 3 \tag{4.141}$$

$$x_{s,T1,2} \leq 1 - x_{s,T1,1} \qquad \forall s \quad \text{st} \quad COMB_1 = 1, NDT_s = 2 \tag{4.142}$$

Weekend shift before week $T1$ shift

When the $T1$ shift is not combined with the Td shift, the week $T1$ shift cannot be scheduled after a weekend $T1$ shift. This is done with constraint (4.134). It is possible to schedule a weekend $T2$ shift before a week $T1$ shift, but preferably not. This is done with constraint (4.143).

$$x_{s,T1,1} \leq wk_1 \qquad \forall s \quad \text{st} \quad COMB_1 = 0, ND_s > 0 \tag{4.143}$$

First Monday M shift

When a surgeon starts at Friday with the M shift, this surgeon cannot start at Monday with the another task. This surgeon is scheduled on the M shift on Monday. However, this surgeon cannot start with the M shift, because the surgeon already started on Friday with the M shift. This is done with constraint (4.144).

$$x_{s,t,1} \leq 1 - WBM_s \qquad \forall s, t \quad \text{st} \quad MW = 1 \tag{4.144}$$

Junior surgeon

Also at the boundary at most one of the surgeons scheduled at the transplant shifts at the same time is a junior surgeon. An extra constraint is needed at the boundary if the $T1$ shift is not combined with the Td shift. In this case, the $T2$ shift of the first day of the schedule overlaps with the $T1$ weekend shift of the last weekend of the last schedule.

$$\sum_s (1 - JR_s)(TW_s + x_{s,T2,1}) \geq 1 - ds_1 \qquad \text{st} \quad COMB_1 = 0 \tag{4.145}$$

4.5 Objective function

The best schedule is a schedule where all tasks are scheduled and all constraints are fulfilled. When it is not possible to make such a schedule, costs are assigned to the possibility that a constraint does not hold during a schedule. By assigning these costs, we can add which constraints are more important than others. The costs are determined together with the transplant surgeons. They have determined which constraints are more important than others. The costs are the numbers before the variables in the objective function (4.146). How the costs are determined, is described below, but first, we will explain what all parts of the objective function represent. First, the number of shortages is minimized with the first twelve summations. A shortage occurs if fewer surgeons than needed are scheduled at some task, when no MOD certified surgeon is scheduled, when no hpb surgeon is scheduled at the operating room during a week or when at some moment no senior surgeon is scheduled at a transplant shift. Some tasks are more important than other tasks. Therefore, different weights are used for the shortages at different tasks. To reduce the number of shortages, some tasks can be combined. This is done with the thirteenth, fourteenth, fifteenth and sixteenth summation. These summations have lower costs than the shortages, because we prefer combinations over shortages. Then, we also minimize the number of weeks a weekend shift is scheduled before a week $T1$ shift with the summation with costs 300 (when the $T1$ shift is not combined with the Td shift). The summation with costs 200 minimizes the weeks where the M shift is done from Monday till Friday by the same surgeon. The $T2$ shift can also be scheduled longer for the same surgeon if this reduces the number of shortages or combinations. This is done with the summations with costs 100. The next summation (with costs 10) minimizes the number of different surgeons at respectively the Hpb and Ch task during a week. The lasts part of the objective function are the maximum number of days a surgeon is scheduled at the different tasks and the maximum number tasks that are scheduled for one surgeon. By minimizing the maximum number of tasks, the workload is equally divided over the surgeons.

$$\begin{aligned}
\text{minimize} \quad & 210\,000 \cdot \sum_{d:d \bmod 14 = 5+7\cdot MW} (d_{M,d} + dm_d) + 140\,000 \cdot \sum_{\substack{d:d \bmod 14 \in \{3+7\cdot(1-MW),\\ 1+7\cdot(1-MW)\}}} (d_{M,d} + dm_d) \\
& + 80\,000 \cdot \sum_{d:d \bmod 7 = 6} d_{T1,d} + 40\,000 \cdot \Big(\sum_{\substack{d:d \bmod 7 \neq 6,\\ COMB_{\lceil \frac{d}{7} \rceil}=1}} d_{T1,d} + 5 \cdot \sum_{\substack{d:d \bmod 7 \neq 6,\\ COMB_{\lceil \frac{d}{7} \rceil}=0}} d_{T1,d} + \sum_w dhpb_w \\
& + \sum_d ds_d \Big) + 60\,000 \cdot \sum_{d:d \bmod 7 = 6} d_{T2,d} + 30\,000 \cdot \sum_{d:d \bmod 7 \neq 6} d_{T2,d} + 20\,000 \cdot \sum_d d_{Td,d} \\
& + 10\,000 \cdot \sum_d \sum_{t \in \{Ch,Hpb\}} d_{t,d} + 5\,000 \cdot \sum_d \sum_{t \in \{OC,OR\}} d_{t,d} + 3\,000 \cdot \sum_d u_d + 2\,000 \cdot \sum_d z_d \\
& + 2\,000 \cdot \sum_d y_d + 1\,000 \cdot \sum_d w_d + 300 \cdot \sum_w wk_w + 200 \cdot \sum_w lm_w + 100 \cdot \Big(\sum_d fd_d + fdm + fdtu \Big) \\
& + 10 \cdot \sum_t \sum_w (ih_{s,w} + ic_{s,w}) + \frac{\sum_s TS_{s,M}}{DTS_M + \sum_s DT_{s,M}} \cdot mds_M + \frac{\sum_s TS_{s,T1}}{DTS_{T1} + \sum_s DT_{s,T1}} \cdot \\
& (mds_{T1} + \frac{1}{2} \sum_{\substack{d:d \bmod 7 \notin \{6,0\},\\ COMB_{\lceil \frac{d}{7} \rceil}=0}} \sum_s FWT1_s \cdot x_{s,T1,d}) + \frac{\sum_s TS_{s,T2}}{DTS_{T2} + \sum_s DT_{s,T2}} \cdot mds_{T2} \\
& + \frac{\sum_s TS_{s,OR}}{DTS_{OR} + \sum_s DT_{s,OR}} \cdot mds_{OR} + \frac{|S|}{\sum_t DTS_t + \sum_t \sum_s DT_{s,t}} \cdot md
\end{aligned} \tag{4.146}$$

Now we will explain how the costs are determined. First, we will determine importance factors for different categories for variables. After that, we will determine the importance per variable within this category. The costs in the objective function are a multiplication of these factors.

4.5.1 Importance factor per category

We can divide the different variables into different categories and determine a ranking of importance of these categories. The importance factors of the different categories are powers of ten. By doing this, we can also have different importance factors within the categories in such a way that the costs within one category will never exceed the costs of another category that has a higher importance factor. The different categories are

1. Shortages
2. Combined tasks
3. Extended duration tasks
4. Different surgeons Hpb/Ch
5. Equally divide tasks

Category 1 has the highest importance factor and Category 5 has the lowest importance factor. The importance factors are shown in Table 4.1. Category 1 includes all shortages that can occur. These are the shortages at the different tasks, but also when no MOD certified surgeon is scheduled or no senior transplant surgeon is scheduled or no hpb surgeon is scheduled during the week on the OR. This category has the highest importance factor of the different categories, because we want to schedule all tasks if possible. The importance factors within this category are described in Section 4.5.2.

Category 2 includes all combinations of the different tasks. These are the different tasks that can be scheduled for the same surgeon during a day. By combining different tasks, the number of shortages can be decreased. However, we do not want to combine more tasks than necessary and therefore, this category has the second highest importance factor. The importance factors within this category are described in Section 4.5.3.

Category 3 includes all tasks that can be scheduled longer for the same surgeon if this reduces the number of shortages or the number of combinations. Therefore, this category has the third highest importance factor. The importance factors within this category are described in Section 4.5.4.

Category 4 are the number of different surgeons scheduled during a week on the Hpb/Ch task. If it does not lead to extra shortages, combinations or extended durations of tasks, we want to minimize the number of different surgeons scheduled at the Hpb tasks during a week. The same thing holds for the Ch task. Therefore, this category has the fourth highest importance factor. The importance factors within this category are described in Section 4.5.5.

Category 5 includes all tasks that must be equally divided over the surgeons. We only want to do this if it does not lead to extra shortages, extra combined tasks, extra tasks with extended durations or more different surgeons at the Hpb or Ch tasks during a week. Therefore, this is the category with the lowest importance factor. The importance factors within this category are described in Section 4.5.6.

Table 4.1: The importance factors for the different categories.

Possibility	Factor
Shortage	10 000
Combined tasks	1 000
Extended duration task	100
Different surgeons Hpb/Ch	10
Equally divide tasks	1

4.5.2 Importance factors within Category 1

In the category shortages, the variable with the highest importance is the M shift. This is because the hospital has a contract with the ZUT ('zelfstandige uitname teams' this means independent organ removal teams) which says that always two surgeons are scheduled at this task with at least one of them MOD certified. We assign the same importance factor to no shortages at the M shift during a day and an MOD certified surgeon scheduled at the M shift during a day. The M shift is scheduled using start moments, therefore more days are scheduled when only the start moment is scheduled. We must take this into account in determining the importance factor.

For example, if one surgeon less is scheduled at the start moment on Monday two days a surgeon less is available for the M shift. But if one surgeon less is scheduled on Friday three days one surgeon less is available for the M shift. The importance factor of no shortages on Friday is therefore higher than the importance factor of no shortages on Monday. The same thing holds for an MOD certified surgeon on Monday or an MOD certified surgeon on Friday. The importance factor for no shortages at the M shift is higher than every possible combination of tasks. If possible the M shift should be scheduled instead of a combination of other tasks and not scheduling the M shift.

The second most important task is the $T1$ shift. This is because this is the first point of contact for transplants that are not scheduled. This shift is scheduled per day, except for the weekend. The whole weekend is scheduled when the $T1$ is scheduled on Saturday. Therefore, the importance factor of scheduling this shift on Saturday is two times the importance factor of scheduling this shift during a weekday. The importance factor of scheduling a senior surgeon at the transplant shifts is the same as the importance factor of no shortage at the $T1$ shift for one day. Also, the importance factor of scheduling a hpb surgeon at the OR during a week is the same.

The next task is the $T2$ shift. This is because this shift is needed to fulfill unscheduled transplants. This shift is also scheduled per day, but on Saturday for the whole weekend. Therefore, the importance factor of scheduling this shift on Saturday is two times the importance of scheduling this shift during weekdays.

When the M, $T1$ and $T2$ shifts are scheduled, the most important task is the Td shift. The Td shift is not more important than the $T2$ shift, because during office hours it is easier to find someone not scheduled at this task to do parts of this task than during other moments of the day. This Td shift is scheduled per day and only during weekdays. Therefore, the importance factor of scheduling this shift is the same for all days.

The Ch and Hpb tasks have the same importance factor. The same thing holds for the OR and OC. The importance factor for scheduling the Ch/Hpb task is higher than for scheduling the OR/OC task. The OR/OC tasks have the lowest importance factor, because the work during these tasks can be scheduled. When for example one surgeon less is scheduled at the outpatient clinic, this is known well ahead. Therefore, fewer patients can be scheduled during this outpatient clinic. The importance factors for Category 1 are shown in Table 4.2. The importance factors shown in this table are not the costs in the objective function. In the objective function, the importance factor of the category is also considered. The costs used in the objective function are the importance factor of Table 4.2 multiplied by the importance factor of Category 1 $(10\ 000)$ from Table 4.1.

Table 4.2: The importance factors for no shortages in the schedule made.

Shortage	Factor
M task Friday	21
MOD certified surgeon Friday	21
M task Monday	14
MOD certified surgeon Monday	14
M task Wednesday	14
MOD certified surgeon Wednesday	14
$T1$ task Saturday	8
$T1$ task weekday	4
Senior at the transplant shifts	4
Hpb surgeon at the OR	4
$T2$ task Saturday	6
$T2$ task weekday	3
Td task	2
Ch/Hpb task	1
OR/OC task	$\frac{1}{2}$

4.5.3 Importance factors within Category 2

To reduce the shortages at the different tasks, some tasks can be combined. This means that one surgeon is scheduled at these tasks instead of different surgeons. Some combinations are better than others. The importance factors of not making the possible combinations are shown in Table 4.3.

When the combination of tasks is not shown in the table, this combination is not possible. The combinations with the highest importance factor are the combinations of other (than the combinations shown in the table) tasks ($T2, Td, Ch, Hpb$). When during a day a combination is made, this leads to 5 costs (without taking into account the importance factor of the category) of that day and the costs of the combination. It is more preferable that two surgeons each have two tasks than that one surgeon has one task and the other surgeon has three tasks. Therefore, not making each combination of two tasks has an importance factor. When one surgeon has three tasks three combinations of two tasks are scheduled, while only two combinations are made when two surgeons each have two tasks. The importance factors are determined in such a way that we do not combine the tasks to reduce the shortages at the outpatient clinic and we will give one surgeon four tasks if this is needed to reduce the other shortages. The next combination is the Ch and Hpb tasks. This is because the tasks can be combined, but during the morning the surgeons scheduled at these tasks should visit inpatients at the same time. The M shift can be combined with the Hpb task on Friday because the biggest part of the Hpb shift is visiting inpatients in the morning and the M shift starts on Friday at 12:00 o'clock. Another possibility is to combine the $T2$ shift with the OR task. This is because the $T2$ shift is a backup for the transplant shifts. This surgeon is available for transplants when two surgeons are needed. When this surgeon is scheduled at the OR, no backup is available during the time the surgeon is at the OR. The last possible combination is the $T2$ shift with the Hpb task. This is also because the $T2$ shift is a backup and the biggest part of the Hpb shift is during the morning. The surgeon scheduled at the Hpb task is most of the time not busy the entire day.

Table 4.3: The importance factors for not making combinations of tasks in the schedule made.

Combination	Factor
Extra combinations during a day	5
Combine two other tasks	3
Ch and Hpb	3
$T2$ and Hpb	2
$T2$ and OR	2
M Friday and Hpb	1

4.5.4 Importance factors within Category 3

It is possible that the shortages at the different tasks can be reduced when tasks can be scheduled at the same surgeon more days. This is possible for the M and $T2$ shift, but also a weekend shift can be scheduled before a week $T1$ shift. The M shift normally has a switching moment on Wednesday. This means that on Wednesday another surgeon is scheduled at the M shift. However, it is possible to schedule the M shift for Monday till Friday for the same surgeon. In this case, the switching moment on Wednesday does not exist. The duration of the $T2$ shift can be extended. The $T2$ shift normally is scheduled two or three days for the same surgeon, but it is possible that a surgeon is scheduled four consecutive days at the $T2$ shift. The weekend before a week $T1$ shift a weekend $T2$ or M shift can be scheduled if this reduces the number of shortages or the combinations of tasks. The importance factors for these cases are shown in Table 4.4.

Table 4.4: The importance factors for not extending the duration of tasks in the schedule made.

Extend duration	factor
Weekend shift before week $T1$	3
M shift	2
$T2$ shift	1

4.5.5 Importance factors within Category 4

The number of different surgeons at the Hbp task during a week is minimized. The same thing is done for the Ch tasks and both tasks have the same importance. This is shown in Table 4.5.

Table 4.5: The importance factors for the number of different surgeons scheduled at tasks during a week in the schedule made.

Task	factor
Ch	1
Hpb	1

4.5.6 Importance factors within Category 5

We want to minimize the differences of the number of days a task is scheduled over the surgeons. To do this, the maximum number of days a surgeon is scheduled at a task is minimized. This is done for the tasks M, $T1$, $T2$, OR and the total number of tasks scheduled. To determine the costs for the different tasks, we must normalize these variables. This is done by dividing the number of surgeons that can be scheduled at the task by the number of days the task is scheduled plus the number of days already scheduled. By doing this we normalise the maximum number of days a surgeon is scheduled at a task by the average number of days a surgeon is scheduled at a task (we multiply the maximum number of days a surgeon is scheduled at a task by $\frac{1}{\text{average number of days a surgeon is scheduled at the task}}$). By using these factors, the number of different surgeons scheduled at Hpb/Ch during a week is not increased to decrease the maximum number of days a surgeon is scheduled at a task. When we look for example at the organ removal shift and all organ removal shifts are scheduled for one surgeon (which is not possible), we will get the value equal to the number of surgeons that can be scheduled at the organ removal shift in the objective function. This is (in our case) a bit bigger than one different surgeon more scheduled at one week at the Ch or Hpb task. Therefore, the value of the maximum number of days scheduled will not increase the factor ten, which is equal to the value in the objective function when one different surgeon extra is scheduled at one week at the Ch or Hpb task. When surgeons who do prefer weekend $T1$ shifts, because they cannot handle a whole week night shifts well, are scheduled at a week night shift, the importance factor is one half times the importance factor for dividing the $T1$ shift equally. This is because it is more important to distribute the $T1$ shift equally. However, we want to minimize the number of times this surgeon is scheduled at the week night shift. These factors are shown in Table 4.6.

Table 4.6: The importance factors for the maximum number of days scheduled.

Task	Factor
M shift	$\frac{\sum_s TS_{s,M}}{DTS_M+\sum_s DT_{s,M}}$
$T1$ shift	$\frac{\sum_s TS_{s,T1}}{DTS_{T1}+\sum_s DT_{s,T1}}$
Prefer weekend $T1$ shift	$\frac{1}{2} \cdot \frac{\sum_s TS_{s,T1}}{DTS_{T1}+\sum_s DT_{s,T1}}$
$T2$ shift	$\frac{\sum_s TS_{s,T2}}{DTS_{T2}+\sum_s DT_{s,T2}}$
OR task	$\frac{\sum_s TS_{s,OR}}{DTS_{OR}+\sum_s DT_{s,OR}}$
Total	$\frac{\lvert S \rvert}{\sum_t DTS_t+\sum_t \sum_s DT_{s,t}}$

4.5.7 Costs in objective function

The costs used in the objective function are shown in Table 4.7. The costs used in the objective function are the importance factors within the category multiplied by the importance factor of the category of the variable.

Table 4.7: The costs used in the objective function.

Variable	Costs
Shortage M shift Friday	$210\,000$
No MOD certified surgeon Friday	$210\,000$
Shortage M shift Monday	$140\,000$
No MOD certified surgeon Monday	$140\,000$
Shortage M shift Wednesday	$140\,000$
No MOD certified surgeon Wednesday	$140\,000$
Shortage $T1$ shift Saturday	$80\,000$
Shortage $T1$ shift weekday	$40\,000$
Senior at the transplant shifts	$40\,000$
Hpb surgeon at the OR	$40\,000$
Shortage $T2$ shift Saturday	$60\,000$
Shortage $T2$ shift weekday	$30\,000$
Shortage Td shift	$20\,000$
Shortage Ch/Hpb task	$10\,000$
Shortage OR/OC task	$5\,000$
Extra combinations during a day	$5\,000$
Combine two other tasks	$3\,000$
Ch and Hpb tasks combined	$3\,000$
$T2$ and Hpb tasks combined	$2\,000$
$T2$ and OR tasks combined	$2\,000$
M and Hpb tasks combined	$1\,000$
Weekend shift before week $T1$	300
Longer M shift	200
Longer $T2$ shift	100
Different surgeons Hpb	10
Different surgeons Ch	10
Maximum number of days M shift	$\frac{\sum_s TS_{s,M}}{DTS_M+\sum_s DT_{s,M}}$
Maximum number of days $T1$ shift	$\frac{\sum_s TS_{s,T1}}{DTS_{T1}+\sum_s DT_{s,T1}}$
Prefer weekend $T1$ shift	$\frac{1}{2} \cdot \frac{\sum_s TS_{s,T1}}{DTS_{T1}+\sum_s DT_{s,T1}}$
Maximum number of days $T2$ shift	$\frac{\sum_s TS_{s,T2}}{DTS_{T2}+\sum_s DT_{s,T2}}$
Maximum number of day OR task	$\frac{\sum_s TS_{s,OR}}{DTS_{OR}+\sum_s DT_{s,OR}}$
Maximum number of days scheduled	$\frac{\lvert S\rvert}{\sum_t DTS_t+\sum_t \sum_s DT_{s,t}}$

4.6 Schedule

By solving the ILP as described in this chapter, we schedule for certain tasks and days only the starting moment. For example the M shift is scheduled on Friday and then we know that this shift is scheduled for the same surgeon on Saturday, Sunday and Monday. Therefore, not the whole schedule is the output of the ILP. Before we have the schedule, we have to execute the procedure shown in Algorithm 2 in Appendix B. We loop over all surgeons, tasks and days. When the day is a Saturday, we know that if a task is scheduled, the task is also scheduled on Sunday. If the day is a weekday or the task is the organ removal shift, we know that this shift is scheduled for more days at once for one surgeon. When the day is a weekday and the task is the $T1$ shift and the $T1$ and Td shifts are not combined, then the whole week $T1$ is scheduled if this task is scheduled on Monday. Otherwise, the task is scheduled for one day for this surgeon. Then, the non-clinical days, days off and days the surgeon does not want to do a shift must be indicated in the schedule. Last, when a surgeon has the weekend organ removal shift during the last weekend before the schedule, the surgeon is also scheduled at this shift the first Monday of the schedule.

4.7 Different structure organ removal shift

As described in Section 3.3, we expect difficulties with scheduling the organ removal shift using the fixed structure. Therefore, we will look at some cases to schedule the organ removal shift differently. To do this, the model has to be adjusted. First, this is done in Section 4.7.1 for the case that the LUMC will cooperate with the Erasmus MC.

This means that the organ removal shift is not scheduled every other week, but constantly. The same thing holds for the Erasmus MC. Then, in Section 4.7.2 we will adjust the model to the old situation and in Section 4.7.3 we will adjust the model such that the organ removal shift is scheduled the same as the $T2$ shift.

4.7.1 Cooperate with Erasmus MC

When the LUMC cooperates with the Erasmus MC, always one surgeon must be scheduled at the organ removal shift instead of two surgeons every other week. However, we still have the structure that one surgeon is scheduled from Friday 12:00 o'clock till Monday 12:00 o'clock, another surgeon is scheduled at the organ removal shift from Monday 12:00 o'clock till Wednesday 12:00 o'clock and a surgeon (possibly the same as scheduled from Monday till Wednesday) is scheduled from Wednesday 12:00 o'clock till Friday 12:00 o'clock. By doing this, it is more difficult to determine when an MOD surgeon is scheduled at this shift, because this depends on the schedule of the Erasmus MC and the LUMC. The model described in this chapter has to be adjusted to schedule the organ removal shift this way. First, the constraints will be described and then the objective function.

Constraint (4.6) is replaced by constraint (4.147). This is because every Friday the surgeon who starts with the organ removal shift can also be scheduled at the Hpb task. Every week a surgeon starts at 12:00 o'clock with the organ removal shift.

$$w_d = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \notin \{1,3,5\} \tag{4.147}$$

Constraint (4.18) is replaces by constraint (4.148). The constraint is adjusted because the organ removal shift starts on more days when the LUMC cooperates with the Erasmus MC.

$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{0,2,4,6\} \tag{4.148}$$

Also, constraints (4.24) and (4.25) are adjusted, because the organ removal shift is scheduled more days. Therefore, we do not have a organ removal shift week. The constraints that without the cooperation hold for a determined day in the organ removal shift week must now hold for this day during all weeks. This is done with constraints (4.149) and (4.150).

$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,3\}, (d+1) \bmod 7 = NCD_s, NNCD_{s,d+1} = 0 \tag{4.149}$$
$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,3\}, ENCD_{s,d+1} = 1 \tag{4.150}$$

Constraints (4.26) – (4.31) are replaced by constraints (4.151) – (4.156) because the constraints should hold every week instead of only during organ removal shift weeks.

$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,3\}, (d+2) \bmod 7 = NCD_s, NNCD_{s,d+2} = 0, MONCD_{s,d+2} = 0 \tag{4.151}$$
$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,3\}, ENCD_{s,d+2} = 1, MONCD_{s,d+2} = 0 \tag{4.152}$$
$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 5, (d+3) \bmod 7 = NCD_s, NNCD_{s,d+3} = 0, MONCD_{s,d+3} = 0 \tag{4.153}$$
$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 5, ENCD_{s,d+3} = 1, MONCD_{s,d+3} = 0 \tag{4.154}$$
$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,3,5\}, d \bmod 7 = NCD_s, NNCD_{s,d} = 0, MNCD_{s,d} = 0 \tag{4.155}$$
$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,3,5\}, d \bmod 7 = NCD_s, ENCD_{s,d} = 1, MNCD_{s,d} = 0 \tag{4.156}$$

Constraints (4.33) – (4.37) are replaced by constraints (4.157) – (4.161) by the same reason.

$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,3,5\}, V_{s,d} = 1, MV_{s,d} = 0 \tag{4.157}$$
$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,3,5\}, V_{s,d+1} = 1 \tag{4.158}$$
$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 5, V_{s,d+2} = 1 \tag{4.159}$$
$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,3\}, V_{s,d+2} = 1, MOV_{s,d+2} = 0 \tag{4.160}$$
$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 5, V_{s,d+3} = 1, MOV_{s,d+3} = 0 \tag{4.161}$$

Also, Constraints (4.46) and (4.47) are replaced by constraints (4.162) and (4.163), because of this.

$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,3,5\}, G_{s,d+1} = 1 \tag{4.162}$$
$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 5, G_{s,d+2} = 1 \tag{4.163}$$

The organ removal shift is not scheduled on Tuesday, Thursday, Saturday and Sunday. Therefore, only on these days the dummy variables for this shift are not created. This is done with constraints (4.164) and (4.165) instead of constraints (4.56) and (4.61).

$$d_{M,d} = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{0,2,4,6\} \tag{4.164}$$
$$dm_d = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{0,2,4,6\} \tag{4.165}$$

Constraint (4.62) is deleted, because now every week the organ removal shift can be scheduled from Monday till Friday for the same surgeon.

We cannot take into account constraint (4.64), because it depends on the schedule of the Erasmus MC when a MOD certified surgeon is scheduled.

Constraint (4.76) is replaced by constraint (4.166). This is because the organ removal shift should be scheduled every week and one surgeon should be scheduled instead of two.

$$\sum_s x_{s,M,d} + d_{M,d} = 1 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{1,3,5\} \tag{4.166}$$

Constraint (4.88) is replaced by constraint (4.167) by the same reason.

$$x_{s,Hpb,d} + x_{s,M,d} \le 1 + w_d \quad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,3,5\} \tag{4.167}$$

Constraint (4.89) is replaced by constraint (4.168) because the organ removal shift is scheduled every week.

$$w_d \le 1 - lm_{\lceil \frac{d}{7} \rceil} \quad \forall d \quad \text{st} \quad d \bmod 7 = 3 \tag{4.168}$$

Also, Constraints (4.103) – (4.109) are replaced by constraints (4.169) – (4.175) because of this.

$$\begin{aligned}
x_{s,t,d} &\le 1 - x_{s,M,d} && \forall s,t,d \quad \text{st} \quad t \in \{T2, Td, OR, Ch\}, \\
& && \quad d \bmod 7 = 5 && (4.169)\\
x_{s,t,d+1} &\le 1 - x_{s,M,d} && \forall s,t,d \quad \text{st} \quad t \in \{T1, T2\}, d \bmod 7 = 5 && (4.170)\\
x_{s,t,d+3} &\le 1 - x_{s,M,d} && \forall s,t,d \quad \text{st} \quad d \bmod 7 = 5 && (4.171)\\
x_{s,t,d'} &\le 1 - x_{s,M,d} && \forall s,t,d,d' \quad \text{st} \quad t \in \{T1, T2, Td, OC, Ch, Hpb\}, \\
& && \quad d \bmod 7 = 1, d' \in \{d, d+1\} && (4.172)\\
x_{s,t,d+2} &\le 1 - x_{s,M,d} && \forall s,t,d \quad \text{st} \quad t \neq M, d \bmod 7 = 1 && (4.173)\\
x_{s,M,d+2} &\le 1 - x_{s,M,d} + lm_{\lceil \frac{d}{7} \rceil} && \forall s,d \quad \text{st} \quad d \bmod 7 = 1 && (4.174)\\
x_{s,t,d'} &\le 1 - x_{s,M,d} && \forall s,t,d,d' \quad \text{st} \quad t \in \{T2, Td, OR, Ch, Hpb\}, \\
& && \quad d \bmod 7 = 3, d' \in \{d, d+1, d+2\} && (4.175)
\end{aligned}$$

Constraint (4.121) is replaced by constraint (4.176) because we have to sum over all weeks instead of only the organ removal shift weeks.

$$\begin{aligned}
&3 \cdot \sum_{d: d \bmod 7 = 5} x_{s,M,d} + 2 \cdot \sum_{d: d \bmod 7 = 1} x_{s,M,d} + \\
&2 \cdot \sum_{d: d \bmod 7 = 3} x_{s,M,d} + DT_{s,M} \le mds_M \qquad \forall s \quad \text{st} \quad \sum_{t \in \{T1,T2,M\}} TS_{t,s} \neq TS_{M,s}
\end{aligned} \tag{4.176}$$

Constraint (4.125) is replaced by (4.177) by the same reason.

$$\begin{aligned}
&3 \cdot \sum_{d: d \bmod 7 = 5} x_{s,M,d} + 2 \cdot \sum_{d: d \bmod 7 = 1} x_{s,M,d} + 2 \cdot \sum_{d: d \bmod 7 = 3} x_{s,M,d} \\
&+ 2 \cdot \sum_{d: d \bmod 7 = 6} \sum_{t \in \{T1,T2\}} x_{s,t,d} + \sum_{d: d \bmod 7 \neq 6} \sum_{t \notin \{M,T1\}} x_{s,t,d} + \sum_{d: d \bmod 7 \notin \{1,6\}} x_{s,T1,d} \\
&+ 5 \cdot \sum_{d: d \bmod 7 = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0} x_{s,T1,d} + \sum_{d: d \bmod 7 = 1, COMB_{\lceil \frac{d}{7} \rceil} = 1} x_{s,T1,d} + 2 \cdot \sum_{d: d \bmod 7 = 6} x_{s,T1,d} \\
&+ \sum_t DT_{s,t} \le md \qquad \forall s
\end{aligned} \tag{4.177}$$

Constraint (4.126) is replaced by (4.178), because the organ removal shift starts every Monday, Wednesday and Friday. Therefore, some surgeons can already be scheduled at the organ removal shift during these days.

$$x_{s,M,d} = X_{s,M,d} \qquad \forall s,d \quad \text{st} \quad \sum_t TS_{t,s} = \sum_{t \in \{M,OR,OC\}} TS_{t,s}, d \bmod 7 \in \{1,3,5\} \tag{4.178}$$

Constraint (4.127) is replaced by constraint (4.179), because we have to sum over all weeks instead of only the organ removal shift weeks.

$$3 \cdot \sum_{d:d \bmod 7=5} x_{s,M,d} + 2 \cdot \sum_{d:d \bmod 7=1} x_{s,M,d} + \\ 2 \cdot \sum_{d:d \bmod 7=3} x_{s,M,d} \leq \frac{|D|}{5} \qquad \forall s \quad \text{st} \quad \sum_{t \in \{T1,T2,M\}} TS_{t,s} = TS_{M,s} \tag{4.179}$$

Constraint (4.144) is replaced by constraint (4.180), because it always holds that the surgeon who is scheduled the last weekend before the new schedule at the organ removal shift cannot be scheduled at a task on Monday.

$$x_{s,t,1} \leq 1 - WBM_S \qquad \forall s,t \tag{4.180}$$

We will also add a new constraint. This constraint ensures that the surgeon who is scheduled on the organ removal shift on Wednesday cannot also be scheduled at the organ removal shift on Friday. This is done with constraint (4.181).

$$x_{s,M,d+2} \leq 1 - x_{s,M,d} \qquad \forall s,d \quad \text{st} \quad d \bmod 5 = 3 \tag{4.181}$$

The objective function has to be adjusted. Also in the objective function we have to sum over all weeks for the organ removal shift instead of the organ removal shift weeks. This is shown in the new objective function (4.182)

$$\begin{aligned}
\text{minimize} \quad & 210\,000 \cdot \sum_{d:d \bmod 7=5} (d_{M,d} + dm_d) + 140\,000 \cdot \sum_{d:d \bmod 7 \in \{1,3\}} (d_{M,d} + dm_d) \\
& + 80\,000 \cdot \sum_{d:d \bmod 7=6} d_{T1,d} + 40\,000 \cdot \Big(\sum_{\substack{d:d \bmod 7 \neq 6, \\ COMB_{\lceil \frac{d}{7} \rceil}=1}} d_{T1,d} + 5 \cdot \sum_{\substack{d:d \bmod 7 \neq 6, \\ COMB_{\lceil \frac{d}{7} \rceil}=0}} d_{T1,d} + \sum_w dhpb_w \\
& + \sum_d ds_d \Big) + 60\,000 \cdot \sum_{d:d \bmod 7=6} d_{T2,d} + 30\,000 \cdot \sum_{d:d \bmod 7 \neq 6} d_{T2,d} + 20\,000 \cdot \sum_d d_{Td,d} \\
& + 10\,000 \cdot \sum_d \sum_{t \in \{Ch,Hpb\}} d_{t,d} + 5\,000 \cdot \sum_d \sum_{t \in \{OC,OR\}} d_{t,d} + 3\,000 \cdot \sum_d u_d + 2\,000 \cdot \sum_d z_d \\
& + 2\,000 \cdot \sum_d y_d + 1\,000 \cdot \sum_d w_d + 300 \cdot \sum_w wk_w + 200 \cdot \sum_w lm_w + 100 \cdot \Big(\sum_d fd_d + fdm + fdtu \Big) \\
& + 10 \cdot \sum_t \sum_w (ih_{s,w} + ic_{s,w}) + \frac{\sum_s TS_{s,M}}{DTS_M + \sum_s DT_{s,M}} \cdot mds_M + \frac{\sum_s TS_{s,T1}}{DTS_{T1} + \sum_s DT_{s,T1}} \cdot \\
& (mds_{T1} + \frac{1}{2} \sum_{\substack{d:d \bmod 7 \notin \{6,0\}, \\ COMB_{\lceil \frac{d}{7} \rceil}=0}} \sum_s FWT1_s \cdot x_{s,T1,d}) + \frac{\sum_s TS_{s,T2}}{DTS_{T2} + \sum_s DT_{s,T2}} \cdot mds_{T2} \\
& + \frac{\sum_s TS_{s,OR}}{DTS_{OR} + \sum_s DT_{s,OR}} \cdot mds_{OR} + \frac{|S|}{\sum_t DTS_t + \sum_t \sum_s DT_{s,t}} \cdot md
\end{aligned} \tag{4.182}$$

4.7.2 Old situation

The old situation is the situation in the LUMC before the structure of three blocks was introduced. Also in this case the organ removal shift is scheduled every other week, but further no rules were available for scheduling the organ removal shift. We have a switching moment at 22:00 o'clock every day, except at Saturday. The whole weekend this shift is done by the same surgeon. Therefore, the surgeon who starts on Friday 22:00 o'clock is scheduled at this shift till Sunday 22:00 o'clock. With no further rules we mean that it does not matter how long the surgeon is scheduled consecutively at this shift. The organ removal week still starts at Friday 12:00 o'clock and ends at the next Friday 12:00 o'clock. Therefore, every Friday a surgeon is scheduled at the organ removal shift, but only half a day. In this section the adjustments of the model shown in this chapter are shown to schedule the organ removal shift in the old situation. The switching moments of the organ removal shift in this case are shown in Figure 4.1. When a block is filled, the organ removal shift must be scheduled at this day and on this time moment. When the pattern of a block changes another transplant surgeon can be scheduled.

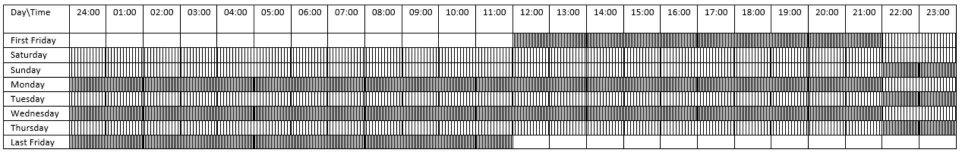

Figure 4.1: The switching moments of the organ removal shift at the different days this shift is scheduled.

Below we will describe which constraints are adjusted and how the new objective function looks like. First, constraint 4.6 is replaced by constraint (4.183) because the Hpb task and M shift can be combined when the organ removal shift starts at 12:00 o'clock. This only is on the first Friday in the old situation.

$$w_d = 0 \qquad \forall d \quad \text{st} \quad d \bmod 14 \neq 5 + 7 \cdot MW \tag{4.183}$$

Also, constraint (4.18) must be adjusted. This is because the organ removal shift is scheduled the whole week during an organ removal shift week instead of only three starting moments. Constraint (4.184) replaces constraint (4.18).

$$\begin{aligned} x_{s,M,d} = 0 \qquad \forall s, d \quad \text{st} \quad & d \bmod 14 \in \{1 + 7 \cdot MW, 2 + 7 \cdot MW, 3 + 7 \cdot MW, 4 + 7 \cdot MW, \\ & 6 + 7 \cdot (1 - MW), 0, 7\} \end{aligned} \tag{4.184}$$

Constraints (4.22) – (4.31) are replaced by constraints (4.185) and (4.186). This is because the organ removal shift is scheduled for a whole day, except at the beginning and end of the organ removal shift week. Therefore, the organ removal shift does not have to be scheduled half a day during the non-clinical day of a surgeon.

$$x_{s,t,d} = 0 \qquad \forall s, t, d \quad \text{st} \quad t \neq T1, d \bmod 7 = NCD_s, NNCD_s = 0 \tag{4.185}$$

$$x_{s,t,d} = 0 \qquad \forall s, t, d \quad \text{st} \quad t \neq T1, ENCD_{s,d} = 1 \tag{4.186}$$

Constraints (4.33) – (4.37) are replaced by constraint (4.187). This is because the organ removal shift is scheduled per day in this case. Therefore, only the organ removal shift during the day off of the surgeon cannot be scheduled. Also in this case we do not need the parameters that indicate if a surgeon can do a half day organ removal shift during the day off.

$$x_{s,t,d} = 0 \qquad \forall s, t, d \quad \text{st} \quad d \bmod 7 \in \{1, 2, 3, 4, 5\}, V_{s,d} = 1 \tag{4.187}$$

The organ removal shift is scheduled for the whole weekend when this shift is scheduled on Saturday. Therefore, this shift cannot be scheduled at Saturday when a surgeon has a day off on Sunday. This is done with constraint (4.43) for the $T1$ and $T2$ shifts. This constraint is replaced by constraints (4.188). This new constraint is also for the organ removal shift.

$$x_{s,t,d} = 0 \qquad \forall s, t, d \quad \text{st} \quad t \in \{T1, T2, M\}, d \bmod 7 = 6, V_{s,d} + V_{s,d+1} \geq 1 \tag{4.188}$$

Constraints (4.45), (4.46) and (4.47) are replaced by constraint (4.189), (4.190) and (4.191) . When a surgeon does not want to do a shift during a day, the organ removal shift cannot be scheduled. The organ removal shift at the last Friday of the organ removal shift week can be scheduled, because this shift ends at 12:00 o'clock. Therefore, the surgeon does not have any shift after office hours. When the shift is scheduled on Saturday, the Sunday is also scheduled. Therefore, we cannot schedule the shift on Saturday when the surgeon does not want to do shifts on Sunday.

$$x_{s,t,d} = 0 \qquad \forall s, t, d \quad \text{st} \quad t \in \{T1, T2\}, d \bmod 7 \in \{1, 2, 3, 4, 5\}, G_{s,d} = 1 \tag{4.189}$$

$$x_{s,M,d} = 0 \qquad \forall s, d \quad \text{st} \quad d \bmod 14 \neq 5 + 7 \cdot (1 - MW), G_{s,d} = 1 \tag{4.190}$$

$$x_{s,M,d} = 0 \qquad \forall s, d \quad \text{st} \quad d \bmod 14 = 6 + 7 \cdot MW, G_{s,d+1} = 1 \tag{4.191}$$

Constraint (4.56) is replaced by constraint (4.192). This is because the dummy variable is equal to zero if the organ removal shift does not have to be scheduled and in this case the organ removal shift is scheduled on more separate days.

$$\begin{aligned} d_{M,d} = 0 \qquad \forall d \quad \text{st} \quad & d \bmod 14 \in \{1 + 7 \cdot MW, 2 + 7 \cdot MW, 3 + 7 \cdot MW, 4 + 7 \cdot MW, \\ & 6 + 7 \cdot (1 - MW), 0, 7\} \end{aligned} \tag{4.192}$$

The same thing holds for the dummy variable for the scheduled MOD certified surgeon. Therefore, constraint (4.61) is replaced by constraint (4.193).

$$\begin{aligned} dm_d = 0 \qquad \forall s, d \quad \text{st} \quad & d \bmod 14 \in \{1 + 7 \cdot MW, 2 + 7 \cdot MW, 3 + 7 \cdot MW, 4 + 7 \cdot MW, \\ & 6 + 7 \cdot (1 - MW), 0, 7\} \end{aligned} \tag{4.193}$$

Constraint (4.62) is not needed any more, because the organ removal shift can be scheduled as long as wanted for the same surgeon in this case. We also do not need the variable lm_w anymore.

Constraint (4.64) is replaced by constraint (4.194). This is done because every switch moment of the organ removal shift at least one MOD certified surgeon must be scheduled. When we schedule the organ removal shift as was done in the old situation, we have more switching moments.

$$\sum_s (MOD_s + MODC_{s,d}) \cdot x_{s,M,d} \geq 1 - dm_d \qquad \forall d \quad \text{st} \quad d \bmod 14 \in \{5 + 7 \cdot MW, 6 + 7 \cdot MW, \\ 1 + 7 \cdot (1 - MW), 2 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW), \\ 4 + 7 \cdot (1 - MW), 5 + 7 \cdot (1 - MW)\} \qquad (4.194)$$

Constraint (4.76) is replaced by constraint (4.195). This is because every switch moment two surgeons must be scheduled and we have more switching moments in the old situation.

$$\sum_s x_{s,M,d} + d_{M,d} = 2 \qquad \forall d \quad \text{st} \quad d \bmod 14 \in \{5 + 7 \cdot MW, 6 + 7 \cdot MW, 1 + 7 \cdot (1 - MW), \\ 2 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW), 4 + 7 \cdot (1 - MW), \\ 5 + 7 \cdot (1 - MW)\} \qquad (4.195)$$

Constraint (4.80) is replaced by constraint (4.196). A weekend shift is not preferred before a week $T1$ shift. When we schedule the organ removal shift as done in the old situation, only a weekend organ removal shift is possible. Therefore, we have to add the weekend organ removal shift in this constraint.

$$x_{s,t,d} \leq 1 - x_{s,T1,d+2} + wk_{\lceil \frac{d+2}{7} \rceil} \qquad \forall s,t,d \quad \text{st} \quad t \in \{T2, M\}, COMB_{\lceil \frac{d+2}{7} \rceil} = 0, \\ d \bmod 7 = 6 \qquad (4.196)$$

By the same reason we replace constraint (4.81) by constraint (4.197). No weekend shift can be scheduled after a week $T1$ shift. Therefore, also no organ removal shift can be scheduled on Saturday.

$$\sum_{t \in \{T1,T2,M\}} x_{s,t,d+5} + x_{s,T1,d} \leq 1 \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 0, d \bmod 7 = 1 \qquad (4.197)$$

Constraint (4.84) is replaced by constraint (4.198). This constraint ensures no three weeks consecutive weekend shift is scheduled for a surgeon. This constraint must be adjusted, because the weekend shift of the organ removal shift does not start on Friday, but on Saturday in the old situation.

$$\sum_{t \in \{T1,T2,M\}} (x_{s,t,d} + x_{s,t,d+7} + x_{s,t,d+14}) \leq 2 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 6 \qquad (4.198)$$

Constraint (4.88) is replaced by constraint (4.199). This is because the Hpb shift can only be combined with the organ removal shift during the first Friday of the organ removal shift. Therefore, during all other switch moments not both tasks can be scheduled for the same surgeon. The variable w_d only can be one during the first Friday of the organ removal shift. This variable is equal to one on the first Friday of the organ removal shift if both tasks are scheduled for the same surgeon.

$$x_{s,Hpb,d} + x_{s,M,d} \leq 1 + w_d \qquad \forall s,d \quad \text{st} \quad d \bmod 14 \in \{5 + 7 \cdot MW, 6 + 7 \cdot MW, \\ 1 + 7 \cdot (1 - MW), 2 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW), \\ 4 + 7 \cdot (1 - MW), 5 + 7 \cdot (1 - MW)\} \qquad (4.199)$$

The variable lm_w is not needed in this case. Therefore, constraint (4.89) is not included in this model.

Constraints (4.103)–(4.109) are not needed in the old situation, because the organ removal shift can be scheduled as wanted. However, during the organ removal shifts no other tasks, except the Hpb task can be scheduled. Scheduling the Hpb task at the same surgeon as the organ removal shift is done with constraint (4.199). For all other tasks this is done with constraint (4.200).

$$x_{s,t,d} \leq 1 - x_{s,M,d} \qquad \forall s,t,d \quad \text{st} \quad t \notin \{Hpb, M\}, d \bmod 14 \in \{5 + 7 \cdot MW, 6 + 7 \cdot MW, \\ 1 + 7 \cdot (1 - MW), 2 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW), \\ 4 + 7 \cdot (1 - MW), 5 + 7 \cdot (1 - MW)\} \qquad (4.200)$$

Constraint (4.121) is replaced by (4.201), because we do not schedule multiple days at once in the old situation, except the weekend. Therefore, we have to adjust the weights in this constraint.

$$\sum_{\substack{d: d \bmod 14 \notin \\ \{6+7 \cdot MW, 5, 12\}}} x_{s,M,d} + 2 \cdot \sum_{\substack{d: d \bmod 14 = \\ 6+7 \cdot MW}} x_{s,M,d} + \frac{1}{2} \cdot \sum_{d: d \bmod 7 = 5} x_{s,M,d} \leq mds_M \qquad \forall s \quad \text{st} \quad \sum_{t \in \{T1,T2,M\}} TS_{t,s} \\ \neq TS_{M,s} \qquad (4.201)$$

The same thing is done for constraint (4.125), where we determine the maximum number of tasks scheduled at a surgeon. The new constraint is constraint (4.202).

$$\sum_{\substack{d:d \bmod 14 \notin \\ \{6+7\cdot MW,5,12\}}} x_{s,M,d} + 2 \cdot \sum_{\substack{d:d \bmod 14= \\ 6+7\cdot MW}} x_{s,M,d} + \frac{1}{2} \cdot \sum_{d:d \bmod 7=5} x_{s,M,d} + 2 \cdot \sum_{d:d \bmod 7=6} \sum_{t\in\{T1,T2\}} x_{s,t,d}$$
$$+ \sum_{d:d \bmod 7\neq 6} \sum_{t\notin\{M,T1\}} x_{s,t,d} + \sum_{d:d \bmod 7\notin\{1,6\}} x_{s,T1,d} + 5 \cdot \sum_{d:d \bmod 7=1, COMB_{\lceil \frac{d}{7}\rceil}=0} x_{s,T1,d}$$
$$+ \sum_{d:d \bmod 7=1, COMB_{\lceil \frac{d}{7}\rceil}=1} x_{s,T1,d} + 2 \cdot \sum_{d:d \bmod 7=6} x_{s,T1,d} + \sum_{t} DT_{s,t} \leq md \qquad \forall s \tag{4.202}$$

Constraint (4.126) is replaced by constraint (4.203). The constraint must hold for all switching moments and the switching moments are changed.

$$x_{s,M,d} = X_{s,M,d} \qquad \forall s,d \quad \text{st} \quad \sum_{t} TS_{t,s} = \sum_{t\in\{M,OR,OC\}} TS_{t,s}, d \bmod 14 \in \{5+7\cdot MW, 6+7\cdot MW,$$
$$1+7\cdot(1-MW), 2+7\cdot(1-MW), 3+7\cdot(1-MW), 4+7\cdot(1-MW),$$
$$5+7\cdot(1-MW)\} \tag{4.203}$$

Also constraint (4.127) has to be adjusted in the same way as constraint (4.121). This is done in constraint (4.204).

$$\sum_{\substack{d:d \bmod 14 \notin \\ \{6+7\cdot MW,5,12\}}} x_{s,M,d} + 2 \cdot \sum_{\substack{d:d \bmod 14= \\ 6+7\cdot MW}} x_{s,M,d} + \frac{1}{2} \cdot \sum_{d:d \bmod 7=5} x_{s,M,d} \leq \frac{|D|}{5} \qquad \forall s \quad \text{st} \quad \sum_{t\in\{T1,T2,M\}} TS_{t,s}$$
$$= TS_{M,s} \tag{4.204}$$

Constraints (4.136) and (4.137) are adjusted in the same way as constraint (4.84), because constraints (4.136) and (4.137) are the boundary constraints of constraint (4.84). This is done in constraints (4.205) and (4.206).

$$\sum_{t\in\{M,T1,T2\}} x_{s,t,6} \leq 2 - WB_s - WB_s \cdot WBT_s \qquad \forall s \tag{4.205}$$
$$\sum_{t\in\{M,T1,T2\}} (x_{s,t,6} + x_{s,t,13}) \leq 2 - WB_s \qquad \forall s \tag{4.206}$$

Constraint (4.143) is adjusted in the same way as constraint (4.80), because constraint (4.143) is the boundary constraint of constraint (4.80). This is done in constraint (4.207)

$$x_{s,T1,1} \leq wk_1 \qquad \forall s \quad \text{st} \quad COMB_1 = 0, (ND_s > 0 \quad \text{or} \quad WBM_s = 1) \tag{4.207}$$

Last, constraint (4.144) is not needed in the old situation, because the surgeon with the weekend organ removal shift is not automatically scheduled on the Monday after this weekend.

Also, the objective function has to be adjusted. The organ removal shift is scheduled per day and therefore shortages can occur per day. The costs of the shortages are adjusted, because this are the costs for a shortage during one day except for the Friday and Saturday. On Saturday the shift is scheduled for two days and therefore these costs are twice the costs of a shortage during one day. On Friday the organ removal shift is scheduled till/from 12:00 o'clock. Therefore we divide the costs by two during these days. We also remove the costs for scheduling the organ removal shift from Monday till Friday for the same surgeon, because this does not matter in the old situation. This is shown in the new objective function (4.208).

$$\text{minimize} \quad 140\,000 \cdot \sum_{\substack{d:d \bmod 14= \\ 6+7\cdot MW}} (d_{M,d} + dm_d) + 70\,000 \cdot \sum_{\substack{d:d \bmod 14\notin \\ \{6+7\cdot MW,5,12\}}} (d_{M,d} + dm_d) + 35\,000 \cdot \sum_{d:d \bmod 7=5}$$
$$(d_{M,d} + dm_d) + 80\,000 \cdot \sum_{d:d \bmod 7=6} d_{T1,d} + 40\,000 \cdot \Bigg(\sum_{\substack{d:d \bmod 7\neq 6, \\ COMB_{\lceil \frac{d}{7} \rceil}=1}} d_{T1,d} + 5 \cdot \sum_{\substack{d:d \bmod 7\neq 6, \\ COMB_{\lceil \frac{d}{7} \rceil}=0}} d_{T1,d}$$
$$+ \sum_w dhpb_w + \sum_d ds_d \Bigg) + 60\,000 \cdot \sum_{d:d \bmod 7=6} d_{T2,d} + 30\,000 \cdot \sum_{d:d \bmod 7\neq 6} d_{T2,d} + 20\,000 \cdot \sum_d d_{Td,d}$$
$$+ 10\,000 \cdot \sum_d \sum_{t\in\{Ch,Hpb\}} d_{t,d} + 5\,000 \cdot \sum_d \sum_{t\in\{OC,OR\}} d_{t,d} + 3\,000 \cdot \sum_d u_d + 2\,000 \cdot \sum_d z_d$$
$$+ 2\,000 \cdot \sum_d y_d + 1\,000 \cdot \sum_d w_d + 300 \cdot \sum_w wk_w + 100 \cdot \Big(\sum_d fd_d + fdm + fdtu \Big)$$
$$+ 10 \cdot \sum_t \sum_w (ih_{s,w} + ic_{s,w}) + \frac{\sum_s TS_{s,M}}{DTS_M + \sum_s DT_{s,M}} \cdot mds_M + \frac{\sum_s TS_{s,T1}}{DTS_{T1} + \sum_s DT_{s,T1}} \cdot$$
$$(mds_{T1} + \frac{1}{2} \sum_{\substack{d:d \bmod 7\notin\{6,0\}, \\ COMB_{\lceil \frac{d}{7} \rceil}=0}} \sum_s FWT1_s \cdot x_{s,T1,d}) + \frac{\sum_s TS_{s,T2}}{DTS_{T2} + \sum_s DT_{s,T2}} \cdot mds_{T2}$$
$$+ \frac{\sum_s TS_{s,OR}}{DTS_{OR} + \sum_s DT_{s,OR}} \cdot mds_{OR} + \frac{|S|}{\sum_t DTS_t + \sum_t \sum_s DT_{s,t}} \cdot md \qquad (4.208)$$

4.7.3 Same structure as the $T2$ shift

The organ removal shift can also be scheduled in the same way as the $T2$ shift. By doing this, scheduling the organ removal shift is easier than with the fixed structure. We have more switching moments in this case and therefore more options of scheduling the organ removal shift. The surgeons cannot be scheduled more than four consecutive days at this shift and no shift can be scheduled till 22:00 o'clock when a surgeon starts at 22:00 o'clock with the organ removal shift. Also, the organ removal shift is scheduled for at least two consecutive days for one surgeon. Therefore, the surgeons scheduled at the organ removal shift are fitter than in the old situation (Section 4.7.2).

When we want to schedule the organ removal shift the same as the $T2$ shift the model as described in this chapter can be used with the adjustments described in Section 4.7.2 and the constraints shown below have to be adjusted added.

Constraints (4.111) and (4.112) have to be adjusted, because not only the $T1/Td$ combination cannot be scheduled after the $T2$ shift, also not after the M shift. This is done with constraints (4.209) and (4.210).

$$x_{s,T1,d+1} \leq 1 - x_{s,t,d} \qquad \forall s,t,d \quad \text{st} \quad t \in \{T2, M\}, COMB_{\lceil \frac{d+1}{7} \rceil} = 1, \; (d+1) \bmod 7 \neq 1 \qquad (4.209)$$
$$x_{s,T1,d+2} \leq 1 - x_{s,t,d} \qquad \forall s,t,d \quad \text{st} \quad t \in \{T2, M\}, COMB_{\lceil \frac{d+2}{7} \rceil} = 1, \; (d+2) \bmod 7 = 1 \qquad (4.210)$$

Also constraints (4.114) and (4.115) have to be adjusted. Not only the $T2$ shift cannot be scheduled after the $T1/Td$ combination, also the M shift cannot be scheduled after the $T1/Td$ combination. This is done with constraints (4.211) and (4.212)

$$x_{s,t,d+1} \leq 1 - x_{s,T1,d} \qquad \forall s,t,d \quad \text{st} \quad t \in \{T2, M\}, COMB_{\lceil \frac{d}{7} \rceil} = 1, \; (d+1) \bmod 7 \neq 1 \qquad (4.211)$$
$$x_{s,t,d+2} \leq 1 - x_{s,T1,d} \qquad \forall s,t,d \quad \text{st} \quad t \in \{T2, M\}, COMB_{\lceil \frac{d}{7} \rceil} = 1, \; (d+2) \bmod 7 = 1 \qquad (4.212)$$

The next constraints have to be added to the model. With the first constraints we ensure that the $T2$ shift is not scheduled after the M shift and vice versa. Constraints (4.213) and (4.214) ensure that the $T2$ shift cannot be scheduled after the M shift. However, the $T2$ shift can be scheduled the Saturday after the last Friday of the organ removal shift. This is because the organ removal shift ends on the last Friday at 12:00 o'clock. Constraints (4.215) and (4.216) ensure that the M shift cannot be scheduled after the $T2$ shift. The start of the organ removal shift week on Friday can be scheduled after the $T2$ shift, because this organ removal shift start at 12:00 o'clock.

$$\begin{aligned}
x_{s,T2,d+1} &\leq 1 - x_{s,M,d} && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1+7\cdot(1-MW), 2+7\cdot(1-MW), \\
& && 3+7\cdot(1-MW), 4+7\cdot(1-MW), 5+7\cdot MW\} && (4.213)\\
x_{s,T2,d+2} &\leq 1 - x_{s,M,d} && \forall s,d \quad \text{st} \quad d \bmod 14 = 6+7\cdot MW && (4.214)\\
x_{s,M,d+1} &\leq 1 - x_{s,T2,d} && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1+7\cdot(1-MW), 2+7\cdot(1-MW), \\
& && 3+7\cdot(1-MW), 4+7\cdot(1-MW), 5+7\cdot(1-MW)\} && (4.215)\\
x_{s,M,d+2} &\leq 1 - x_{s,T2,d} && \forall s,d \quad \text{st} \quad d \bmod 14 = 6+7\cdot MW && (4.216)
\end{aligned}$$

We want to schedule the organ removal shift for at least two consecutive days. This is done with constraints (4.217) and (4.218).

$$\begin{aligned}
x_{s,M,d+1} &\geq x_{s,M,d} - x_{s,M,d-2} && \forall d \quad \text{st} \quad d \bmod 14 = 1+7\cdot(1-MW) && (4.217)\\
x_{s,M,d+1} &\geq x_{s,M,d} - x_{s,M,d-1} && \forall d \quad \text{st} \quad d \bmod 7 \in \{2+7\cdot(1-MW), 3+7\cdot(1-MW), \\
& && 4+7\cdot(1-MW), 5+7\cdot(1-MW), 5+7\cdot MW\} && (4.218)
\end{aligned}$$

The M shift can be scheduled at most four days consecutively. We want to indicate when the M shift is scheduled four consecutive days, because we want to minimize the number of times the M shift is scheduled four days consecutively. To do this, we need a new variable fdm_d as shown below.

$$fdm_d = \begin{cases} 1 & \text{if the } M \text{ shift is scheduled for four days consecutively at the same surgeon,} \\ & \text{starting at day } d \\ 0 & \text{otherwise} \end{cases}$$

This variable is minimized in the objective function with the same costs as the lm_w variable has in the model for the fixed structure of the organ removal shift. The value of fdm_d is determined with constraints (4.219) – (4.224).

$$\begin{aligned}
\sum_{d'=d}^{d+3} x_{s,M,d'} &\leq 3 + fdm_d && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1+7\cdot(1-MW), \\
& && 2+7\cdot(1-MW)\} && (4.219)\\
\sum_{d'=d}^{d+3} x_{s,M,d'} &\leq 3 && \forall s,d \quad \text{st} \quad d \bmod 14 = 3+7\cdot(1-MW) && (4.220)\\
\sum_{d'=d}^{d+2} x_{s,M,d'} &\leq 2 + fdm_d && \forall s,d \quad \text{st} \quad d \bmod 14 = 4+7\cdot(1-MW) && (4.221)\\
\sum_{d'=d}^{d+3} x_{s,M,d'} &\leq 2 + fdm_d && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{5+7\cdot MW, 6+7\cdot MW\} && (4.222)\\
x_{s,M,d+4} &\leq 2 - fdm_d - x_{s,M,d} && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1+7\cdot(1-MW), 2+7\cdot(1-MW), \\
& && 4+7\cdot(1-MW), 5+7\cdot MW, 6+7\cdot MW\} && (4.223)\\
fdm_d &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1+7\cdot MW, 2+7\cdot MW, 3+7\cdot MW, \\
& && 4+7\cdot MW, 3+7\cdot(1-MW), 6+7\cdot(1-MW), \\
& && 5+7\cdot(1-MW), 0, 7\} && (4.224)
\end{aligned}$$

We have to adjust the objective function in this case. The objective function is almost the same as the objective function in the old situation (Section 4.7.2). However, we also want to minimize the number of times the organ removal shift is scheduled four consecutive days for the same surgeon. The objective function (4.225) is the objective function for this case.

$$
\begin{aligned}
\text{minimize}\quad & 140\,000 \cdot \sum_{\substack{d:d \bmod 14= \\ 6+7\cdot MW}} (d_{M,d} + dm_d) + 70\,000 \cdot \sum_{\substack{d:d \bmod 14 \notin \\ \{6+7\cdot MW,5,12\}}} (d_{M,d} + dm_d) + 35\,000 \cdot \sum_{d:d \bmod 7=5} \\
& (d_{M,d} + dm_d) + 80\,000 \cdot \sum_{d:d \bmod 7=6} d_{T1,d} + 40\,000 \cdot \Bigg(\sum_{\substack{d:d \bmod 7\neq 6, \\ COMB_{\lceil \frac{d}{7} \rceil}=1}} d_{T1,d} + 5 \cdot \sum_{\substack{d:d \bmod 7\neq 6, \\ COMB_{\lceil \frac{d}{7} \rceil}=0}} d_{T1,d} \\
& + \sum_w dhpb_w + \sum_d ds_d \Bigg) + 60\,000 \cdot \sum_{d:d \bmod 7=6} d_{T2,d} + 30\,000 \cdot \sum_{d:d \bmod 7\neq 6} d_{T2,d} + 20\,000 \cdot \sum_d d_{Td,d} \\
& + 10\,000 \cdot \sum_d \sum_{t\in\{Ch,Hpb\}} d_{t,d} + 5\,000 \cdot \sum_d \sum_{t\in\{OC,OR\}} d_{t,d} + 3\,000 \cdot \sum_d u_d + 2\,000 \cdot \sum_d z_d \\
& + 2\,000 \cdot \sum_d y_d + 1\,000 \cdot \sum_d w_d + 300 \cdot \sum_w wk_w + 200 \cdot \sum_d fdm_d + 100 \cdot \Big(\sum_d fd_d + fdm + fdtu \Big) \\
& + 10 \cdot \sum_t \sum_w (ih_{s,w} + ic_{s,w}) + \frac{\sum_s TS_{s,M}}{DTS_M + \sum_s DT_{s,M}} \cdot mds_M + \frac{\sum_s TS_{s,T1}}{DTS_{T1} + \sum_s DT_{s,T1}} \cdot \\
& (mds_{T1} + \frac{1}{2} \sum_{\substack{d:d \bmod 7\notin\{6,0\}, \\ COMB_{\lceil \frac{d}{7} \rceil}=0}} \sum_s FWT1_s \cdot x_{s,T1,d}) + \frac{\sum_s TS_{s,T2}}{DTS_{T2} + \sum_s DT_{s,T2}} \cdot mds_{T2} \\
& + \frac{\sum_s TS_{s,OR}}{DTS_{OR} + \sum_s DT_{s,OR}} \cdot mds_{OR} + \frac{|S|}{\sum_t DTS_t + \sum_t \sum_s DT_{s,t}} \cdot md
\end{aligned}
\tag{4.225}
$$

Chapter 5

Results for scheduling tasks

The model shown in Chapter 4 is programmed in the 64 bits version of AIMMS 4.50 and is solved with the 64 bits version of GUROBI 7.5.1 on an HP Elitebook 8570w laptop with intel core i7 processor (8GB RAM) and 64 bits Windows 10. Section 5.1 describes how the model is validated, Section 5.2 shows the input used for making the quarter schedule, Section 5.3 shows the results for this quarter schedule, Section 5.4 shows the results when the LUMC cooperates with the Erasmus MC to schedule the organ removal shift, Section 5.5 shows the results when the organ removal shift is scheduled by the old structure, Section 5.6 shows the results when the organ removal shift is scheduled by the same structure as the $T2$ shift, Section 5.7 compares the results for the different cases of scheduling the organ removal shift and Section 5.8 describes the schedules made by hand during the first and second quarter of 2018.

5.1 Validation

The validation of the model was done with help of the transplant surgeons. The constraints used in the model were reviewed by the transplant surgeons and also the schedule made was reviewed by transplant surgeons. The first case is to check if the constraints used in the model are correct, while the second case is to check if constraints have been missed. So, the conditions for scheduling the different tasks, where the constraints are derived from, were checked if they are correct. After that, the schedule made was checked to see if the schedule has impossibilities. When this is the case, a condition has been missed and must be added. This procedure was repeated a couple of times.

5.2 Case-study specific input

To make the quarter schedule for the transplant surgeons, the parameters have values according to that quarter. The schedule is made for October till January 2018. In total thirteen surgeons should be scheduled during 98 days (14 weeks). Which surgeons are MOD certified, which surgeons are junior and which surgeons are hpb surgeons is shown in Table C.1 in Appendix C. Which tasks can be scheduled for the different surgeons is shown in Table C.2 in Appendix C. The first Monday the organ removal shift must not be scheduled. Therefore $MW = 0$. The days off or congress days of the surgeons are shown in Figure C.1 in Appendix C. When surgeon s has a V at day d, the surgeon has a day off during day d. The non-clinical day can be replaced. The non-clinical days during the quarter are shown in Figure C.1 in Appendix C, indicated with NCD. During some weeks no non-clinical days are scheduled, because these weeks are vacation weeks. During these weeks surgeons have vacation and therefore the surgeons present do not have a non-clinical day. The days a surgeon does not want to do shifts are shown in Figure C.1 in Appendix C and are indicated with G. Also the tasks that are already scheduled before the schedule is made, are shown in Figure C.1 in Appendix C.

We also must know when the $T1$ and Td shifts are combined in one shift. During which weeks this is done is shown in Table C.3 in Appendix C. During this schedule some public holidays occur, these are shown in Table C.4 in Appendix C. In Table C.5 in Appendix C is shown when the organ removal shift can be scheduled during a non-clinical day or a day off. We start with zero days already done for each surgeon and shift, because this data is not known. The parameters for the boundary are shown in Table C.6 in Appendix C and the number of days a task should be scheduled is shown in Table C.7 in Appendix C.

5.3 Case-study results

When we solve the model in AIMMS with GUROBI for the parameters as described in Section 5.2, an optimal solution is found within four hours. The objective function has the value 2802079.12. This means that the optimal schedule, given the availability of the surgeons, that fulfills all constraints is found. Figure 5.1 shows the schedule made by the model described in Chapter 4. The schedule is divided in five blocks of a number of weeks. Each row in this block corresponds to a surgeon and each column to a day. The text in the block belonging to day d and surgeon s specifies the task(s) scheduled for surgeon s at day d. When a surgeon has a block without text, no task is scheduled that day for this surgeon and the surgeon is available to be scheduled at all tasks that can be scheduled for this surgeon. The days with a grey background are weekend days. It can be seen that fewer tasks are scheduled during the weekend. This is because only some shifts must be scheduled during the weekend.

In Figure 5.2 the shortages during the quarter schedule are shown. We see that while the shortages of the organ removal shift have the highest costs in the objective function, we do have shortages at the organ removal shift. This is because the organ removal shift is scheduled multiple days at once and no other surgeons are available all these days. Therefore, we have a shortage at the organ removal shift during all the days that must be scheduled for the same surgeon. We also see that most of the shortages occur at the OR task. This is because these shortages have the lowest costs of the shortages at tasks in the objective function. The surgeries during the OR task can be scheduled, therefore fewer surgeries can be scheduled when fewer surgeons are scheduled at the operating room task. The figure also shows that the maximum number of shortages during a day is five during day 26 and 89. In Figure 5.1 is shown that at day 26 only three surgeons are available and day 26 is the last Friday of an organ removal shift week. Day 89 also is on Friday and in this case only two surgeons are available. This is the first Friday of the organ removal shift week, but the organ removal shift cannot be scheduled at these surgeons because they are not available on Monday.

Figure 5.3 shows the shortages of the MOD certified surgeon. The maximum shortage is one, because only one MOD certified surgeon is needed. The shortage can only occur during an organ removal week, because no MOD certified surgeon is needed when the organ removal shift is not scheduled. The figure shows the shortages per day. We see that a shortage occurs in blocks (multiple consecutive days). This is because the organ removal shift is scheduled for multiple consecutive days at once. In the figure is shown that a block of shortages occurs four times. The last two are because no transplant surgeon is scheduled at the organ removal shift as shown in Figure 5.2. The other two times at least one transplant surgeon is scheduled at the organ removal shift, but no MOD certified surgeon is scheduled.

Figure 5.4 shows the shortages of the senior surgeons. Always one senior surgeon should be available. The figure shows the shortages per day. We see in this figure that three times a shortage occurs for a senior surgeon, during day 26, 85 and 86. During day 26 no $T2$ shift is scheduled and the $T1$ and Td shift are scheduled for surgeon 7 who is a junior surgeon. During days 85 and 86 surgeons 10 and 11 are scheduled at the transplant shifts and both surgeons are junior surgeons.

Figure 5.5 shows the shortages of the hpb surgeon. Always one hpb surgeon should be scheduled at the operating room during a week. When no hpb surgeon is scheduled at the operating room during a week, a shortage occurs. The figure shows the shortages per week. We see that we have a shortage during week 1, 4 and 13. We have three hbp surgeons, surgeons 1, 3 and 6. However, surgeon 3 cannot be scheduled at the OR task. Therefore, we only have two surgeons who are hpb surgeons and can be scheduled at the OR task. During week 1 both surgeons are on holiday and therefore it is not possible to schedule an hpb surgeon at the OR task this week. The same thing holds for week 13. During week 4 surgeon 1 is available. However, this surgeon is scheduled the whole week at the organ removal shift because this is the only surgeon (except for surgeon 10 who is also scheduled at the organ removal shift) who is available from Wednesday till Friday and also the only surgeon (except for surgeon 4 who is also scheduled at the organ removal shift) who is available from Monday till Wednesday for the organ removal shift. The organ removal shift is more important than the hpb surgeon scheduled at the OR task and therefore no OR task is scheduled for this surgeon during this week.

Figure 5.6 shows the distribution of the different tasks over the surgeons. We want to equally divide the total number of tasks over the surgeons and also the M shift, $T1$ shift, $T2$ shift and the OR task. We can see that these tasks are not equally divided. This can be explained by the days off, the non-clinical days and the number of surgeons available. For example, surgeon 9 is not available during this schedule. Therefore, no tasks are scheduled for this surgeon. Surgeon 7 is not available from day 31 till the end of the schedule. Therefore, surgeon 7 is scheduled at fewer tasks than other surgeons. When we look at the organ removal shift, we see that surgeon 6 is not scheduled at the organ removal shift while surgeon 11 is scheduled fourteen days at this shift. This can be explained by the fact that surgeon 6 has the standard non-clinical day at Wednesday. Therefore, two of the three organ removal shift blocks (consecutive days) cannot be scheduled for this surgeon, because these blocks contain the Wednesday. The block that can be scheduled for this surgeon is from Friday till Monday. However, this surgeon is only available two weekends when the organ removal shift must be scheduled. Surgeon 11 is scheduled the most at the organ removal shift. This surgeon is not available on Thursday and Friday and has the standard non-clinical on Wednesday. However, this surgeon can start with the organ removal shift on Friday and can end the organ removal shift on Wednesday. When we look at the $T1$ shift, we see that this shift is not scheduled for surgeons 3 and 12. This is because these surgeons cannot be scheduled at this shift. We also see large differences in the number of days scheduled at this shift per surgeon. Surgeons 10 and 11 are only scheduled five and six days, respectively, at this shift while surgeons 4 and 6 are scheduled twenty-one and twenty days, respectively, at this shift. This can be explained by the fact that surgeon 11 can never be scheduled at the week $T1$ shift, because this surgeon is not available at Thursday and Friday. Surgeon 10 also has multiple weeks one day off and both surgeons are scheduled much at the organ removal shift.

The number of days scheduled at the $T2$ shift varies from zero days till fifteen days per surgeon. This shift is not scheduled for surgeon 3, because this surgeon cannot be scheduled at the $T2$ task. The fact that surgeon 2 is scheduled the most at the $T2$ shift is because this surgeon is scheduled fewer days at the organ removal shift and the $T1$ shift. This surgeon therefore has more possibilities to be scheduled at the $T2$ shift.

For the OR task we see that surgeons 3, 7 and 11 are not scheduled at this task. This is because surgeon 3 cannot be scheduled at this task, surgeon 7 is not the whole schedule available and surgeon 11 is never available when this task should be scheduled. We also see that surgeon 12 is scheduled the most at this task. This can be explained by the fact that if this surgeon is available during the days that the OR task should be scheduled and no organ removal shift is scheduled for this surgeon, the only task that can be scheduled for this surgeon is the OR task.

Despite the fact that shortages occur, the made schedule fulfills all conditions for scheduling the different tasks. When the schedules made by hand are used, tasks are assigned to surgeons while the conditions are not fulfilled. Then, the surgeon has to complain that the schedule is not doable and a last minute solution has to be found. Therefore, we will look in Section 5.8 to the schedules made by hand.

Surgeon\Day	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	V	V	V	V	V			Ch	Ch	G/OR	Td/Ch	NCD				V	V	V	V	V	V
2	T2	T2/Hpb	or	NCD	Td	T2	T2	T2/Hpb	Hpb	Ch/Hpb	NCD	Td/Ch	t1	t1	FromT1	Td	V	V	V	V	
3	Hpb	NCD	Hpb	Hpb	Hpb			M	M	NCD/M	Hpb	Hpb			Hpb	Hpb	NCD	V	V	V	V
4	Ch	Ch	Ch	or	NCD			OC	T2	T2/OR	V	V	V		t1	t1	t1	t1	t1		
5	G/Poli	NCD	T2	T2	T2/OR			OC	NCD	M	M	G/M	G	G	G	NCD	Td	Td	T2/OR	T2	T2
6	V	V	V	V	V	V	V	t1	t1	t1	t1	t1			Td		NCD	or	or	G	G
7	t1	t1	t1	t1	t1			Td	NCD	M	M	M	T2	T2	OC	NCD	Ch/Hpb	Hpb	Ch/Hpb		V
8	FromT1	NCD	or	Td	M	M	M	M		NCD	T2/OR	T2/OR			Ch	Ch	NCD	Ch	Td	V	V
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	OC	NCD	Td	Ch	Ch	t1	t1	FromT1	NCD	Td	V	V	G		OC	NCD	T2/OR	T2	V	t1	t1
11	Td	Td	NCD	V	V/M	M	M	M	G/Td	V	V	V	G	G	T2	T2	NCD	V	V/M	M	M
12	V	V		or	V			M	M	M	or	V			V	V	V	or	V		
13																					

Surgeon\Day	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42
1	M	M	M	M	M	T2	T2	T2	Ch	NCD	or	Td/Ch			T2/Hpb	T2	T2/Hpb	Td	NCD	T2	T2
2	T2/Hpb	T2	V	V	V	t1	t1	FromT1	T2	T2/OR	NCD	V	V	V	V	Hpb	V	V	V	V	
3	V	V	V	V	V	V	V	Hpb	Hpb	NCD	Hpb	Hpb			M	M	NCD/M	Hpb	Hpb		
4	M	M	M	Ch	V	V	V	V	V	V	V	V	t1	t1	V	Td	Td/Ch	Ch	NCD	t1	t1
5	V	Hpb	T2/Hpb	T2/Hpb	V	G	G	G/Ch	NCD	Ch/Hpb	Ch	M	M	M	M	NCD	M	M	G/M	G	G
6	V	V	V	V	V	V		t1	t1	t1	t1	t1	G	G	Td/Ch	Ch	NCD	T2/OR	T2/Td/Ch	G	G
7	V	V	V	T1/Td	T1/Td			OC	NCD	Td	V	V	V	V	V	V	V	V	V	V	V
8	V	V	V	V	V	V	V	Td		NCD	T2	T2/OR			t1	t1	t1	t1	t1	G	G
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	T1/Td	Ch	M	M	M			V	NCD	or	Td	M	M	M	M	NCD	M	M	M		
11	M	T1/Td	T1/Td	V	V	G	G	OC	Td	NCD	V	V	T2	T2	M	M	NCD/M	V	V	G	G
12	V	V	V	V	V			V	V	V	or	V			V		V	V	V		
13																					

Surgeon\Day	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
1	t1	t1	t1	t1	t1			Td	T2/Hpb	T2/Hpb	Td	NCD			t1	t1	t1	t1	t1		
2	Td	Ch	Ch/Hpb	NCD	M	M	M	M	V	V	V	V			Td	V	Td	NCD	M	M	M
3	Hpb	Hpb	NCD	Hpb	Hpb			M	M	NCD/M	V	V	V	V	Hpb	Hpb	NCD	Hpb	Hpb		
4	FromT1		or	Td	NCD			t1	t1	t1	t1	t1			T2	T2	T2/OR	or	NCD	V	V
5	G	NCD	T2/OR	T2	T2/OR	t1	t1	FromT1	NCD	M	M	G/M	G	G	G	NCD	Ch/Hpb	Ch	Ch	t1	t1
6	G/Poli		NCD	or	Td	G	G	G/Ch	Ch	NCD	or	Td/Ch	T2	T2	Ch	Ch	NCD	T2	T2/OR		
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	T2	T2	NCD	or	M	M	M	M	Td	NCD	T2/OR	T2/Hpb	t1	t1	FromT1		NCD	or	Td		
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	Ch	NCD	Td	Ch	Ch	T2	T2	T2/Hpb	NCD	Td/Ch	Ch/Hpb	V	V	V	OC	NCD	or	Td	M	M	M
11	OC	G/Td	V	V	V	V		M	M	NCD/M	V	V	G	G	OC	Td	NCD	V	V	T2	T2
12	V	V	V		V			V		V	V	V			V	V	V		V		
13																					

Surgeon\Day	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84
1	Td/Ch	Ch	T2/Td/Hp	T2/OR	NCD					NCD	or	M	M	M	M	T2	T2/OR	Td	NCD	t1	t1
2	M	Td	V	V	V			Ch	Ch	Ch/Hpb	NCD	Ch	T2	T2	T2/Hpb	Hpb	Ch/Hpb	V	V	V	V
3	M	M	NCD/M	Hpb	Hpb			Hpb	Hpb	NCD	Hpb	Hpb			M	M	NCD/M	Hpb	Hpb	V	V
4	V	V	V	V	V	V	V	t1	t1	t1	t1	t1			Ch	Ch	Td	Ch	NCD	V	V
5	FromT1	NCD	M	M	G/M	G	G	G/Poli	NCD	T2/OR	T2	T2/OR	t1	t1	FromT1	NCD	M	M	G/M	G	G
6	t1	t1	t1	t1	t1			T2	T2	NCD	Ch	Td			t1	t1	t1	t1	t1	V	V
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	T2/Hpb	T2/Hpb	NCD	Ch	Td/Ch	t1	t1	FromT1		NCD	Td	M	M	M	M	Td	NCD	T2/OR	T2/Td/Ch	V	V
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	M	NCD	V	Td	T2/OR	T2	T2	Td	NCD	Td	or	or			Td	NCD	M	M	M	G	G
11	M	M	NCD/M	V	V	G	G	OC	Td	NCD	V	V			M	M	NCD/M	V	V	G	G
12	V		V	V	V			V	V	V		V			OC	V	or	or	V		
13																					

Surgeon\Day	85	86	87	88	89	90	91	92	93	94	95	96	97	98
1	V	V	V	V	V	V	V	M	M	M	or	NCD		
2	V	V	V	V	V	V	V	T2/Hpb	T2	T2/Hpb	NCD	T1/Td	t1	t1
3	V	V	V	V	V	V	V	M	M	NCD/M	Hpb	Hpb		
4	V	V	V	V	V	T2	T2	OC		T1/Td	T1/Td	NCD	T2	T2
5	G/Ch		t1	T1/Td	T1/Td			V	V	V	V	V	V	V
6	V	V	V	V	V	V	V	Ch		NCD	Ch	Ch		
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	V	V	V	V	V	V	V	T1/Td	t1	NCD	T2	T2/OR		
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	T2/Hpb	T2		T2/Hpb	T2/Hpb			V	V	V	V	V	V	V
11	T1/Td	t1		V	V	t1	t1	V	V	V	V	V	V	V
12	V	V	V	or	V			OC		or	or	or		
13														

Figure 5.1: The generated schedule of the different tasks during quarter 4 of 2018.

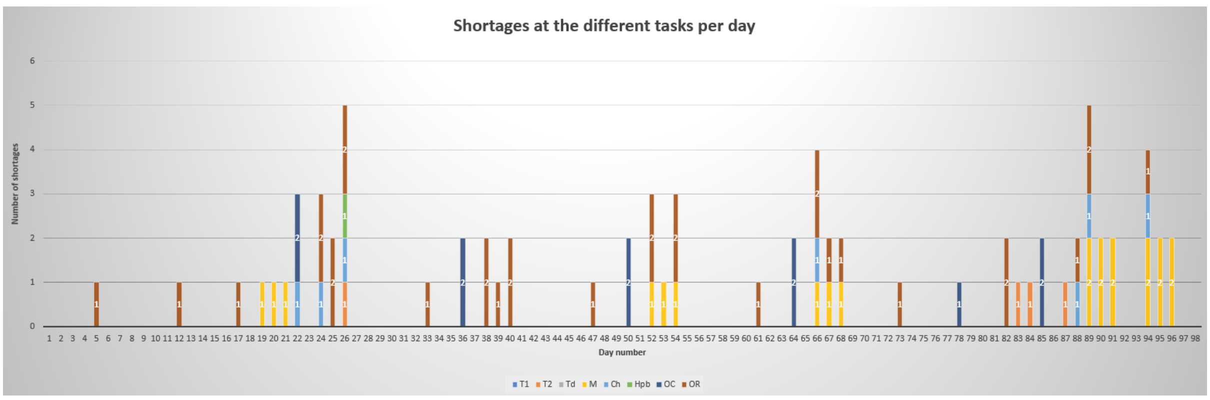

Figure 5.2: The shortages at the different tasks in the quarter schedule per day.

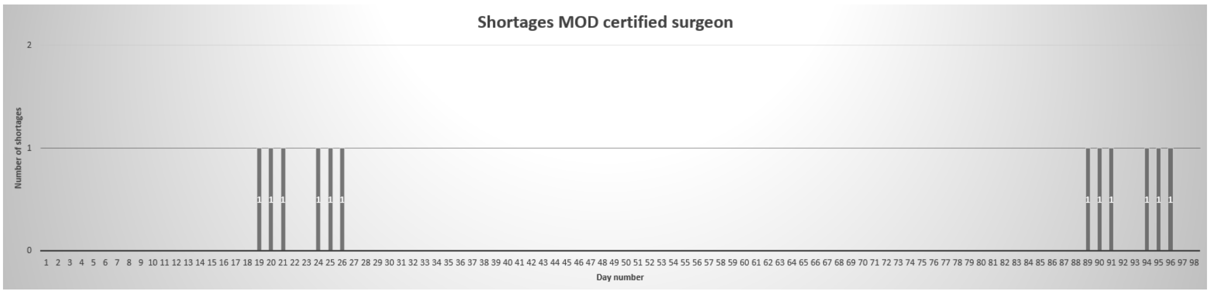

Figure 5.3: The shortages of the MOD certified surgeon in the quarter schedule per day.

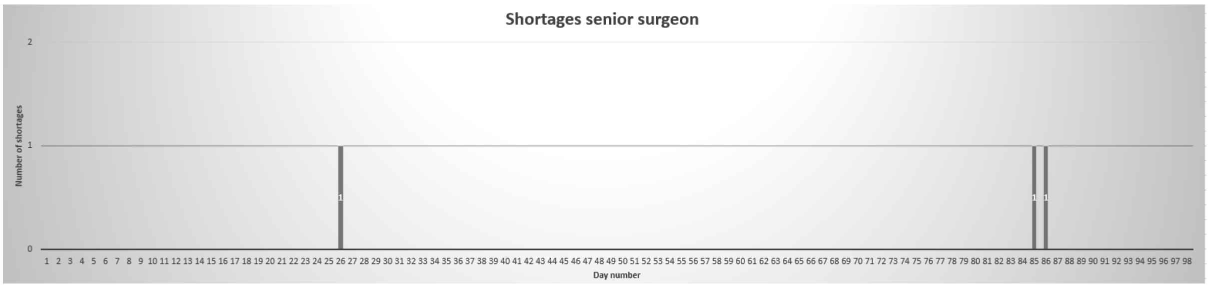

Figure 5.4: The shortages of the senior surgeon in the quarter schedule per day.

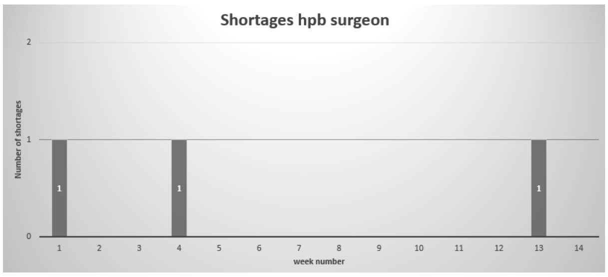

Figure 5.5: The shortages of the hpb surgeon in the quarter schedule per week.

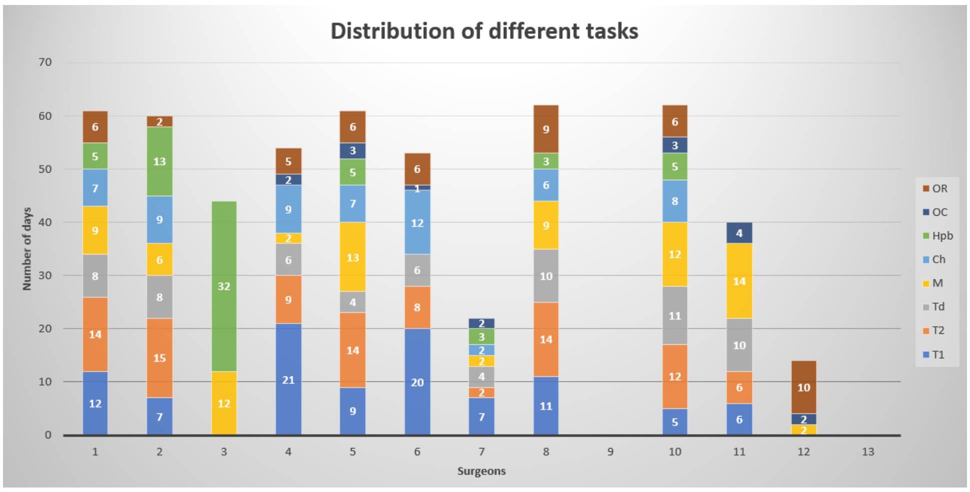

Figure 5.6: The distribution of the different tasks over the surgeon.

5.4 Case-study results when cooperate with Erasmus MC

The model described in Section 4.7.1 is programmed in AIMMS and solved with GUROBI, where we use the parameters as described in Section 5.2. Not all parameters are needed in this case. The organ removal shift is scheduled every week and therefore we do not need MW. We stopped the solver after about four hours. The objective value of the best solution found is $1\,431\,211.89$ and the optimality gap is equal to 0.92%. This means that it may be possible to find a better solution. However, this solution has an objective value not better than $1\,418\,104.97$. Therefore, the difference between the best solution found and the optimal solution is at most $13\,106.92$. We know the costs for shortages at different tasks from objective function (4.182). This means that no solution can be found with less shortages at the shifts than the shortages in the solution found, because the costs for a shortage are at least $20\,000$. It is possible that the optimal schedule has fewer shortages at the Ch, Hpb, OR or OC task. The quarter schedule is shown in Figure D.1 in Appendix D.1. The shortages that occur during the schedule at the different tasks are shown in Figure D.2 in Appendix D.1. Notice that we do not have costs for shortages of MOD certified surgeons, because we do not know which surgeons from the Erasmus MC are MOD certified. Therefore, this objective value can be lower than the other cases.

5.5 Case-study results when the organ removal shift is scheduled as done in the old situation

The model described in Section 4.7.2 is programmed in AIMMS and solved with GUROBI, where we use the parameters as described in Section 5.2. We stopped the solver after about four hours. The objective value of the best solution found is $1\,061\,467.12$ and the optimality gap is equal to 0.60%. This means that it may be possible to find a better solution. However, this solution has an objective value not better than $1\,054\,282.35$. Therefore, the difference between the best solution found and the optimal solution is at most $7\,184.77$. We know the costs for shortages at different tasks from objective function (4.208). This means that no solution can be found with fewer shortages than the shortages in the solution found, except one shortage fewer at the OR/OC task is possible. The costs for a shortage at the OR/OC task is $5\,000$ and the costs for shortages at other tasks is at least $10\,000$. The quarter schedule is shown in Figure D.3 in Appendix D.2. The shortages that occur during the schedule at the different tasks are shown in Figure D.4 in Appendix D.2.

5.6 Case-study results when the organ removal shift is scheduled the same as the $T2$ shift

The model described in Section 4.7.3 is programmed in AIMMS and solved with GUROBI, where we use the parameters as described in Section 5.2. We stopped the solver after about four hours. The objective value of the best solution found is $1\,241\,693.74$ and the optimality gap is equal to 0.20%. This means that it may be possible to find a better solution. However, this solution has an objective value not better than $1\,239\,210.35$. Therefore, the difference between the best solution found and the optimal solution is at most $2\,483.39$. We know the costs for shortages at different tasks from objective function (4.225). This means that no solution can be found with fewer shortages than the shortages in the solution found, because the costs for a shortage are at least $5\,000$. The quarter schedule is shown in Figure D.5 in Appendix D.3. The shortages that occur during the schedule at the different tasks are shown in Figure D.6 in Appendix D.3.

5.7 Compare cases

When we look at the objective values of the different possibilities of scheduling the organ removal shift, we see that the old structure has the lowest objective value. This can be explained by the fact that this structure has the lowest number of constraints for scheduling the organ removal shift. Therefore, we have more possibilities for scheduling the organ removal shift and therefore we have fewer times that the organ removal shift cannot be scheduled.

The second lowest objective value is when we schedule the organ removal shift by the same structure as the $T2$ shift. This can be explained by the fact that we have more constraints than when we schedule the organ removal shift using the old structure. However, we have more flexibility of scheduling the organ removal shift than using the fixed structure of the switching moments on Friday, Monday and Wednesday.

When the LUMC cooperates with the Erasmus MC we also have a lower objective value than when this cooperation does not exist. However, the fact that an MOD certified surgeon must be scheduled at the organ removal shift is not taken into account in this case. Therefore, no shortages can occur for the MOD certified surgeon.

5.8 Schedules made by hand

When we want to schedule the fourth quarter, we see that shortages occur at the different tasks. Therefore, we want to know how the schedules made by hand look like. For this we look at the schedules of the first and second quarter of 2018. We do not take into account the third quarter, because this was during the holiday and two surgeons were unexpectedly not available during this schedule. Therefore, this quarter is not representative. In Table 5.1 is shown that the schedules made by hand do not always fulfill the rules that should be fulfilled. During the first quarter schedule the M, $T1$, $T2$, Td, Ch and Hpb tasks were scheduled. During the second quarter schedule only the M, $T1$, $T2$ and Td task were scheduled. During these quarters the organ removal shift was scheduled by the old structure. Therefore this shift was easier to schedule.

Table 5.1: The number of times a rule is broken during the schedule of the first and second quarter of 2018.

Rule	Quarter 1 2018	
	Number of times	Percentage
$T1$ shift the whole week done by the same surgeon	1	8%
At least two consecutive days $T2$ shift	2	3%
No tasks on Monday after weekend $T1$ shift	3	23%
Ch task and M shift cannot be combined	1	3%
$T2/M/Ch$ not possible for one surgeon	1	3%
Td and M cannot be combined	1	3%
$T2$ and M cannot be combined	2	5%
No weekend $T1$ shift before week $T1$ shift	1	8%
Weekend shift done by the same surgeon	1	3%
Rule	**Quarter 2 2018**	
	Number of times	**Percentage**
$T1$ shift the whole week done by the same surgeon	1	8%
At least two consecutive days $T2$ shift	15	23%
$T2$ and M cannot be combined	1	3%
Weekend shift done by the same surgeon	3	8%

Chapter 6

Generic mathematical model for equally dividing orders

Now we want to include the number of surgeries done and we want to distribute this as equally as possible over the surgeons per year. To do this, we will first look at a more generic problem: on fixed moments a schedule must be made to divide different tasks over different persons. The length of the schedule is fixed. So, it is possible that more tasks must be scheduled for one person, because more time blocks (e.g., days) must be scheduled during one schedule. The moments that the schedule must be made and the length of one schedule are known. During a task an order can arrive with a certain probability. This probability can differ per task and when the task is scheduled. The tasks should be scheduled in such a way that at the end of the horizon the number of orders is divided equally over the persons who are scheduled at the tasks. This is modelled as a Stochastic Dynamic Program.

This chapter is organized as follows: Section 6.1 describes what a stochastic dynamic program is, Section 6.2 shows the notation that is used, Section 6.3 describes the assumptions made for modelling the process, Section 6.4 derives the transition probabilities, Section 6.5 describes the upper boundary used for the state space, Section 6.6 proves the existence of a deterministic Markovian policy which is optimal, Section 6.7 describes the dynamic program that can be used to solve the SDP exactly and Section 6.8 describes how we can reduce the number of states in the state space.

6.1 Stochastic Dynamic Programming (SDP)

An SDP is a process that is observed at a finite number of discrete time points. At each of these time points the system is at any of the possible states. After observing the state of the system an action should be chosen. When an action is chosen, the next state of the system for the subsequent point in time is determined according to the transition probabilities. These transition probabilities depend on the current state and the chosen action at the current state. The transition probabilities cannot depend on previous states. The chosen actions have immediate costs. At the subsequent point in time, the decision-maker faces a similar problem. However, the state of the system can differ and the set of actions to choose from can differ. This is done for a finite number of points in time, because a horizon is determined. The decisions are made during time points before the end of the horizon. At the end of the horizon we have costs of being in the state at the end of the horizon. More about SDP can be found in Puterman [2005].

6.2 Notation

For modelling the arrival of orders during tasks, a Poisson Proces is used. A Poisson Process is a counting proces. The definition of a counting process is shown in Definition 6.2.1. The definition of a Poisson process is shown in Definition 6.2.2.

Definition 6.2.1. [Ross, 2010] A stochastic process $\{N(t), t \geq 0\}$ is said to be a counting process if $N(t)$ represents the total number of events that occur by time t.
Definition 6.2.2. [Ross, 2010] The counting process $\{N(t), t \geq 0\}$ is said to be a Poisson process having rate λ, $\lambda > 0$ if

1. $N(0) = 0$
2. The process has independent increments (the number of events that occur in disjoint time intervals are independent)
3. The number of events in any interval of length t is Poisson distributed with mean $\lambda \cdot t$. That is, for all $s, t \geq 0$

$$\mathbb{P}\Big(N(t+s) - N(s) = n\Big) = e^{-\lambda \cdot t}\frac{(\lambda \cdot t)^n}{n!}, \qquad n = 0, 1, \ldots$$

Now we will describe how we model the process as an SDP and which notation is used. By process we mean scheduling tasks over persons, where orders arrive during the different tasks and we want to equally divide these orders over the persons at the end of the horizon.

- **Persons**
The persons who can be scheduled at the different tasks are notated as follows:

$$\begin{aligned} P &= \{1, 2, \ldots, J\} \\ p &\in P \end{aligned}$$

- **Tasks**
Different tasks must be scheduled over the persons. The notation of the different tasks is shown below.

$$\begin{aligned} TA &= \{1, \ldots, M\} \\ t &\in TA \end{aligned}$$

- **Blocks**
During one schedule, different time blocks must be scheduled. An example of a time block is a day. In this case, the tasks are scheduled per day and during one schedule multiple days can be scheduled. The notation used is shown below.

$$\begin{aligned} D &= \{1, \ldots, B\} \\ d &\in D \end{aligned}$$

The total number of different time blocks that must be scheduled is shown below. These time blocks are all the time blocks during the different schedules.

$$\begin{aligned} TD &= \{1, \ldots, TB\} \\ td &\in TD \end{aligned}$$

- **Decision epochs**
The moment an action should be taken is a decision epoch. In this case, the action is to make a schedule. This is shown in Figure 6.1. So, at decision epoch 1 schedule 1 is made. The last decision epoch is at N, but in this case no decisions is made, because this is the end of the horizon. During this decision epoch we can assign costs of being in a state at the end of the horizon.

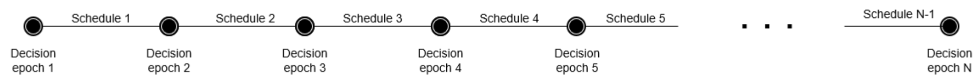

Figure 6.1: The decision epochs and schedules

The following notation is used for the decision epochs:

$$\begin{aligned} T &= \{1, 2, \ldots, N\} \\ n &\in T \end{aligned}$$

- **States**
The state of one person is the number of orders this person has completed before decision epoch n ($\sigma_{p,n}$). The state of the system is the state of all persons who can be scheduled at the tasks. This is notated as follows:

$$\sigma_n = (\sigma_{1,n}, \ldots, \sigma_{J,n})$$

The state space are all the states that can be reached. The states in the state space must fulfill the following conditions:

– The number of orders completed per person is non-negative

– The number of orders completed per person before the first schedule is equal to zero

When all states that can be reached are used, we have an infinite number of states. An upper boundary is used to decrease the number of states (see Section 6.3). How to determine the upper boundary is described in Section 6.5. The notation is shown below.

$$\begin{aligned} S_1 &= \{\sigma_1 = (\sigma_{1,1}, \dots, \sigma_{J,1}) \mid \sigma_{p,1} = 0 \qquad \forall p \in P\} \\ S_n &= \{\sigma_n = (\sigma_{1,n}, \dots, \sigma_{J,n}) \mid 0 \leq \sigma_{p,n} \leq U_{p,n} \qquad \forall p \in P\} \qquad \forall\, 1 < n \leq N \end{aligned}$$

- **Decisions**
  Each decision epoch a new schedule must be made. The decisions that must be made are which tasks are scheduled for which person and when, with the conditions shown below.

  – Each person has at most one task per time block

  – All tasks are scheduled exactly once during a time block

  The notation used for this is shown below.

$$\begin{aligned} A &= \{x_{p,t,d} \qquad \forall p,t,d \mid x_{p,t,d} \in \{0,1\}, \sum_t x_{p,t,d} \leq 1, \sum_p x_{p,t,d} = 1\} \\ x_{p,t,d} &= \begin{cases} 1 & \text{if person } p \text{ is scheduled at task } t \text{ at block number } d \\ 0 & \text{otherwise} \end{cases} \\ a &\in A \end{aligned}$$

- **Direct costs**
  The costs that must be minimized are the deviation of the line with the number of orders equally divided over the persons. When we start scheduling, we do not know how may orders arrive before reaching the end of the horizon. So, we do not know which point of the line with the number of orders equally divided over the persons we want to reach. So, we want to approach the line as good as possible. In Figure 6.2 this is shown when two persons are scheduled. The red points have the same costs, but the green square is cheaper. It does not matter how we reach the line. So, whether we had large differences between the number of orders completed per person before the end of the horizon or the differences were small all the time does not matter. This is shown in Figure 6.3 when two persons are scheduled. The purple dots are the states at decision epochs for one path with endpoint the yellow square. The blue triangles are the states at decision epochs for another path with endpoint the yellow square. Both paths have the same endpoint. Therefore, these paths have the same costs. However, before the yellow square, they have nothing in common.

  The costs of being in a state during a decision epoch is zero when the decision epoch is not the last decision epoch. When we are at the end of the horizon the costs are equal to the deviation from the line where the orders are divided equally. The notation is shown below.

$$c_n(\sigma_n) = \begin{cases} \sum_k \left(\sigma_{k,n} - \frac{\sum_j \sigma_{j,n}}{J}\right)^2 & \text{if } n = N \\ 0 & \text{otherwise} \end{cases}$$

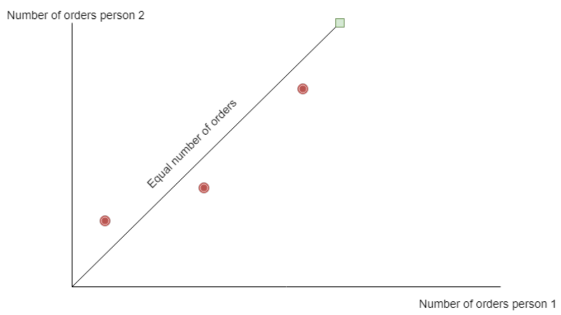

Figure 6.2: The line with equal number of orders when the tasks are scheduled over two persons.

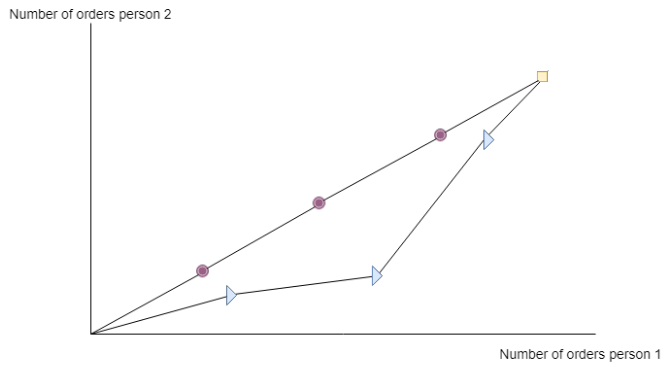

Figure 6.3: Different paths with the same endpoint.

- **Transition probabilities**
  The transition probabilities between two states depend on the arrival probabilities of orders during the tasks and the chosen action (the schedule). The derivation of the transition probabilities is shown in Section 6.4 and the transition probabilities are shown below.

$$\mathbb{P}\Big(i \mid \sigma_n, a\Big) \quad = \quad \begin{cases} \prod_{p:i_p<U_{n+1,p}} e^{-\sum_d \sum_t x_{p,t,d}\cdot\lambda_{t,d}} \frac{(\sum_d \sum_t x_{p,t,d}\cdot\lambda_{t,d})^{i_p-\sigma_{p,n}}}{(i_p-\sigma_{p,n})!} \cdot \\ \prod_{p:i_p=U_{n+1,p}} \Big(1 - \sum_{m=\sigma_{p,n}}^{i_p-1} e^{-\sum_d \sum_t x_{p,t,d}\cdot\lambda_{t,d}} \frac{(\sum_d \sum_t x_{p,t,d}\cdot\lambda_{t,d})^{m-\sigma_{p,n}}}{(m-\sigma_{p,n})!}\Big) & \text{if } i_p \geq \sigma_{n,p} \quad \forall p \\ 0 & \text{otherwise} \end{cases}$$

- **Optimal value function**
  The goal is to minimize the difference between the state at the horizon and the line where the number of completed orders is equally divided over the persons. This can be done by minimizing the expected future costs, because the costs determine this difference. At the end of the horizon we have the direct costs of being in a state at the end of the horizon.

$$\begin{aligned} f_n(\sigma_n) &= \text{min}_{a\in A} \sum_{i\in S_{n+1}} \mathbb{P}(i \mid \sigma_n, a) \cdot f_{n+1}(i) \qquad \forall n < N \\ f_N(\sigma_N) &= c_N(\sigma_N) \end{aligned}$$

6.3 Assumptions

Some assumptions must be made to use an SDP and solve the SDP. Assumption 1 gives us the distribution of the arriving orders. The orders arrive independent of each other, therefore a Poisson Process is used.

Assumption 1. *Orders arrive during tasks t on time block d according to a Poisson Process with rate $\lambda_{t,d}, \forall t, d$ per time block. The arrival process is independent of the arrival processes at other tasks and other time blocks. This process is also independent of the person scheduled at the task.*

Assumption 2 is used to solve the SDP. To be able to solve the SDP, the state space must be finite. Therefore, Assumption 2 is needed.

Assumption 2. *The number of orders that can be completed by a person during a decision epoch has an upper boundary. Therefore, at every decision epoch a finite number of states can be reached.*

Assumption 3 is used to determine the transition probabilities.

Assumption 3. *All time blocks have the same length.*

6.4 Derivation transition probabilities

For the derivation of the transition probabilities we will first look at the arrival process of orders during tasks. Then we will look at the arrival process of orders during a time block per person. After that, we will look at the arrival process of orders during a schedule per person. Last, we will use the other parts of this section to derive the transition probabilities.

Arrival process orders during tasks

We know the distribution of the arrival process of orders during the different tasks from Assumption 1, Possion Process with arrival rate $\lambda_{t,d}, \forall t, d$ per time block. We also know from this assumption that the arrival process of orders during a task on a time block d is independent of the arrival process of orders during other tasks and other time blocks. This process is also independent of the person scheduled at the task.

Arrival process orders during time block per person

Now we will derive the arrival process of orders during a time block per person. By doing this, we now derive the arrival process when more tasks can be scheduled during one time block for the same person. Therefore, this is more generic than the decisions described earlier. However, we know the transition probabilities when we do not include the condition that each person has at most one task per time block. This is done, because this condition does not hold in the LUMC context.

We can derive the probability that a certain number of orders arrive during a time block for a particular person given the schedule, from the arrival processes of orders during tasks. This derivation is shown below. The total number of orders that arrive during a time block is equal to the sum over the tasks of the orders that arrive during a task that is scheduled during the time block. This is done with the first equal sign. The second equal sign is because we know which tasks are scheduled during a day for a person when we know the chosen action. The third equal sign will be explained below and the last equal sign is because we know the probability mass function of a Poisson distribution.

$$\begin{aligned}
\mathbb{P}\Big(\text{\# orders at time block } d \text{ for person } p = k \mid a_n\Big) &= \\
\mathbb{P}\Big(\sum_{t:x_{p,t,d}=1} \text{\# orders arriving at task } t \text{ during time block } d \text{ for person } p = k \mid a_n\Big) &= \\
\mathbb{P}\Big(\sum_{t:x_{p,t,d}=1} \text{\# orders arriving at task } t \text{ during time block } d \text{ for person } p = k \mid x_{p,t,d} \quad \forall p, t, d\Big) &= \\
\mathbb{P}\Big(\text{\# arrivals in Poisson Process with rate } \sum_t x_{p,t,d} \cdot \lambda_{t,d} = k) &= \\
e^{-\sum_t x_{p,t,d} \cdot \lambda_{t,d}} \frac{(\sum_t x_{p,t,d} \cdot \lambda_{t,d})^k}{k!} &
\end{aligned}$$

The third row is a sum of independent Poisson Processes (Assumption 1). The sum of independent Poisson Processes is a new Poisson process with an arrival rate equal to the sum of the arrival rates of the Poisson Processes, see Theorem 6.4.1.

Theorem 6.4.1. *[Durrett, 2012] Suppose that* $\{N_1(t), t \geq 0\}, \{N_2(t), t \geq 0\}, \ldots, \{N_k(t), t \geq 0\}$ *are independent Poisson processes with respective rates* $\lambda_1, \lambda_2, \ldots, \lambda_k$*. Let* $N(t) = N_1(t) + N_2(t) + \ldots + N_k(t), t \geq 0$ *Then the merged process* $\{N(t), t \geq 0\}$ *is a Poisson process with rate* $\lambda = \lambda_1 + \lambda_2 + \ldots + \lambda_k$*.*

Arrival process orders during schedule per person

We now know the probability of a certain number of orders arrive during a time block for a person. Now we can derive the probability of a certain number of orders arrive during a schedule for a person. Notice that the number of orders that arrive during a schedule for a certain person is equal to the sum of the number of orders that arrive during the time blocks of this schedule. Now we can derive this probability in the same way as for the number orders during a task. Property 3 of Definition 6.2.2 is used to add Poisson processes during different time blocks. From this property we know that it does not matter which time block the task is scheduled, only the length of the time block matters. We also know that all time blocks have the same length (Assumption 3). However, the time block does determine the arrival rate of orders during a task.

$$
\begin{aligned}
\mathbb{P}\Big(\text{\# orders during the schedule for person } p = k \mid \text{schedule}\Big) &= \\
\mathbb{P}\Big(\sum_d \sum_{t:x_{p,t,d}=1} \text{\# orders arriving at task } t \text{ during time block } d \text{ for person } p = k \mid a\Big) &= \\
\mathbb{P}\Big(\sum_d \sum_{t:x_{p,t,d}=1} \text{\# orders arriving at task } t \text{ during time block } d \text{ for person } p = k \mid x_{p,t,d} \quad \forall p,t,d\Big) &= \\
\mathbb{P}\Big(\text{\# arrivals in Poisson Process with rate } \sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d} = k \mid x_{p,t,d} \quad \forall p,t,d) &= \\
e^{-\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d}} \frac{(\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d})^k}{k!}
\end{aligned}
$$

Transition probabilities

The derivation of the transition probabilities is shown below. First we will look at the case that the number of completed orders at the end of the schedule higher is than before the schedule for all persons. Then we will look at the case that the number of completed orders at the end of the schedule is lower than before the schedule for some person.

The derivation of the transition probabilities when the number of completed orders at the end of the schedule is higher than before the schedule for all persons, is shown below. The second equal sign in the derivation holds, because the arrival processes during tasks are independent of the arrival processes during other tasks, time blocks and the person scheduled at the task. From this follows that the arrival process of the orders for one person is independent of the arrival process of orders for another person when the schedule is known. The third and fourth equal signs have the same reasoning as the arrival process of orders during a schedule per person.

$$
\begin{aligned}
\mathbb{P}\Big(i \mid \sigma_n, a\Big) &= \mathbb{P}\Big(\text{\# orders during schedule for person } p = i_p - \sigma_{p,n} \quad \forall p \mid \sigma_n, a\Big) \\
&= \prod_p \mathbb{P}\Big(\text{\# orders during schedule for person } p = i_p - \sigma_{p,n} \mid \sigma_n, a\Big) \\
&= \prod_p \mathbb{P}\Big(\text{\# arrivals in Poisson Process with rate } \sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d} = i_p - \sigma_{p,n} \mid x_{p,t,d} \quad \forall p,t,d) \\
&= \prod_p e^{-\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d}} \frac{(\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d})^{i_p - \sigma_{p,n}}}{(i_p - \sigma_{p,n})!}
\end{aligned}
$$

Now we will look at the case that the number of completed orders after the schedule is lower than before the schedule for some person. This means that for some person the number of orders done decreases. This is not possible, so the transition probability in this case is equal to zero.

$$
\mathbb{P}\Big(i \mid \sigma_n, a\Big) = 0
$$

When we combine both cases, we have the following transition probabilities.

$$
\mathbb{P}\Big(i \mid \sigma_n, a\Big) = \begin{cases} \prod_p e^{-\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d}} \frac{(\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d})^{i_p - \sigma_{p,n}}}{(i_p - \sigma_{p,n})!} & \text{if } i_p \geq \sigma_{n,p} \quad \forall p \\ 0 & \text{otherwise} \end{cases}
$$

These transition probabilities can be used, but we have an upper boundary for the state space and the transition probabilities must sum to one for all possible number of orders per person. A person does not have other possibilities of a number of orders completed than the states. Therefore, the state in which the number of orders completed is equal to the upper boundary has a transition probability for that person equal to one minus the sum of the transition probabilities to the other numbers of completed orders. The transition probabilities used are shown below. We define the empty product as one and the empty sum as zero.

$$
\mathbb{P}\Big(i \mid \sigma_n, a\Big) = \begin{cases} \prod_{p:i_p < U_{n+1,p}} e^{-\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d}} \frac{(\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d})^{i_p - \sigma_{p,n}}}{(i_p - \sigma_{p,n})!} \cdot \\ \prod_{p:i_p = U_{n+1,p}} \Big(1 - \sum_{m=\sigma_{p,n}}^{i_p - 1} e^{-\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d}} \frac{(\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d})^{m - \sigma_{p,n}}}{(m - \sigma_{p,n})!}\Big) & \text{if } i_p \geq \sigma_{n,p} \quad \forall p \\ 0 & \text{otherwise} \end{cases}
$$

6.5 Upper boundary state space

An upper boundary for the state space is used to have a finite number of states (Assumption 2). We determine for each person an upper boundary of number of orders. This is done by determining the maximal arrival rate of orders during a task that can be scheduled for the person before the decision epoch. So, the tasks that can be scheduled for a person with the highest arrival rate are used for this. The upper boundary used for this person is the maximum number of orders that can arrive during a Poisson Process with this maximal arrival rate with a probability greater than q. The derivation of this upper boundary is shown below.

$$\begin{aligned}
&\mathbb{P}\Big(i_p \text{ orders before decision epoch } n \text{ during tasks that can be scheduled for person } p\Big) &=\\
&\mathbb{P}\Big(i_p \text{ orders during Poisson Process with rate } \sum_{td:\text{time block before schedule } n} \max_t\{TS_{p,t}\cdot\lambda_{t,td}\}\Big) &=\\
&e^{-\sum_{td:\text{time block before schedule } n}\max_t\{TS_{p,t}\cdot\lambda_{t,td}\}}\frac{(\sum_{td:\text{time block before schedule } n}\max_t\{TS_{p,t}\cdot\lambda_{t,td}\})^{i_p}}{i_p!}
\end{aligned}$$

where

$$TS_{p,t} = \begin{cases} 1 & \text{if person } p \text{ can be scheduled at task } t \\ 0 & \text{otherwise} \end{cases}$$

The states taken into account are the states with the number of orders per person lower than the upper boundary. The upper boundary per person is the biggest number of orders with a probability higher than q. This is shown below.

$$\operatorname{argmax}_{i_p}\{e^{-\sum_{td:\text{time block before schedule } n}\max_t\{TS_{p,t}\cdot\lambda_{t,td}\}}\frac{(\sum_{td:\text{time block before schedule } n}\max_t\{TS_{p,t}\cdot\lambda_{t,td}\})^{i_p}}{i_p!} \geq q\}$$

Only one state can be reached at decision epoch 1. This state is the state where all persons have done zero orders. The upper boundary for the next decision epoch is described below. To determine the maximal arrival rate before decision epoch 2 per person, the maximal arrival rate per time block during schedule one is determined. For each time block we determine the tasks that can be scheduled for this person with the highest arrival rate. When we sum these maximal arrival rates per time block over the time blocks during schedule 1, we get the maximal arrival rate during schedule 1 for this person. Now we can determine the upper boundary for this person at decision epoch 2. This upper boundary is equal to the maximal number of orders that can arrive during a Poisson Process with the determined maximal arrival rate with probability greater or equal than q.

When all numbers of arriving orders have a probability less than q, the upper boundary used is the arrival rate rounded up to the nearest integer. This is done because the probability mass function of the Poisson distribution has a top at the number of events equal to the arrival rate. The upper boundary used is shown below.

$$\begin{aligned}
U_{p,n} = \max\Big\{&\operatorname{argmax}_{i_p}\{e^{-\sum_{td:\text{time block before schedule } n}\max_t\{TS_{p,t}\cdot\lambda_{t,td}\}}\cdot\\
&\frac{(\sum_{td:\text{time block before schedule } n}\max_t\{TS_{p,t}\cdot\lambda_{t,td}\})^{i_p}}{i_p!}\geq q\},\\
&\Big\lceil\sum_{td:\text{time block before schedule } n}\max_t\{TS_{p,t}\cdot\lambda_{t,td}\}\Big\rceil\Big\}
\end{aligned}$$

6.6 Existence of a deterministic Markovian policy which is optimal

We want to find the optimal action for each state of the system which only depends on history by the current state of the system and the action is chosen with certainty. This means at every decision epoch for every state one of the possible schedules is chosen as action and this is optimal. Such a decision rule is called a deterministic Markovian policy.

Theorem 1. *[Puterman, 2005]*
Assume S is finite or countable and that

1. *A_s is finite for each $s \in S$, or*

2. A_s *is compact,* $c_n(s,a)$ *is continuous in* a *for each* $s \in S$*, there exists an* $M < \infty$ *for which* $|c_n(s,a)| \le M$ *for all* $a \in A_s$*, and* $\mathbb{P}_n\Big(j \mid s,a\Big)$ *is continuous in* a *for each* $j \in S$ *and* $s \in S$ *and* $n = 1, 2, \dots, N$*, or*
3. A_s *is compact ,* $c_n(s,a)$ *is upper semicontinuous in* a *for each* $s \in S$*, there exists an* $M < \infty$ *for which* $|c_n(s,a)| \le M$ *for all* $a \in A_s$*,* $s \in S$*, and for* $n = 1, 2, \dots, N$*.*

Then there exists a deterministic Markovian policy which is optimal.

Now we know some conditions for the existence of a deterministic Markovian policy, we need the following theorem to say something about the size of the action space.

Theorem 2. *[Grimaldi, 2003]*
The number of permutations of k *objects from* n *elements without repetition is equal to*

$$\frac{n!}{(n-k)!}$$

Now we want to prove that in our case a deterministic Moarkovian policy is optimal. This is done with Theorem 3.

Theorem 3. ***(deterministic Markovian policy)***
There exists a deterministic Markovian policy which is optimal for the problem formulated in Sections 6.2 – 6.5

Proof. The number of states is finite by Assumption 2. Now we want to prove that the action state is finite for each $s \in S$ (Theorem 1.1). The action space is the same for each decision epoch. Therefore, we only have to prove that this action space is finite.

The number of actions in the action space when one block is scheduled per schedule is equal to the number of permutations without repetition of P (number of persons) elements (assign each person to a task). From Theorem 2 follows that this is equal to $P!$. When more blocks are scheduled during one schedule, for each block $P!$ different schedules can be chosen. Hence, the number of actions in the action space is equal to

$$\Big(P!\Big)^B$$

where $P < \infty$ and $B < \infty$. Therefore, the action space is finite. From Theorem 1.1. follows that in this case a deterministic Markovian policy exists which is optimal. □

A deterministic Markovian policy can be found by using dynamic programming. This is done in the next section.

6.7 Dynamic Programming (DP)

This model can be solved using dynamic programming. The idea of dynamic programming is to solve the problem backwards. The costs at the horizon ($n = N$) are known, so the value of the objective function at the horizon is known for every state. Then we can determine the optimal action at decision epoch $n = N - 1$ for every state by finding the action that leads to the minimum expected costs. This is the action where the expected costs are equal to the optimal value function. This can be done for all decision epochs. Algorithm 1 shows this procedure.

```
Set n = N
for s ∈ S_N do
|   f_N(s) = c_N(s)
end
while n > 1 do
    n ← n − 1
    for s ∈ S_n do
        f_n(s) = min_{a∈A} {c_n(s) + Σ_{i∈S} P(i | s,a) · f_{n+1}(i)}
        A*_{s,n} = argmin_{a∈A} {c_n(s) + Σ_{i∈S} P(i | s,a) · f_{n+1}(i)}
    end
end
```

Algorithm 1: The dynamic programming algorithm.

6.8 Reduce state space

When we use the states as defined in Section 6.2, the number of states in the state space is equal to

$$\prod_{p \in P} (U_{p,n} + 1)$$

The state of a person can take the values $0, \ldots, U_{p,n}$. So, we have $U_{p,n} + 1$ possible values for the state of person p. When we want to know the number of possibilities for the state of the system, we multiply the possibilities for the state of the persons. To give an indication of the number of states, two examples are shown below.

- When we have for example five persons who could have completed four orders each, the number of states is equal to

$$5^5 = 3\,125$$

- When we have ten persons who could have completed fifty orders each, the number of states is equal to

$$51^{10} = 119\,042\,423\,827\,613\,001$$

Therefore, we want to reduce the state space. First we will determine the state as the difference between the number of orders completed. This is done in Section 6.8.1. In Section 6.8.2 we will order the states lexicographically.

6.8.1 State equal to the difference between the number of orders completed

We want to distribute the number of orders completed as equally as possible over the persons. Therefore, it does not matter how many orders a person has completed. The difference with the other persons only matters. Therefore, the states are reformulated as follows.

The state of one person is the number of orders this person has completed more than the person with the least number of orders completed. The state is the state of all persons who can be scheduled at the tasks. This is notated as follows:

$$\sigma_n = (\sigma_{1,n}, \ldots, \sigma_{J,n})$$

The state space are all the states that can be reached. The states must fulfill the following conditions:

- The number of orders completed per person is non-negative
- The number of orders completed per person before the first schedule is equal to zero
- The number of orders completed per person more than the person with the least number of orders completed is non-negative

When all states that can be reached are used, we have an infinite number of states. An upper boundary is used to decrease the number of states (Assumption 2). The upper boundary used in this case is the same as described in Section 6.5.

$$\begin{aligned} S_1 &= \{\sigma_1 = (\sigma_{1,1}, \ldots, \sigma_{J,1}) \mid \sigma_{p,1} = 0 \qquad \forall p \in P\} \\ S_n &= \{\sigma_n = (\sigma_{1,n}, \ldots, \sigma_{J,n}) \mid 0 \leq \sigma_{p,n} \leq U_{p,n} \qquad \forall p \in P, \exists p \text{ st } \sigma_{p,n} = 0\} \qquad \forall 1 < n \leq N \end{aligned}$$

By using this definition all vectors shown below belong to the same state (the state with at least one element equal to zero), while these vectors were different states in the state space described in Section 6.2.

$$\begin{array}{c} \big(\sigma_{1,n}, \quad \ldots \quad , \sigma_{J,n}\big) \\ \big(\sigma_{1,n} + 1, \quad \ldots \quad , \sigma_{J,n} + 1\big) \\ \cdot \\ \cdot \\ \cdot \\ \big(\sigma_{1,n} + k, \quad \ldots \quad , \sigma_{J,n} + k\big) \end{array}$$

where $k \in \mathbb{N}$ and $\sigma_{p,n} + k \leq U_{p,n} \quad \forall p$.

When we use the states as defined above, also the transition probabilities change. This is because we now have more possibilities to reach a state. The transition probabilities are shown below. The transition probabilities are almost the same, but now we can also reach a state $(i_{1,n}, \ldots, i_{J,n})$ from $(\sigma_{1,n}, \ldots, \sigma_{J,n})$ when $i_{p,n} - \sigma_{p,n} + k$ orders arrive for all persons p, where $k \in \mathbb{N}$ and $\sigma_{p,n} + k \leq U_{p,n+1} \quad \forall p$.

When the original state space is used, state $(i_{1,n}, \ldots, i_{J,n})$ can only be reached from state $(\sigma_{1,n}, \ldots, \sigma_{J,n})$ when $i_{p,n} - \sigma_{p,n}$ orders arrive for all persons p.

$$\mathbb{P}\Big(i \mid \sigma_n, a\Big) = \sum_{\substack{k \in \mathbb{N}, \\ i_p + k \leq U_{p,n+1} \forall p, \\ i_p + k \geq \sigma_{p,n} \quad \forall p}} \prod_{p: i_p + k < U_{n+1,p}} e^{-\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d}} \frac{(\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d})^{i_p + k - \sigma_{p,n}}}{(i_p + k - \sigma_{p,n})!} \cdot$$
$$\prod_{p: i_p + k = U_{n+1,p}} \Big(1 - \sum_{m = \sigma_{p,n}}^{i_p + k - 1} e^{-\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d}} \frac{(\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d})^{m - \sigma_{p,n}}}{(m - \sigma_{p,n})!}\Big)$$

When we use the states as defined above, the number of states in the state space is equal to

$$\prod_{p \in P} (U_{p,n} + 1) - \prod_{p \in P} U_{p,n}$$

This can be explained by the fact that one of the persons should have state zero, because we could subtract more from the states for all persons otherwise. Therefore, all states that do not have a zero for any person are now not part of the state space. The number of states in this case for the examples are shown below.

- When we have for example five persons who could have completed four orders, each the number of states is equal to

$$5^5 - 4^5 = 2\,101$$

- When we have ten persons who could have completed fifty orders, each the number of states is equal to

$$51^{10} - 50^{10} = 21\,386\,173\,827\,613\,001$$

The number of states is still big. Therefore, we want to reduce the number of states further by ordering the states lexicographically.

6.8.2 Order state lexicographically

When we use Assumptions 4 and 5, we can further reduce the state space. This is because it does not matter which person is scheduled with these assumptions. When in the state two persons are switched, the optimal action remains the same, only the schedule from these two persons is switched.

Assumption 4. *Every person can be scheduled at each task.*
Assumption 5. *Every person is always available to be scheduled at each task.*

The definition of the states in this case is shown below. The state of one person is the number of orders this person has completed more than the person with the least number of orders completed. The state is the state of all persons who can be scheduled at the tasks lexicographic ordered. This is notated as follows:

$$\sigma_n = (\sigma_{1,n}, \ldots, \sigma_{J,n})$$
$$\text{where} \qquad \sigma_{1,n} \geq \sigma_{2,n} \geq \ldots \geq \sigma_{J,n}$$

The state space are all the states that can be reached. The states must fulfill the following conditions.

- The number of orders completed per person is non-negative
- The number of orders completed per person before the first schedule is equal to zero
- The number of orders completed per person more than the person with the least number of orders completed is non-negative

- The number of completed orders of person p is at least the number of completed orders of persons $p+1$ and is at most the number of completed orders of person $p-1$

When all states that can be reached are used, we have an infinite number of states. An upper boundary is used to decrease the number of states (Assumption 2). The upper boundary used in this case is the same as described in Section 6.5, except for the last element of the state. This element should be equal to zero. When this element is not equal to zero, we can subtract this number from all elements. In this case the upper boundary is not dependent on the person as in Section 6.5, because we assume that all persons can be scheduled at all tasks (Assumption 4). Therefore, we will use U_n instead of $U_{p,n}$ in this case. The state space is shown below.

$$\begin{aligned} S_1 &= \{\sigma_1 = (\sigma_{1,1}, \dots, \sigma_{J,1}) \mid \sigma_{p,1} = 0 \qquad \forall p \in P\} \\ S_n &= \{\sigma_n = (\sigma_{1,n}, \dots, \sigma_{J,n}) \mid 0 = \sigma_{J,n} \leq \sigma_{J-1,n} \leq \dots \leq \sigma_{1,n} \leq U_n \qquad \forall p \in P\} \qquad \forall 1 < n \leq N \end{aligned}$$

By using this definition all vectors shown below belong to the same state, while these vectors were different states in the state space described in Section 6.8.1.

$$\begin{gathered} \Big(\sigma_{1,n}, \sigma_{2,n}, \quad \dots \quad, \sigma_{J-1,n}, \sigma_{J,n}\Big) \\ \Big(\sigma_{2,n}, \sigma_{1,n}, \quad \dots \quad, \sigma_{J-1,n}, \sigma_{J,n}\Big) \\ \vdots \\ \Big(\sigma_{1,n}, \sigma_{2,n}, \quad \dots \quad, \sigma_{J,n}, \sigma_{J-1,n}\Big) \end{gathered}$$

When we use the states as defined above, also the transition probabilities change. This is because we now have more possibilities to reach a state. The transition probabilities are shown below. The transition probabilities are almost the same as described in Section 6.8.1. However, we now have more possibilities to reach a state, namely all permutations of the state. Therefore, we have an extra summation in the transition probabilities. In Section 6.8.1 the transition from state σ_n to state i was possible with $i_p + k - \sigma_{p,n}$ arrivals per person with the following conditions $k \in \mathbb{N}$, $i_p + k \leq U_{n+1}\ \forall p$ and $i_p + k \geq \sigma_{p,n}\ \forall p$. Now also the permutations of the vector $i + k - \sigma_n$ are possible arrivals per person for this transition if $E_p \geq \sigma_{p,n}\ \forall p$ where E is a permutation of the vector $i + k$.

$$\mathbb{P}\Big(i \mid \sigma_n, a\Big) = \sum_{\substack{k \\ k \in \mathbb{N}, \\ i_p + k \leq U_{n+1}\ \forall p, \\ i_p + k \geq \sigma_{p,n}\ \forall p}} \sum_{\substack{E, \\ E \text{ permutation of } i+k, \\ E_p \geq \sigma_{p,n}\ \forall p}} \prod_{p: E_p < U_{n+1,p}} e^{-\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d}} \frac{\left(\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d}\right)^{E_p - \sigma_{p,n}}}{(E_p - \sigma_{p,n})!} \cdot \prod_{p: E_P = U_{n+1,p}} \Big(1 - \sum_{m=\sigma_{p,n}}^{E_p - 1} e^{-\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d}} \frac{\left(\sum_d \sum_t x_{p,t,d} \cdot \lambda_{t,d}\right)^{m - \sigma_{p,n}}}{(m - \sigma_{p,n})!}\Big)$$

Now we want to know the number of states, when we use the states as defined above. Theorem 4 is used for this.

Theorem 4. *[Grimaldi, 2003]*
The number of combinations of k objects from n elements with repetition is equal to $C(n+k-1, k)$, where

$$C(n,k) = \begin{pmatrix} n \\ k \end{pmatrix} = \frac{n!}{k!(n-k)!}$$

The number of states is equal to the number of combinations of $P-1$ objects from U_n+1 elements. The objects are the persons. However, we know that the last person has element zero. Therefore, we only need $P-1$ objects. Every person has a state between zero and U_n and this are U_n+1 possibilities. We only include the states which are ordered lexicography. Therefore we do not include two different states when one can be obtained from the other by permutation of the terms. Because of this, the number of states in the state space is the number of combinations of $P-1$ objects from U_n+1 elements with repetition. From Theorem 4 follows that the number of states in the state space is equal to

$$C\Big((U_n+1) + (P-1) - 1, P-1\Big) = \begin{pmatrix} (U_n+1) + (P-1) - 1 \\ P-1 \end{pmatrix} = \frac{(P-1+U_n)!}{U_n!(P-1)!}$$

The number of states in this case for the examples are shown below.

- When we have for example five persons who could have completed four orders, each the number of states is equal to

$$\frac{(5-1+4)!}{4!(5-1)!} = \frac{8!}{4!4!} = 70$$

- When we have ten persons who could have completed fifty orders, each the number of states is equal to

$$\frac{(10-1+50)!}{50!(10-1)!} = \frac{59!}{50!9!} = 12\,565\,671\,261$$

We will use this definition of states and the results of the DP with these states are shown in the next chapter.

Chapter 7

Results SDP

The dynamic programming algorithm is programmed in Python and results for small test instances are discussed in this chapter. Section 7.1 shows the results for a small instance when the original state space is used, Section 7.2 shows the results for instances when the lexicographically ordered states are used. Section 7.3 describes the structure for the optimal actions of the different states at the different decision epochs.

7.1 Results original states

Results from the DP with three persons who can be scheduled over three tasks with one time block and when two schedules are made, are shown in Table E.1 in Appendix E.1. This table shows an optimal action given the state and decision epoch. A 1 corresponds to task one with λ_1, a 2 corresponds to task two with λ_2 and a 3 corresponds to task three with λ_3. When '$-$' is in the table, the corresponding state is not part of the state space at the corresponding decision epoch. The parameters used for this case are shown below.

$$\begin{aligned}
\lambda_1 &= 0.01 \\
\lambda_2 &= 2.02 \\
\lambda_3 &= 1.00 \\
q &= 0.01 \\
N &= 3 \\
J &= 3 \\
M &= 3 \\
B &= 1
\end{aligned}$$

From these results is concluded that the DP works as expected, because in order of increasing number of orders completed, persons are scheduled at tasks with decreasing arrival rate. Therefore, the person with the highest number of orders completed is scheduled at the task with the lowest arrival rate, the person with the second highest number of orders completed is scheduled at the task with the second lowest arrival rate and the person with the lowest number of orders completed is scheduled at the task with the highest arrival rate.

7.2 Results states ordered lexicographically

In this section the results are shown from the dynamic programming algorithm with the states that are lexicographically ordered. By looking at different small instances we hope that we find a structure in the optimal actions we can extend to bigger instances. First we will look at the case that the arrival rate of a task is the same during the different schedules. So, the arrival process of the orders is the same for every schedule. In Section 7.2.1 the number of time blocks during a schedule is equal to the number of different tasks. Then, in Section 7.2.2 we will look at the case that the arrival process of orders during a schedule is the same for each schedule, but the number of time blocks during a schedule is not equal to the number of different tasks. Section 7.2.3 shows the results when the arrival process differs per schedule. Section 7.2.4 shows the results when the arrival rates differ in the last schedule and Section 7.2.5 shows the results when the differences between the arrival rates of the different tasks are bigger.

7.2.1 Same arrival rate during schedules and number of time blocks equal to number of different tasks

The results from the DP with three persons who can be scheduled over three tasks with three time blocks and when five times a schedule is made, are shown in Table E.2 for the fifth schedule, Table E.3 for the fourth schedule, Table E.4 for the third schedule, Table E.5 for the second schedule and Table E.6 for the first schedule in Appendix E. These tables show an optimal action given the state and decision epoch. A 1 corresponds to task one with λ_1, a 2 corresponds to task two with λ_2 and a 3 corresponds to task three with λ_3. The parameters used in this case are shown below.

$$\begin{aligned}
\lambda_1 &= 0.05 \\
\lambda_2 &= 0.5 \\
\lambda_3 &= 0.3 \\
q &= 0.01 \\
N &= 6 \\
J &= 3 \\
M &= 3 \\
B &= 3
\end{aligned}$$

The last column of Tables E.2 – E.6 is the expected new state. This expected new state is equal to the state before the schedule made plus the arrival rates from the tasks that are scheduled for the persons. Sometimes the variance of the schedule also influences the optimal action. First, this will be explained. We will also explain the variance of the schedule, because we will refer to this variance later. Then, the optimal action will be described.

Expected new state

The arrival process of orders during a schedule for a person p is a Poisson Process. Therefore, the number of orders arriving is Poisson distributed with mean $\gamma_{p,n} \cdot t$ (Definition 6.2.2) . $\gamma_{p,n}$ is the sum over all arrival rates of orders during tasks that are scheduled for that person during schedule n. The arrival rates are per time block, because the tasks are scheduled per time block. Therefore, t is equal to one when we want to know the expected number of arrivals during the schedule. When we now want to know the expected new state, we can add for all persons the total arrival rate γ_p to the state before the made schedule. We do not take into account that the states are equal to the differences of the number of orders completed and ordered lexicographically. Therefore, it is possible no zero is in the state space and the expected new states are not ordered lexicographically. It is also possible that the expected new state is not a state in the state space because the states in the state space are integers and the expected new state is not necessarily an integer. The formula for the expected new state is shown below.

$$\mathbb{E}[(\sigma_{1,n+1}, \ldots \sigma_{J,n+1})] = (\sigma_{1,n} + \gamma_{1,n}, \ldots, \sigma_{J,n} + \gamma_{J,n})$$

Variance of the schedule

We can also look at the variance of the schedule. To do this, we need to know the variance of the number of orders arriving during a schedule per person. The number of orders arriving during a schedule for a person p is Poisson distributed with parameter $\gamma_{p,n}$. Therefore, the variance of the number of orders arriving during the schedule is equal to $\gamma_{p,n}$ [Ross, 2010]. The variance in this case is the mean difference between the number of orders that arrive during a Poisson process with parameter $\gamma_{p,n}$. The variance of the schedule is equal to

$$\mathrm{Var}(\text{schedule } n) = (\gamma_{1,n}, \ldots, \gamma_{J,n})$$

Optimal action

In Tables E.2 – E.6 in Appendix E.2 is shown that the optimal action is chosen in such a way that the expected new state is as close as possible to the line where all persons have completed the same number of orders. Therefore, it does not matter which schedule is made, the optimal action only depends on the current state of the system.

When we now look at the optimal actions at the different states of the system, we see that the same action is chosen for a group of states. The different groups of states are states when all persons have the same number of completed orders (only state $(0,0,0)$), two persons have the same number of orders completed but fewer than the other person, two persons have the same number of completed orders and more than the other person, and when the three persons all have different numbers of completed orders. This is shown in Table 7.1. For this table we need a, b as defined below.

$$a, b \in \mathbb{N}$$
$$a > b > 0$$

Table 7.1: The optimal action for the different groups of states.

State group	Optimal action								
	$p=1$			$p=2$			$p=3$		
	$d=1$	$d=2$	$d=3$	$d=1$	$d=2$	$d=3$	$d=1$	$d=2$	$d=3$
$(0,0,0)$	1	3	2	2	1	3	3	2	1
$(a,0,0)$	1	1	1	2	2	3	3	3	2
$(a,a,0)$	1	1	3	3	3	1	2	2	2
$(a,b,0)$	1	1	1	3	3	3	2	2	2

There are a couple of exceptions to this rule. The optimal actions chosen at state $(2,0,0)$ of the fifth schedule, states $(1,0,0)$, $(1,1,0)$, $(2,2,0)$ and $(6,0,0)$ of the fourth schedule, states $(1,0,0)$ and $(1,1,0)$ of the third schedule, and states $(1,0,0)$ and $(1,1,0)$ of the second schedule differ from Table 7.1.

For the states $(2,0,0)$, $(2,2,0)$ and $(6,0,0)$ the schedule for the two persons with the same number of completed orders is switched in schedules two, three and four in comparison with schedule five. Therefore, the differences in the expected new state are the same. Therefore, we would expect that the expected future costs are the same for both actions. However, we should take the same optimal action in this case, because we use the same order of the actions to evaluate the expected future costs and only update the optimal action if the costs of this action are lower than the expected future costs of the current optimal action. The expected future costs for the actions are shown in Table 7.2. In this table is shown that the absolute differences of the expected future costs of the different actions (|expected future costs with action following the rule $-$ expected future costs with optimal action|) are low. This can be explained by the rounding errors.

Table 7.2: The expected future costs for the optimal action and the action that follows the rule for states $(2,0,0)$ in the fifth schedule and $(2,2,0)$ and $(6,0,0)$ of the fourth schedule .

State	Action	Expected future costs	Absolute difference expected future costs
$(2,0,0)$	Optimal	2.3216666664275727	
$(2,0,0)$	Follow the rule	2.3216666664275730	$3 \cdot 10^{-16}$
$(2,2,0)$	Optimal	2.4549623347776874	
$(2,2,0)$	Follow the rule	2.4549623347776880	$6 \cdot 10^{-16}$
$(6,0,0)$	Optimal	12.990209643789226	
$(6,0,0)$	Follow the rule	12.990209643789228	$2 \cdot 10^{-15}$

Rounding errors appear because of the finite precision of the computer arithmetic (floating points) [Trefethen and III, 1997]. These rounding errors appear and spread whenever any arithmetic operation is performed. Even a simple input of a real number x into a computer leads to an error. Instead of x we get the floating point representation of x, $fl(x)$:

$$\begin{aligned} x &\rightarrow fl(x) \\ fl(x) &= x(1+\epsilon) \\ |\epsilon| &\leq \epsilon_c \end{aligned}$$

Table 7.3: The expected future costs for the optimal action and the action that follows the rule for states at the boundary.

Schedule five			
State	Action	Expected future costs	Absolute difference expected future costs
$(12,12,0)$	Optimal	82.1750603578536	
$(12,12,0)$	Follow the rule	82.33263282899317	0.1575724711395
Schedule four			
State	Action	Expected future costs	Absolute difference expected future costs
$(10,10,0)$	Optimal	45.77322055699952	
$(10,10,0)$	Follow the rule	45.93552501646413	0.16230445946461
Schedule two			
State	Action	Expected future costs	Absolute difference expected future costs
$(5,5,0)$	Optimal	5.314926724627012	
$(5,5,0)$	Follow the rule	5.358994449565818	0.044067724938806

where ϵ_c is the machine precision ($\epsilon_c \approx 2.22 \cdot 10^{-16}$ in our case). The floating point representation of the state of the system $(a, b, 0)$ is

$$\begin{array}{rcl} (a,b,0) & \rightarrow & (a+\epsilon_1, b+\epsilon_2, 0+\epsilon_3) \\ |\epsilon_1| & \leq & \epsilon_c \\ |\epsilon_2| & \leq & \epsilon_c \\ |\epsilon_3| & \leq & \epsilon_c \end{array}$$

This floating point representation does not only affect the input, but also every elementary arithmetic operation. Let $*$ be one of the operations $+, -, \cdot, :$ then

$$\begin{array}{rcl} x * y & \rightarrow & (x*y)(1+\epsilon) \\ |\epsilon| & \leq & \epsilon_c \end{array}$$

This can affect the costs at the horizon, the transition probabilities and the expected future costs for states not at the horizon. Therefore, the expected future costs of the actions shown in Table 7.2 can differ a little.

The optimal action also differs three times at the boundary from the actions in Table 7.1, in schedule five and state $(12, 12, 0)$, in schedule four and state $(10, 10, 0)$, and in schedule two and state $(5, 5, 0)$. In these cases the expected new state would be closer to the line where all persons have completed the same number of orders when person one is scheduled at one day at tasks three and the other days at tasks one, person two one day at task one and the other days at task three and person three is always scheduled at task two. The differences between the expected future costs of the optimal action and the action that follows the rule are shown in Table 7.3. We see that this difference is too big to be a result of the rounding error. It is possible that this is a result of using a boundary for the state space, because both states are the last state of the state space of the corresponding decision epoch. Therefore, these states are at the boundary. This can be explained by the fact that we cut down the state space at the upper boundary. The upper boundaries used for every decision epoch are shown in Table 7.4. In this table we see that at decision epoch five, four and two the difference between the upper boundary of this decision epoch and the upper boundary for the next decision epoch is equal to two. The differences between the upper boundary of the decision epochs three and one and the upper boundary of the next decision epoch are bigger. This is equal to three for the upper boundary between decision epochs three and four and equal to five for the difference between the upper boundaries of decision epoch two and one. When the number of completed orders is equal to the boundary, the transition probability is equal to the probability that this person has this number of orders completed or more. Therefore, the distance from the line where all persons have the same number of orders completed at the boundary is bigger in reality. Because of this, one person with the state at the boundary is scheduled three times at task three (where in all other cases this person is scheduled two times at task three and once at task one) and the other person with the state at the boundary is scheduled three times at task one (where in all other cases this person is schedules two times at task one and once at task three).

When we compute the future costs for states $(1, 0, 0)$ and $(1, 1, 0)$ we get Table 7.5. We see that the differences between the optimal action and the action that follows the rule are bigger than the differences in Table 7.2. These differences cannot be explained by the rounding errors. Therefore, we now

Table 7.4: The upper boundaries used per decision epoch.

Decision epoch	Upper boundary state space
6	14
5	12
4	10
3	7
2	5
1	0

will look at the expected new state and the variance of the schedule of the states $(1,0,0)$ and $(1,1,0)$. These expected new states and variances of the schedule are shown in Table 7.6. The schedule for state $(1,0,0)$ in the fourth schedule differs from schedule two and three, because persons two and three are switched. The distance between the expected new state in this case and the line where all persons completed the same number is the same when these persons are switched. The variance of the system only changes by switching the variance for these persons, but the order of the variance of the system does not matter when two persons also have the same state. However, the order of the persons for the expected new state and the variance of the system has to be the same.

Table 7.5: The expected future costs for the optimal action and the action that follows the rule for states $(1,0,0)$ and $(1,1,0)$ in the second, third, and fourth schedule.

Schedule four			
State	Action	Expected future costs	Absolute difference expected future costs
$(1,0,0)$	Optimal	2.139580314195619	
$(1,0,0)$	Follow the rule	2.144141881838927	0.004561567643308
$(1,1,0)$	Optimal	2.1514951004157923	
$(1,1,0)$	Follow the rule	2.1524231775891844	0.0009280771733921
Schedule three			
State	Action	Expected future costs	Absolute difference expected future costs
$(1,0,0)$	Optimal	2.3206349001807216	
$(1,0,0)$	Follow the rule	2.3273136640022862	0.0066787638215646
$(1,1,0)$	Optimal	2.326958995984065	
$(1,1,0)$	Follow the rule	2.329512163533935	0.002553167549870
Schedule two			
State	Action	Expected future costs	Absolute difference expected future costs
$(1,0,0)$	Optimal	2.413511008836633	
$(1,0,0)$	Follow the rule	2.4167499445594425	0.003238935722808
$(1,1,0)$	Optimal	2.4168189948615746	
$(1,1,0)$	Follow the rule	2.4175105350679833	0.0006915402064087

In Table 7.6 we see that when the expected new state is close to the line where the number of completed orders is equally divided over the persons, the optimal action in schedules two, three and four is to increase this distance a little to decrease the variance of the person with the highest variance. For example, when we start at the state $(1,0,0)$ before the fifth schedule we see that the expected new state after the fifth schedule is equal to $(1.15,1.3,1.1)$. However, when we start in state $(1,0,0)$ before the second third or fourth schedule, we see that the expected new state after the schedule is equal to $(1.4,1.1,1.05)$. Therefore, the maximum differences is bigger between the elements of the expected new state after the second, third or fourth schedule (0.3) than after the fifth schedule (0.2). However, the maximum difference between the elements of the variance of schedule two, three or four is equal to 0.7 while this is equal to 0.95 in schedule five.

In the last schedule of the year we want to have the smallest distance between the expected new state and the line where the number of completed orders is equally divided over the persons, also if this means that we have big differences between the elements of the variance of the schedule. In Section 7.2.5 we will have bigger differences between the arrival rates to have a better look at these optimal actions that are not the actions with the expected new state as close as possible to the line where all persons have completed the same number of orders.

The optimal actions are chosen such that the expected new state is as close as possible to the line

Table 7.6: The optimal action for states $(1,0,0)$ and $(1,1,0)$.

State	Schedule	Expected new state	Variance of the schedule
$(1,0,0)$	Schedules two, three and four	$(1.4,1.1,1.05)$	$(0.4,1.1,1.05)$
$(1,0,0)$	Schedule five	$(1.15,1.3,1.1)$	$(0.15,1.3,1.1)$
$(1,1,0)$	Schedules two, three and four	$(1.6,1.65,1.3)$	$(0.6,0.65,1.3)$
$(1,1,0)$	Schedule five	$(1.4,1.65,1.5)$	$(0.4,0.65,1.5)$

where the number of completed orders is equally divided over the persons. When the expected new state is very close to this line and the schedule made is not the last one, an action is chosen such that the expected new state is still close to this line, but the distance can be increased a little such that the elements of the variance of the schedule are closer to each other.

That the optimal actions do not create big differences between the expected new state of the different persons can be explained by the fact that in this case in the next schedules big differences between the variance of these persons arise to approach the line where the number of completed orders is equally divided over the persons at the horizon, because the variance is equal to the mean number of orders that arrive. When the differences between the variances of the persons are big, also the variance of a person is big. When a person has a bigger variance, the mean differences between the numbers of orders that can arrive during a Poisson process with this parameter is bigger. Therefore, the distance between the expected new state and the line where the number of completed orders is equally divided can again be big.

7.2.2 Same arrival rate during schedules and the number of time blocks not equal to the number of different tasks

Now we will look at an example when the arrival process of orders is the same during the different schedules, but the number of time blocks is not equal to the number of different tasks. This is done, to see if the same structure in the optimal actions can be seen in this case. The results from the DP with three persons who can be scheduled over three tasks with two time blocks and when five schedules are made, are shown in Tables E.7 (fifth schedule) – E.11 (first schedule) in Appendix E.2. These tables show an optimal action given the state and decision epoch. A 1 corresponds to task one with λ_1, a 2 corresponds to task two with λ_2 and a 3 corresponds to task three with λ_3. The parameters used in this case are shown below.

$$\begin{aligned}
\lambda_1 &= 0.05 \\
\lambda_2 &= 0.5 \\
\lambda_3 &= 0.3 \\
q &= 0.01 \\
N &= 6 \\
J &= 3 \\
M &= 3 \\
B &= 2
\end{aligned}$$

In Tables E.7 – E.11 in Appendix E.2 we see that the optimal action is chosen in the same way as is done in Section 7.2.1. Therefore, the optimal action is chosen in such a way that the expected new state is as close as possible to the line where all persons have completed the same number of orders. We also see that it does not matter in which decision epoch we are, the optimal action only depends on the current state of the system. The optimal actions for the different groups of states (same groups as Table 7.1) are shown in Table 7.7. For this table we need a, b as defined below.

$$\begin{gathered}
a, b \in \mathbb{N} \\
a > b > 0
\end{gathered}$$

There are a couple of exceptions to this rule. In schedule five and states $(8,8,0)$ and $(9,9,0)$, in schedule four and state $(7,7,0)$, in schedule three and states $(5,5,0)$, $(6,5,0)$ and $(6,6,0)$, in schedule two and state $(4,4,0)$, and in schedule one and state $(0,0,0)$.

In the optimal action in state $(0,0,0)$ of schedule one the schedules of persons two and three are

Table 7.7: The optimal action for the different groups of states.

State group	Optimal action					
	$p = 1$		$p = 2$		$p = 3$	
	$d = 1$	$d = 2$	$d = 1$	$d = 2$	$d = 1$	$d = 2$
$(0, 0, 0)$	1	2	2	1	3	3
$(a, 0, 0)$	1	1	2	3	3	2
$(a, a, 0)$	1	3	3	1	2	2
$(a, b, 0)$	1	1	3	3	2	2

switched. By doing this, the distance between the expected new state and the line where all persons have completed the same number of orders remains the same. The optimal action differs from the rule because of the rounding error (Section 7.2.1).

The differences of the expected future costs of the optimal action and the action that follows the rule are shown in Table 7.8. We see that this difference in all cases is too big to be explained by the rounding error. Now we will look at the upper boundary, like is done in Section 7.2.1. The upper boundaries used for every decision epoch are shown in Table 7.9. In this table we see that the difference between the upper boundary at a decision epoch and the upper boundary at the next decision epoch is less than or equal to two, except the difference between the upper boundary at decision epoch one and two. Therefore, the state at the boundary at decision epochs two, three, four and five has another optimal action. This is the same as in Section 7.2.1. In schedule five and three more states close to the boundary are an exception of the rule. This can be explained by the fact that the difference between the upper boundary of these decision epochs and the next decision epoch is equal to one. However, in schedule three state $(6, 5, 0)$ is an exception of the rule, while state $(9, 8, 0)$ follows the rule in schedule five. When we use the parameters shown below, we do not have optimal actions that differ from Table 7.7 and the optimal action that differs from the table cannot be explained by upper boundary or rounding error.

Table 7.8: The expected future costs for the optimal action and the action that follows the rule for states at the boundary.

Schedule five			
State	Action	Expected future costs	Absolute difference expected future costs
$(8, 8, 0)$	Optimal	37.05292776431982	
$(8, 8, 0)$	Follow the rule	37.053391864072324	0.00046409975250
$(9, 9, 0)$	Optimal	46.58281482290619	
$(9, 9, 0)$	Follow the rule	46.81893682158964	0.23612199868345
Schedule four			
State	Action	Expected future costs	Absolute difference expected future costs
$(7, 7, 0)$	Optimal	23.615159598349912	
$(7, 7, 0)$	Follow the rule	23.63164711757915	0.01648751922924
Schedule three			
State	Action	Expected future costs	Absolute difference expected future costs
$(5, 5, 0)$	Optimal	9.029214031778451	
$(5, 5, 0)$	Follow the rule	9.040196892198145	0.010982860419694
$(6, 5, 0)$	Optimal	11.107878939861648	
$(6, 5, 0)$	Follow the rule	11.185160625814213	0.077281685952565
$(6, 6, 0)$	Optimal	13.304363390836167	
$(6, 6, 0)$	Follow the rule	13.433198470125204	0.128835079289037
Schedule two			
State	Action	Expected future costs	Absolute difference expected future costs
$(4, 4, 0)$	Optimal	4.63063946253339	
$(4, 4, 0)$	Follow the rule	4.6368545698440355	0.00621510731064

7.2.3 Arrival rates differ per schedule

In the previous example the arrival rates were the same during all schedules. Now we will look at an example where the arrival rates differ per schedule. In this example three persons are scheduled over three tasks with two time blocks and four schedules are made. The arrival rate of task two during

Table 7.9: The upper boundaries used per decision epoch.

Decision epoch	Upper boundary state space
6	10
5	9
4	7
3	6
2	4
1	0

schedule three differs from the arrival rate of task two during the other schedules. The parameters used are shown below.

$$\begin{aligned}
\lambda_1 &= 0.05 \\
\lambda_{2,d} &= \begin{cases} 0.5 & \text{if day } d \text{ is during schedule 1, 2 or 4} \\ 5.0 & \text{if day } d \text{ is during schedule 3} \end{cases} \\
\lambda_3 &= 0.3 \\
q &= 0.01 \\
N &= 5 \\
J &= 3 \\
M &= 3 \\
B &= 2
\end{aligned}$$

In Tables E.12 and E.13 in Appendix E.2 is shown that the optimal action is chosen in such a way that the expected new state is as close as possible to the line where all persons have completed the same number of orders. The actions shown in Table E.12 in Appendix E (fourth schedule) follow the rule shown in Table 7.7. The optimal actions in schedule three (Table E.13 in Appendix E.2) are shown in Table 7.10. For some states the schedule for the different persons is switched when these persons have the same state before the schedule. Therefore, the distance between the expected new state and the line where all persons have completed the same number of orders remains the same.

$$a, b \in \mathbb{N}$$
$$a \geq b > 0$$

Table 7.10: The optimal action for the third schedule.

State group	Optimal action					
	$p=1$		$p=2$		$p=3$	
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$
$(a,b,0)$, $a<5$	3	3	1	2	2	1
$(a,0,0)$, $a\geq 5$	1	1	2	3	3	2
$(a,a,0)$, $a\geq 5$	1	3	3	1	2	2
$(a,b,0)$, $a\geq 5>b$	1	3	2	1	3	2
$(a,b,0)$, $a\geq 5=b$	1	1	3	3	2	2

However, in Tables E.14 and E.15 in Appendix E.2 (second and first schedule) the optimal action chosen is different. This can be explained by the fact that in schedule three a task must be scheduled with a high arrival rate. Therefore, during schedules one and two the optimal action is such that the expected new state for one person is high. The task with high arrival rate is scheduled two days (in schedule three) and therefore, the person with the highest expected new state is not higher than the arrival rate of the task with the highest arrival rate. We do not know what will happen with states higher than the arrival rate of the task with the highest arrival rate. Also, the number of states is not so big. Therefore, we will look at another example with more decision epochs before the schedule with the big task.

7.2.4 Arrival rates differ in last schedule

We now look at the same case as in Section 7.2.3, but in this case the arrival rate of task two is higher in the last schedule and five schedules are made. The results from the DP are shown in Table E.16

in Appendix E.2 for the fourth schedule, Table E.17 in Appendix E.2 for the third schedule, Table E.18 in Appendix E.2 for the second schedule and Table E.19 in Appendix E.2 for the first schedule. These tables show an optimal action given the state and decision epoch. No results for schedule five are shown, because we only want to know what happens before schedule five. The parameters used are shown below.

$$
\begin{aligned}
\lambda_1 &= 0.05 \\
\lambda_{2,d} &= \begin{cases} 0.5 & \text{if day } d \text{ is during schedule 1, 2, 3 or 4} \\ 5.0 & \text{if day } d \text{ is during schedule 5} \end{cases} \\
\lambda_3 &= 0.3 \\
q &= 0.01 \\
N &= 6 \\
J &= 3 \\
M &= 3 \\
B &= 2
\end{aligned}
$$

The optimal actions shown in Tables E.16 – E.19 in Appendix E.2 follow the rules shown in Table 7.11. Where a, b are defined as below. For the case $a = 5$ the optimal action depends on the decision epoch. Therefore, these results are not shown in Table 7.11.

$$
\begin{aligned}
a, b \in \mathbb{N} \\
a \geq b > 0
\end{aligned}
$$

Table 7.11: The optimal action for the schedule before the schedule with the task with a high arrival rate.

State group	Optimal action					
	$p=1$		$p=2$		$p=3$	
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$
$(0,0,0)$	3	3	2	2	1	1
$(a,0,0)$, $a=1$	2	2	1	1	3	3
$(a,0,0)$, $1<a<5$	2	2	1	3	3	1
$(a,0,0)$, $a>5$	1	1	2	3	3	2
$(a,a,0)$, $a=1$	2	2	3	3	1	1
$(a,a,0)$, $1<a<5$	1	1	2	2	3	3
$(a,b,0)$, $5>a>b$	2	2	1	1	3	3
$(a,b,0)$, $a>5$, $a-b=1$	3	3	2	2	1	1
$(a,b,0)$, $a>5$, $1<a-b<5$	3	3	1	1	2	2
$(a,b,0)$, $a>5$, $a-b=5$	1	3	3	1	2	2
$(a,b,0)$, $a>5$, $a-b>5$	1	1	3	3	2	2

From these results we can conclude that the structure described in Section 7.2.1 does not hold when one schedule has a task with a higher arrival rate of orders than the other schedules. This can be explained by the fact that the big task must be scheduled and after that schedule the number of orders must be divided over the persons as equally as possible. Therefore, differences are created between the expected new state of persons. We also see in Table 7.11 that the rule for the optimal action for states with $a < 5$ and $a > 5$ differs. This can be explained by the fact that the big task has an arrival rate of five. When $a < 5$ the tasks scheduled for the first person have a bigger arrival rate than when $a > 5$. This is because we want to have differences to schedule the task with arrival rate five. When $a < 5$ the optimal action is such that the expected state of one person is approximately five more than the other two persons. This is also done when $a > 5$, but when two persons have more than five orders completed or exactly completed five orders before the schedule is made the other person is scheduled at the task with the lowest arrival rate. In this case, the person with zero completed orders can be scheduled two days at the task with the arrival rate of five. When the person with the highest number of completed orders is equal to five it depends on the decision epoch what is the optimal action. The differences between the expected new state of the persons are bigger in the third schedule than in the fourth schedule. The schedule for the second person is the same in both cases, but the schedules of the first and third person differ.

7.2.5 Bigger differences arrival rates

Now we will look at an example when the arrival rates of orders of the different tasks differ more. The arrival rates are the same during the different schedules. The results from the DP with three persons who can be scheduled over three tasks with two time blocks and when three schedules are made, are shown in Tables E.20 (third schedule) – E.22 (first schedule) in Appendix E.2. These tables show an optimal action given the state and decision epoch. A 1 corresponds to task one with λ_1, a 2 corresponds to task two with λ_2 and a 3 corresponds to task three with λ_3.

$$\begin{aligned} \lambda_1 &= 0.1 \\ \lambda_2 &= 3 \\ \lambda_3 &= 1 \\ q &= 0.01 \\ N &= 4 \\ P &= 3 \\ B &= 2 \end{aligned}$$

In Tables E.20 – E.22 in Appendix E.2 we see that the optimal action is chosen in the same way as is done in Section 7.2.1. Therefore, the optimal action is chosen in such a way that the expected new state is as close as possible to the line where all persons have completed the same number of orders. We also see that most of the times it does not matter in which decision epoch we are, the optimal action only depends on the current state of the system. The optimal action can differ during the different decision epochs for the same schedule. This is as described in Section 7.2.1 because the differences between the variances for the different persons are less while the differences in the expected new schedule increase a little when the expected new state is close to the line where all persons have completed the same number of orders during schedules which are not the last one. The optimal actions for the different groups of states are shown in Table 7.12. For this table we need a, b as defined below.

$$\begin{aligned} a, b &\in \mathbb{N} \\ a > b &> 1 \end{aligned}$$

Table 7.12: The optimal action for the different groups of states.

State group	Optimal action					
	$p=1$		$p=2$		$p=3$	
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$
$(0,0,0)$	1	2	2	1	3	3
$(a,0,0)$, $a=1$	3	3	1	2	2	1
$(a,a,0)$, $a=1$	1	2	3	3	2	1
$(a,1,0)$, $1<a<4$ or $(a,2,0)$, $2<a<5$	1	3	2	1	3	2
$(a,a,0)$, $a=2$	1	2	3	1	2	3
$(a,0,0)$ or $(a,1,0)$, $a>2$	1	1	2	3	3	2
$(a,b,0)$, $a>3$, $b>1$ and not $(4,2,0)$	1	1	3	3	2	2
$(a,a,0)$, $a>2$	1	3	3	1	2	2

All optimal actions of states in the third schedule follow the rule shown in Table 7.12. The states with an optimal action in the second schedule that differ from Table 7.12 are shown in Table 7.13. In this table the expected new state and the variance of the schedule are shown for both the different schedules made in schedule two and schedule three for the states with a different optimal action in schedule two and schedule three. States $(2,0,0)$, $(2,1,0)$, $(2,2,0)$, $(a,2,0)$, $4<a<8$ and $(12,2,0)$ have the same elements in the expected new state after schedule two and three. However, the expected new state is ordered differently and the variance of the schedule differs. The expected new state of states $(3,0,0)$, $(3,3,0)$, $(4,1,0)$, $(4,3,0)$, $(5,2,0)$, $(5,3,0)$ and $(a,a-1,0)$, $5<a$ after schedules two and three differs a little. The differences between the expected costs in schedule two of the optimal action chosen in schedule two and the optimal action chosen in schedule three are shown in Table C.3. The same thing is done for the expected costs in schedule three. This is shown in Table 7.15.

Table 7.13: The optimal action for the states with different optimal actions (and the different action is not switching schedules of persons with the same state) in schedule three and two.

State	Schedule	Expected new state	Variance of the schedule
$(2, 0, 0)$	Schedule two	$(4.0, 3.1, 3.1)$	$(2.0, 3.1, 3.1)$
$(2, 0, 0)$	Schedule three	$(3.1, 3.1, 4.0)$	$(1.1, 3.1, 4.0)$
$(2, 1, 0)$	Schedule two	$(4.0, 4.1, 3.1)$	$(2.0, 3.1, 3.1)$
$(2, 1, 0)$	Schedule three	$(3.1, 4.1, 4.0)$	$(1.1, 3.1, 4.0)$
$(2, 2, 0)$	Schedule two	$(5.1, 4.0, 3.1)$	$(3.1, 2.0, 3.1)$
$(2, 2, 0)$	Schedule three	$(5.1, 3.1, 4.0)$	$(3.1, 1.1, 4.0)$
$(3, 0, 0)$	Schedule two	$(4.1, 4.0, 3.1)$	$(1.1, 4.0, 3.1)$
$(3, 0, 0)$	Schedule three	$(3.2, 4.0, 4.0)$	$(0.2, 4.0, 4.0)$
$(3, 3, 0)$	Schedule two	$(4.1, 6.1, 4.0)$	$(1.1, 3.1, 4.0)$
$(3, 3, 0)$	Schedule three	$(4.1, 4.1, 6.0)$	$(1.1, 1.1, 6.0)$
$(4, 1, 0)$	Schedule two	$(5.1, 4.1, 4.0)$	$(1.1, 3.1, 4.0)$
$(4, 1, 0)$	Schedule three	$(4.2, 3.0, 4.0)$	$(0.2, 2.0, 4.0)$
$(4, 3, 0)$	Schedule two	$(5.1, 4.1, 6.0)$	$(1.1, 1.1, 6.0)$
$(4, 3, 0)$	Schedule three	$(4.2, 5.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(5, 2, 0)$	Schedule two	$(5.1, 5.1, 4.0)$	$(1.1, 3.1, 4.0)$
$(5, 2, 0)$	Schedule three	$(5.2, 4.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(5, 3, 0)$	Schedule two	$(6.1, 4.1, 4.0)$	$(1.1, 3.1, 4.0)$
$(5, 3, 0)$	Schedule three	$(5.2, 5.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(a, 2, 0)$,$5 < a < 8$	Schedule two	$(a + 0.2, 6.0, 4.0)$	$(0.2, 4.0, 4.0)$
$(a, 2, 0)$,$5 < a < 8$	Schedule three	$(a + 0.2, 4.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(a, a - 1, 0)$,$5 < a$	Schedule two	$(a + 1.1, a + 0.1, 6.0)$	$(1.1, 1.1, 6.0)$
$(a, a - 1, 0)$,$5 < a$	Schedule three	$(a + 0.2, a + 1.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(12, 2, 0)$	Schedule two	$(12.2, 6.0, 4.0)$	$(0.2, 4.0, 4.0)$
$(12, 2, 0)$	Schedule three	$(12.2, 4.0, 6.0)$	$(0.2, 2.0, 6.0)$

Table 7.14: The expected future costs second schedule.

State	Action	Expected future costs	Absolute differences
$(2,0,0)$	Optimal schedule 3	6.401861204840630	
$(2,0,0)$	Optimal schedule 2	6.338521280022710	0.063339924817920
$(2,1,0)$	Optimal schedule 3	6.382421228907967	
$(2,1,0)$	Optimal schedule 2	6.351037415965354	0.031383812942613
$(2,2,0)$	Optimal schedule 3	6.517568406626449	
$(2,2,0)$	Optimal schedule 2	6.493849069208849	0.023719337417599
$(3,0,0)$	Optimal schedule 3	6.404652358345826	
$(3,0,0)$	Optimal schedule 2	6.378155305834787	0.023719337417599
$(3,3,0)$	Optimal schedule 3	6.812822159632798	
$(3,3,0)$	Optimal schedule 2	6.621041359602996	0.191780800029802
$(4,1,0)$	Optimal schedule 3	6.436456043596647	
$(4,1,0)$	Optimal schedule 2	6.347082766378246	0.089373277218401
$(4,3,0)$	Optimal schedule 3	6.622460549056092	
$(4,3,0)$	Optimal schedule 2	6.497913227984748	0.124547321071344
$(5,2,0)$	Optimal schedule 3	6.776203830641944	
$(5,2,0)$	Optimal schedule 2	6.396728907175095	0.37947492346685
$(5,3,0)$	Optimal schedule 3	6.528779225264985	
$(5,3,0)$	Optimal schedule 2	6.464197823345366	0.06458140191962
$(6,2,0)$	Optimal schedule 3	6.638101141064784	
$(6,2,0)$	Optimal schedule 2	6.472776169442353	0.165324971622431
$(6,5,0)$	Optimal schedule 3	6.358708842837822	
$(6,5,0)$	Optimal schedule 2	6.356973995712993	0.001734847124829
$(7,2,0)$	Optimal schedule 3	6.5334192421137365	
$(7,2,0)$	Optimal schedule 2	6.4697798456039120	0.063639396509824
$(7,6,0)$	Optimal schedule 3	6.413944644006913	
$(7,6,0)$	Optimal schedule 2	6.381512459277475	0.032432184729439
$(8,7,0)$	Optimal schedule 3	6.586558239808126	
$(8,7,0)$	Optimal schedule 2	6.535751904128724	0.050806335679402
$(9,8,0)$	Optimal schedule 3	6.978666345660920	
$(9,8,0)$	Optimal schedule 2	6.923543350033924	0.055122995626996
$(10,9,0)$	Optimal schedule 3	7.772194404132056	
$(10,9,0)$	Optimal schedule 2	7.723199883112608	0.048994521019448
$(11,10,0)$	Optimal schedule 3	9.190336690098840	
$(11,10,0)$	Optimal schedule 2	9.152512397903530	0.037824292195310
$(12,2,0)$	Optimal schedule 3	15.805938549208275	
$(12,2,0)$	Optimal schedule 2	15.777059411013568	0.028879138194707
$(12,11,0)$	Optimal schedule 3	11.450763983079582	
$(12,11,0)$	Optimal schedule 2	11.425674542035331	0.025089441044251

Table 7.15: The expected future costs third schedule.

State	Action	Expected future costs	Absolute differences
$(2,0,0)$ $(2,0,0)$	Optimal schedule 3 Optimal schedule 2	6.0066666666663595 6.0066666666666070	 $2.48 \cdot 10^{-13}$
$(2,1,0)$ $(2,1,0)$	Optimal schedule 3 Optimal schedule 2	6.073333333332986 6.073333333333229	 $2.43 \cdot 10^{-13}$
$(2,2,0)$ $(2,2,0)$	Optimal schedule 3 Optimal schedule 2	7.473333333332966 7.473333333333193	 $2.27 \cdot 10^{-13}$
$(3,0,0)$ $(3,0,0)$	Optimal schedule 3 Optimal schedule 2	5.893333333332792 6.073333333333030	 0.180000000000239
$(3,3,0)$ $(3,3,0)$	Optimal schedule 3 Optimal schedule 2	7.8733333308462505 8.2733333333328000	 0.400000002486550
$(4,1,0)$ $(4,1,0)$	Optimal schedule 3 Optimal schedule 2	6.0266666666650280 6.2066666666663455	 0.180000000001318
$(4,3,0)$ $(4,3,0)$	Optimal schedule 3 Optimal schedule 2	7.093333330898823 7.673333333332790	 0.580000002433967
$(5,2,0)$ $(5,2,0)$	Optimal schedule 3 Optimal schedule 2	7.4933333308988255 7.6733333333329930	 0.180000002434167
$(5,3,0)$ $(5,3,0)$	Optimal schedule 3 Optimal schedule 2	6.026666664284733 8.406666666666156	 2.380000002381424
$(6,2,0)$ $(6,2,0)$	Optimal schedule 3 Optimal schedule 2	8.426666664284731 8.426666666657903	 $2.373172 \cdot 10^{-9}$
$(6,5,0)$ $(6,5,0)$	Optimal schedule 3 Optimal schedule 2	6.026666664442417 6.206666664442458	 0.180000000000041
$(7,2,0)$ $(7,2,0)$	Optimal schedule 3 Optimal schedule 2	10.693333331003982 10.693333333324757	 $2.320775 \cdot 10^{-9}$
$(7,6,0)$ $(7,6,0)$	Optimal schedule 3 Optimal schedule 2	7.493333331214229 7.673333331214280	 0.180000000000051
$(8,7,0)$ $(8,7,0)$	Optimal schedule 3 Optimal schedule 2	10.293333331319220 10.473333331319438	 0.180000000000218
$(9,8,0)$ $(9,8,0)$	Optimal schedule 3 Optimal schedule 2	14.426666664755983 14.606666664757897	 0.180000000001915
$(10,9,0)$ $(10,9,0)$	Optimal schedule 3 Optimal schedule 2	19.893333331510990 20.073333331529664	 0.180000000018673
$(11,10,0)$ $(11,10,0)$	Optimal schedule 3 Optimal schedule 2	26.693333331459730 26.873333331634715	 0.180000000174985
$(12,2,0)$ $(12,2,0)$	Optimal schedule 3 Optimal schedule 2	42.02666666460017 42.02666666665909	 $2.058918 \cdot 10^{-9}$
$(12,11,0)$ $(12,11,0)$	Optimal schedule 3 Optimal schedule 2	34.826666663518246 35.006666665071180	 0.180000001552933

7.3 Structure optimal schedules

As described in Section 7.2 most of the optimal actions are the action that divides the number of orders over the persons as equally as possible. However, we also have seen some exceptions. Most of these exceptions could be explained by the fact that we use an upper boundary for the state space. The number of completed orders equal to the upper boundary can be reached when the number of completed orders at the end of the schedule is equal to this upper bound, but also when the number of completed orders at the end of the schedule is bigger than this upper bound.
When during a schedule a task is scheduled with a big arrival rate with respect to the other arrival rates, the schedules before this the optimal actions are not equal to the action that divides the number of orders over the persons as equally as possible. This is because the number of completed orders after the schedule with the task with a big arrival rate can be divided more equally over the persons when the differences between the number of completed orders are bigger when the number of days this task should be scheduled is not equal to the number of persons that are scheduled. When the arrival rates of the different tasks are fixed over the schedules, but the differences between the arrival rates are bigger we also have seen some exceptions. However, we did not manage to explain these results. We expect that some exceptions can be explained by the same reason as for the schedule with the task with a big arrival rate or the variance of the schedule.

Chapter 8

Model including the number of surgeries done per surgeon

Now that we have a generic mathematical model, we want to apply this model for scheduling the transplant surgeons in the LUMC to distribute the surgeries as equally as possible over the surgeons by the end of the year. To do this, every quarter a schedule must be made of the different tasks over the surgeons. The notation used is shown in Section 8.1, the assumptions made are shown in Section 8.2, the derivation of the transition probabilities is shown in Section 8.3, solving the model is discussed in Section 8.4, in Section 8.5 an ILP approximation is given, Section 8.6 shows the input for the ILP model, Section 8.7 shows the results of the ILP model with this input and Section 8.8 compares the results with the results described in Section 5.3.

8.1 Notation

In this section we will adjust parts of the model shown in Section 6.2 to the situation of the transplant surgeons in the LUMC. This is shown below.

- **Persons**
  The different tasks are scheduled over the surgeons. Not all surgeons are used for equally dividing the surgeries. This is because some surgeons can for example only be scheduled at the organ removal shift and these shifts are scheduled before the rest of the schedule is made. To indicate the number of surgeons that are used for the objective function, the parameter shown below is used. This is the number of surgeons the surgeries must be equally divided over.

$$NS \quad = \quad \text{number of surgeons to equally divide surgeries over}$$

- **Tasks**
  Different tasks must be scheduled over the surgeons. The different tasks are described in Section 3.1. In total eight different tasks must be scheduled. This is because the $M1$ and $M2$ are the same tasks and therefore modelled as one M shift where two surgeons should be scheduled. This is the same as done in Section 4.2.

- **Blocks**
  A task is scheduled for one day.

- **Decision epochs**
  Each quarter a schedule must be made, so at these moments an action should be taken. The last decision epoch is at 5, because four times a schedule must be made and the last decision epoch is at the end of the last schedule.

- **States**
  The states are defined in the same way as in Section 6.2. However, we do not take into account all surgeons. We will not take into account the surgeons who can only be scheduled at the organ removal shift or the organ removal shift, outpatient clinic and operating room. This is because we cannot schedule the organ removal shift for these surgeons because this is done before the rest of the schedule. One surgeon in the LUMC can also be scheduled at the operating room and outpatient clinic, but these tasks are scheduled based on the availability of this surgeon. This

surgeon is available only one day a week, except for the weeks this surgeon is scheduled at the organ removal shift. Therefore, we do not take into account the number of surgeries done by this surgeon. To make sure that the constraints at the boundary hold, we must include some information of the end of the last schedule in the states.

$$\begin{aligned}\sigma_n &= (\sigma_{1,n},\ldots,\sigma_{NS,n},WB_{1,n},\ldots,WB_{J,n},WBT_{1,n},\ldots,WBT_{J,n},TWB_{1,n},\ldots,TWB_{J,n},\\ &\quad TW_{1,n},\ldots,TW_{J,n},WBM_{1,n},\ldots,WBM_{J,n},ND_{1,n},\ldots,ND_{J,n},NDT_{1,n},\ldots,NDT_{J,n})\end{aligned}$$

where the new parts of the state are as defined in Section 4.2.3. However, in Section 4.2.3 we did not have the subscript n and now we have. This is because we make more schedules in this model and in Section 4.2.3 we make every schedule separate.

- **Decisions**
The decisions in the LUMC case are the quarter schedules. These schedules must fulfill all constraints described in Section 4.3 and Section 4.4. For the constraints at the boundary in decision epoch n we will use the boundary parameters of decision epoch $n-1$. For example, if we want to schedule a task at the first day of decision epoch n, we need to know if the last weekend of decision epoch $n-1$ the $T1$ shift is scheduled for this surgeon. When this is the case, we cannot schedule a task on the first day of the schedule of decision epoch n.

$$\begin{aligned}A &= \{x_{p,t,d} \quad \forall p,t,d \mid \text{constraints in Section 4.3 and Section 4.4 hold}\}\\ x_{p,t,d} &= \begin{cases}1 & \text{if surgeon } p \text{ is scheduled at task } t \text{ at day number } d\\ 0 & \text{otherwise}\end{cases}\\ a &\in A\end{aligned}$$

- **Direct costs**
We want to distribute the number of surgeries as equally as possible over the surgeons by the end of the year. Therefore the costs shown below are used. These costs are the same as described in Section 6.2. However, we do not take into account all surgeons. Also, some surgeons do not work full time. Therefore, we do not want to distribute the number of surgeries as equally as possible, because we want to take into account the work week of every surgeon. Therefore, a surgeon who does not work full time has ideally done fewer surgeries than a surgeon who works full time.

$$c_n(\sigma_n) = \begin{cases}\text{objective function as defined in Section 4.5} + K\cdot\sum_{k:\sum_t TS_{k,t}\neq TS_{k,M}+TS_{k,OC}+TS_{k,OR}} & \\ \left(\sigma_{k,n} - \frac{\sum_j \sigma_{j,n}}{\sum_{l:\sum_t TS_{l,t}\neq TS_{l,M}+TS_{l,OC}+TS_{l,OR}} AV_l}\cdot AV_k\right)^2 & \text{if } n=5\\ \text{objective function as defined in Section 4.5} & \text{otherwise}\end{cases}$$

where

$$\begin{aligned}K &= \text{constant to determine the importance of the costs of the distribution of the completed}\\ &\quad\ \text{surgeries over the surgeons}\\ AV_p &= \frac{\text{number of hours per week working}}{\text{number of hours per full time week}}\end{aligned}$$

- **Transition probabilities**
The transition probabilities differ, because no surgeries can be done during Hpb and Ch tasks and the distribution of surgeries during the OR task is not a Poisson Process. This is because the surgeries are scheduled during the OR task. Assumptions 6, 7 and 8 in Section 8.2 give the distribution of the number of surgeries during a task. The transition probabilities are shown below and the derivation of the transition probabilities is shown in Section 8.3.

$$\mathbb{P}\Big(i \mid \sigma_n, a\Big) = \begin{cases}\prod_{p:i_p<U_{n,p}} e^{-\sum_d\sum_{t\notin\{Hpb,Ch,OR\}} x_{p,t,d}\cdot\lambda_{t,d}} \frac{(\sum_d\sum_{t\notin\{Hpb,Ch,OR\}} x_{p,t,d}\cdot\lambda_{t,d})^{i_p-\sigma_{p,n}-\sum_d x_{p,OR,t}\cdot\lambda_{OR,d}}}{(i_p-\sigma_{p,n}-\sum_d x_{p,OR,t}\cdot\lambda_{OR,d})!}\cdot & \\ \prod_{p:i_p=U_{n,p}}\Big(1-\sum_{k<i_p} & \\ e^{-\sum_d\sum_{t\notin\{Hpb,Ch,OC,OR\}} x_{p,t,d}\cdot\lambda_{t,d}} \frac{(\sum_d\sum_{t\notin\{Hpb,Ch,OC,OR\}} x_{p,t,d}\cdot\lambda_{t,d})^{k-\sigma_{p,n}-\sum_d x_{p,OR,t}\cdot\lambda_{OR,d}}}{(k-\sigma_{p,n}-\sum_d x_{p,OR,t}\cdot\lambda_{OR,d})!}\Big) & \\ \quad \text{if } i_p-\sum_d x_{p,OR,t}\cdot\lambda_{OR,d}\geq\sigma_{n,p}\quad\forall p & \\ \quad \text{and the other parts of the state as defined below} & \\ 0 \quad \text{otherwise} & \end{cases}$$

First, below is shown how i looks like.

$$i = (\sigma_{1,n+1},\dots,\sigma_{NS,n+1}, WB_{1,n+1},\dots,WB_{J,n+1}, WBT_{1,n+1},\dots,WBT_{J,n+1}, TWB_{1,n+1},\dots,TWB_{J,n+1}, TW_{1,n+1},\dots,TW_{J,n+1}, WBM_{1,n+1},\dots,WBM_{J,n+1}, ND_{1,n+1},\dots,ND_{J,n+1}, NDT_{1,n+1},\dots,NDT_{J,n+1})$$

Now we can define the other parts of the state when we know $x_{s,t,d}\ \forall s,t,d$. This is shown below.

$$WB_{p,n+1} = \begin{cases} 1 & \text{if } x_{s,M,B-2} + \sum_{t\in\{T1,T2\}} x_{s,t,B-1} > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$WBT_{p,n+1} = \begin{cases} 1 & \text{if } x_{s,M,B-9} + \sum_{t\in\{T1,T2\}} x_{s,t,B-8} > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$TWB_{p,n+1} = \begin{cases} 1 & \text{if } x_{s,T1,B-6} = 1 \text{ and } COMB_W = 0 \\ 0 & \text{otherwise} \end{cases}$$

$$TW_{p,n+1} = \begin{cases} 1 & \text{if } x_{s,T1,B-1} = 1 \\ 0 & \text{otherwise} \end{cases}$$

$$WBM_{p,n+1} = \begin{cases} 1 & \text{if } x_{s,M,B-1} = 1 \\ 0 & \text{otherwise} \end{cases}$$

$$ND_{p,n+1} = \begin{cases} 0 & \text{if } x_{s,T2,B-1} = 0 \\ 2 & \text{if } x_{s,T2,B-1} = 1 \text{ and } x_{s,T2,B-2} = 0 \\ 3 & \text{if } x_{s,T2,B-1} = 1 \text{ and } x_{s,T2,B-2} = 1 \text{ and } x_{s,T2,B-3} = 0 \\ 4 & \text{if } x_{s,T2,B-1} = 1 \text{ and } x_{s,T2,B-2} = 1 \text{ and } x_{s,T2,B-3} = 1 \end{cases}$$

$$NDT_{p,n+1} = \begin{cases} 0 & \text{if } x_{s,T1,B-1} = 0 \\ 2 & \text{if } x_{s,T1,B-1} = 1 \text{ and } x_{s,T1,B-2} = 0 \\ 3 & \text{if } x_{s,T1,B-1} = 1 \text{ and } x_{s,T1,B-2} = 1 \end{cases}$$

8.2 Assumptions

Some assumptions must be made to use an SDP and solve the SDP. Assumptions 6 and 7 give the distribution of the arriving orders. The orders arrive independent of each other during shifts, therefore a Poisson Process is used for the arrival process during shifts. We assume that a fixed number of surgeries is done during the operating room task (deterministic distribution). When multiple tasks are scheduled simultaneously for a surgeon, we assume that the surgeon is always available to do surgeries during these tasks. This is done with Assumption 8.

Assumption 6. *Surgeries arrive during shift t on time block d according to a Poisson Process with rate $\lambda_{t,d}, \forall t,d$ per day. The arrival process is independent of the arrival processes at other tasks and other days. This process is also independent of the surgeon scheduled at the task.*
Assumption 7. *The arrival process of surgeries during the operating room is deterministic distributed with rate $\lambda_{OR,d}$.*
Assumption 8. *When a task is scheduled, the surgeon is always available for surgeries during this task, also when more tasks are scheduled for this surgeon.*

Assumption 9 is used to solve the SDP. To be able to solve the SDP, the state space must be finite. Therefore, Assumption 9 is needed.

Assumption 9. *The number of orders that can be completed by a person during a decision epoch has an upper boundary. Therefore, at every decision epoch a finite number of states can be reached.*

When the new schedule is made, we need to know how many surgeries the surgeons have completed after the previous schedule. For this, Assumption 10 is used.

Assumption 10. *The new schedule can be made after the end of the previous schedule.*

8.3 Derivation transition probabilities

From Section 6.4 we know the transition probabilities when surgeries arrive according to a Poisson Process during all tasks. These transition probabilities are shown below. The states contain more information in the LUMC context, but we will only look at the number of surgeries completed in this case, because all other parts of the states have transition probabilities equal to zero or one (depends on the schedule made). Therefore, the new state except the number of surgeries is fixed when the optimal action (schedule) is determined.

$$\mathbb{P}\Big(i \mid \sigma_n, a\Big) = \begin{cases} \prod_{p:i_p<U_{n,p}} e^{-\sum_d \sum_t x_{p,t,d}\cdot\lambda_{t,d}} \frac{(\sum_d \sum_t x_{p,t,d}\cdot\lambda_{t,d})^{i_p-\sigma_{p,n}}}{(i_p-\sigma_{p,n})!} \cdot \\ \prod_{p:i_p=U_{n,p}} \left(1-\sum_{k<i_p} e^{-\sum_d \sum_t x_{p,t,d}\cdot\lambda_{t,d}} \frac{(\sum_d \sum_t x_{p,t,d}\cdot\lambda_{t,d})^{k-\sigma_{p,n}}}{(k-\sigma_{p,n})!}\right) & \text{if } i_p \geq \sigma_{n,p} \quad \forall p \\ 0 & \text{otherwise} \end{cases}$$

In the LUMC case, no surgeries can be done during an Hpb, Ch or OC task. Therefore, we do not take these tasks into account in the transition probabilities. This is shown below.

$$\mathbb{P}\Big(i \mid \sigma_n, a\Big) = \begin{cases} \prod_{p:i_p<U_{n,p}} e^{-\sum_d \sum_{t\notin\{Hpb,Ch,OC\}} x_{p,t,d}\cdot\lambda_{t,d}} \frac{(\sum_d \sum_{t\notin\{Hpb,Ch,OC\}} x_{p,t,d}\cdot\lambda_{t,d})^{i_p-\sigma_{p,n}}}{(i_p-\sigma_{p,n})!} \cdot \\ \prod_{p:i_p=U_{n,p}} \left(1-\sum_{k<i_p} e^{-\sum_d \sum_{t\notin\{Hpb,Ch,OC\}} x_{p,t,d}\cdot\lambda_{t,d}} \frac{(\sum_d \sum_{t\notin\{Hpb,Ch,OC\}} x_{p,t,d}\cdot\lambda_{t,d})^{k-\sigma_{p,n}}}{(k-\sigma_{p,n})!}\right) \\ \qquad \text{if } i_p \geq \sigma_{n,p} \quad \forall p \\ 0 \qquad \text{otherwise} \end{cases}$$

The number of surgeries during the OR task is deterministic distributed (Assumption 7). Therefore, the transition probability for person p to state i_p from state $\sigma_{p,n}$ is the probability that a surgeon performs $i_p - \sigma_{p,n} - \sum_d x_{p,OR,t} \cdot \lambda_{OR,d}$ surgeries during the scheduled shifts. This is because the number of surgeries done during the OR task is known and no surgeries can be done during the Hpb, Ch and OC tasks. The transition probabilities are shown below.

$$\mathbb{P}\Big(i \mid \sigma_n, a\Big) = \begin{cases} \prod_{p:i_p<U_{n,p}} e^{-\sum_d \sum_{t\notin\{Hpb,Ch,OC,OR\}} x_{p,t,d}\cdot\lambda_{t,d}} \frac{(\sum_d \sum_{t\notin\{Hpb,Ch,OC,OR\}} x_{p,t,d}\cdot\lambda_{t,d})^{i_p-\sigma_{p,n}-\sum_d x_{p,OR,t}\cdot\lambda_{OR,d}}}{(i_p-\sigma_{p,n}-\sum_d x_{p,OR,t}\cdot\lambda_{OR,d})!} \cdot \\ \prod_{p:i_p=U_{n,p}} \Big(1-\sum_{k<i_p} \\ e^{-\sum_d \sum_{t\notin\{Hpb,Ch,OC,OR\}} x_{p,t,d}\cdot\lambda_{t,d}} \frac{(\sum_d \sum_{t\notin\{Hpb,Ch,OC,OR\}} x_{p,t,d}\cdot\lambda_{t,d})^{k-\sigma_{p,n}-\sum_d x_{p,OR,t}\cdot\lambda_{OR,d}}}{(k-\sigma_{p,n}-\sum_d x_{p,OR,t}\cdot\lambda_{OR,d})!}\Big) \\ \qquad \text{if } i_p - \sum_d x_{p,OR,t}\cdot\lambda_{OR,d} \geq \sigma_{n,p} \quad \forall p \\ 0 \qquad \text{otherwise} \end{cases}$$

8.4 Solving model

The model described in Section 8.3 cannot be solve by dynamic programming, due to the curse of dimensionality. Therefore, only small instances can be solved as is done in Chapter 7 for the generic model. We can use the structure found in the results of the generic model to approximate the results using an ILP. From Chapter 7 follows that the optimal action for most of the states is the action for which the expected state after the schedule is as close as possible to the line where the number of orders is equally divided over the persons. Therefore, we can approximate the optimal action by solving the ILP described in Chapter 4 and add the costs of the distribution of the number of completed surgeries over the surgeons. This model will be described in Section 8.5.

8.5 Integer linear program

The ILP that approximates the optimal action of the model described in Section 8.1 is shown below. We will adjust the ILP described in Chapter 4. Below is described what should be added to the model described in Chapter 4.

First, we need a parameter that contains the information of the number of completed orders per surgeon before the new schedule that will be made. This parameter is shown below.

$\sigma_s \in \mathbb{N}$ number of surgeries surgeon s has completed before the schedule made

The next parameter is equal to the arrival rate of surgeries during the different tasks.

$$\lambda_{t,d} \quad \in \quad \mathbb{N} \qquad \text{the arrival rate of surgeries during task } t \text{ on day } d$$

The following variable is equal to the maximum number of expected completed surgeries after the new schedule per surgeon.

$$i \quad \in \quad \mathbb{N} \qquad \text{maximum number of expected surgeries after the new schedule}$$

We want to take into account the work week of the surgeon. This means, the standard availability of the surgeon. When for example one surgeon is never available on Wednesday, Thursday and Friday, this surgeon should have done fewer surgeries than a surgeon who is standard every day available (we do not take into account the days off). The parameter shown below is used for this availability.

$$AV_s \quad \in \quad \mathbb{R} \qquad \text{number of hours work week surgeon } s \text{ divided by the number of hours full time work week}$$

The following constraint has to be added. We need a constraint to determine the maximum number of expected completed surgeries after the new schedule per surgeon. To do this, we have to take into account if the surgeon works full time. This is done with constraint (8.1). We do not take into account the surgeons who can only be scheduled at the organ removal shift or at the organ removal shift, outpatient clinic and operating room.

$$\begin{aligned}
&\frac{1}{AV_s}\Bigg(\sigma_s + \sum_{t \notin \{T1,M\}} \sum_{d:d \bmod 7 \neq 6} x_{s,t,d} \cdot \lambda_{t,d} + \sum_{\substack{d:d \bmod 14 \in \\ \{1+7\cdot(1-MW),3+7\cdot(1-MW)\}}} x_{s,M,d} \cdot (0.5 \cdot \lambda_{M,d} + \lambda_{M,d+1} + 0.5 \cdot \lambda_{M,d+2}) \\
&+ \sum_{\substack{d:d \bmod 14 = \\ 5+7\cdot MW}} x_{s,M,d} \cdot (0.5 \cdot \lambda_{M,d} + \lambda_{M,d+1} + \lambda_{M,d+2} + 0.5 \cdot \lambda_{M,d+3}) + \sum_{t \neq M} \sum_{d:d \bmod 7 = 6} x_{s,t,d} \cdot (\lambda_{t,d} + \lambda_{t,d+1}) \\
&+ \sum_{\substack{d:COMB_{\lceil \frac{d}{7} \rceil}=1 \\ d \bmod 7 \neq 6}} x_{s,T1,d} \cdot \lambda_{T1,d} + \sum_{\substack{d:COMB_{\lceil \frac{d}{7} \rceil}=0 \\ d \bmod 7 \neq 6}} x_{s,T1,d} \cdot \sum_{d'=d}^{d+4} \lambda_{T1,d'}\Bigg) \leq i \\
&\forall s \quad \text{st} \quad \sum_t TS_{s,t} \neq TS_{s,M} + TS_{s,OC} + TS_{s,OR}
\end{aligned} \tag{8.1}$$

We will equally divide the performed surgeries over the surgeons by minimizing i. For this, we need to determine the cost factor for the maximum number of surgeries performed (the factor before i in the objective function). The shortages, combined tasks, extended duration tasks and different surgeons Hpb/Ch are more important than the maximum number of surgeries scheduled. However, the maximum number of surgeries scheduled is more important than equally dividing tasks over the surgeons. Therefore, the importance factor of the new category equally dividing the number of performed surgeries has an importance factor between the importance factor of the category different surgeons Hpb/Ch and the importance factor of the category equally divide tasks. The new importance factors for the different categories are shown in Table 8.1.

Table 8.1: The importance factors for the different categories.

Possibility	Factor
Shortage	100 000
Combined tasks	10 000
Extended duration task	1 000
Different surgeons Hpb/Ch	100
Equally divide performed surgeries	10
Equally divide tasks	1

Now we need to determine the importance factor within the new category. We will do this the same as done for the factors for equally dividing the tasks. We will divide the number of surgeons over who we want to equally divide the number of surgeries by the total arrival rate of tasks during the schedule. By doing this we normalise the maximum number of surgeries done per surgeon by the average number of surgeries per surgeon. The average number of surgeries for a surgeon that works full time is equal to the total number of surgeries during the schedule divided by the surgeons available. This is shown below

$$\frac{\sum_{\substack{s:\sum_t TS_{s,t} \neq \\ TS_{s,M}+TS_{s,OC}+TS_{s,OR}}} AV_s}{\sum_t \sum_d \lambda_{t,d}}$$

Now we will include this in the objective function. This is shown in the new objective function (8.2).

$$\begin{aligned}
\text{minimize} \quad & 2\,100\,000 \cdot \sum_{d:d \bmod 14=5+7\cdot MW} (d_{M,d} + dm_d) + 1\,400\,000 \cdot \sum_{\substack{d:d \bmod 14\in\{3+7\cdot(1-MW),\\ 1+7\cdot(1-MW)\}}} (d_{M,d} + dm_d) \\
& + 800\,000 \cdot \sum_{d:d \bmod 7=6} d_{T1,d} + 400\,000 \cdot \Bigg(\sum_{\substack{d:d \bmod 7\neq 6,\\ COMB_{\lceil \frac{d}{7} \rceil}=1}} d_{T1,d} + 5 \cdot \sum_{\substack{d:d \bmod 7\neq 6,\\ COMB_{\lceil \frac{d}{7} \rceil}=0}} d_{T1,d} + \sum_w dhpb_w \\
& + \sum_d ds_d \Bigg) + 600\,000 \cdot \sum_{d:d \bmod 7=6} d_{T2,d} + 300\,000 \cdot \sum_{d:d \bmod 7\neq 6} d_{T2,d} + 200\,000 \cdot \sum_d d_{Td,d} \\
& + 100\,000 \cdot \sum_d \sum_{t\in\{Ch,Hpb\}} d_{t,d} + 50\,000 \cdot \sum_d \sum_{t\in\{OC,OR\}} d_{t,d} + 30\,000 \cdot \sum_d u_d + 20\,000 \cdot \sum_d z_d \\
& + 20\,000 \cdot \sum_d y_d + 10\,000 \cdot \sum_d w_d + 3\,000 \cdot \sum_w wk_w + 2\,000 \cdot \sum_w lm_w + 1\,000 \cdot \Big(\sum_d fd_d + fdm + fdtu \Big) \\
& + 100 \cdot \sum_t \sum_w (ih_{s,w} + ic_{s,w}) + 10 \cdot \frac{\sum_{\substack{s:\sum_t TS_{s,t}\neq \\ TS_{s,M}+TS_{s,OC}+TS_{s,OR}}} AV_s}{\sum_t \sum_d \lambda_{s,d}} \cdot i + \frac{\sum_s TS_{s,M}}{DTS_M + \sum_s DT_{s,M}} \cdot mds_M \\
& + \frac{\sum_s TS_{s,T1}}{DTS_{T1} + \sum_s DT_{s,T1}} \cdot (mds_{T1} + \frac{1}{2} \sum_{\substack{d:d \bmod 7\notin\{6,0\},\\ COMB_{\lceil \frac{d}{t} \rceil}=0}} \sum_s FWT1_s \cdot x_{s,T1,d}) + \frac{\sum_s TS_{s,T2}}{DTS_{T2} + \sum_s DT_{s,T2}} \cdot \\
& mds_{T2} + \frac{\sum_s TS_{s,OR}}{DTS_{OR} + \sum_s DT_{s,OR}} \cdot mds_{OR} + \frac{|S|}{\sum_t DTS_t + \sum_t \sum_s DT_{s,t}} \cdot md
\end{aligned} \tag{8.2}$$

8.6 Case-study specific input

To make a quarter schedule while we take into account the number of performed surgeries, we need to assign values to the parameters of the ILP. We will use the same input as described in Section 5.2. However, we also need to assign values to the new parameters described in Section 8.5. We will start with $\sigma_s = 0$ for all surgeons, because no data is available on the number of surgeries completed per surgeon. To determine the arrival rate of surgeries during the different tasks we have the data of the surgeries performed from 2014 till 2017. This data we received by collecting the surgeries entered in HiX where the first surgeon was one of the transplant surgeons working on the transplant department that time. However, we do not know during which tasks these surgeries are performed because this data was not available. However, we do know which surgeries are done during the organ removal shift and we have this data from 2015 till April 2018. This data we received from the transplant center. Therefore, we will approximate the arrival rates of the surgeries during the different tasks and we will assume that this arrival rate does not depend on the day, except for the weekend. This is done in Section 8.6.1 – 8.6.6. The availability of the surgeons is shown in Section 8.6.7

8.6.1 Arrival rate surgeries during M

To determine the arrival rate of surgeries during the organ removal shift, we assume that the arrival rate does not depend on the day. We first look at the number of days the organ removal shift is scheduled from 2015 till April 2018. When a Friday is scheduled, this day counts for half a day. This is because the organ removal shift is on Friday scheduled till/from 12:00 o'clock. Then we will divide the total number of surgeries performed by the number of days the organ removal shift is scheduled. This gives the arrival rate of surgeries during one day. The arrival rates are shown below. When only half a day is scheduled during a day, we will divide the arrival rate by two. This is done in constraint (8.1).

$$\lambda_{M,d} = \begin{cases} 0.43 & \text{if } d \bmod 14 \in \{7\cdot MW, 6+7\cdot MW, 1+7\cdot(1-MW), 2+7\cdot(1-MW), \\ & 3+7\cdot(1-MW), 4+7\cdot(1-MW), 5, 12\} \\ 0 & \text{otherwise} \end{cases}$$

8.6.2 Arrival rate surgeries during Ch, Hpb and OC

During the Ch, Hpb and OC tasks no surgeries are done. Therefore, the arrival rate of surgeries is equal to zero for these tasks.

$$\lambda_{t,d} = 0 \qquad t \in \{Ch, Hpb, OC\}$$

8.6.3 Arrival rate surgeries during OR

To determine the arrival rate of surgeries during the OR we will first determine the arrival rate of surgeries during shifts on Monday and Tuesday, because no OR is scheduled during these days. Then we will determine the arrival rate of surgeries during Wednesday, Thursday and Friday. For the arrival rate of surgeries during the OR task, we will take the difference of these two arrival rates. This is done because the same shifts are scheduled on Wednesday, Thursday and Friday as on Monday and Tuesday. However, on Wednesday, Thursday and Friday the OR task is scheduled where surgeries are performed and this task is not scheduled on Monday and Tuesday. The arrival rate used is shown below.

$$\lambda_{OR,d} = \begin{cases} 1.29 & \text{if } d \bmod 7 \in \{3,4,5\} \\ 0 & \text{otherwise} \end{cases}$$

8.6.4 Arrival rate surgeries during $T1$

We will determine an arrival rate of surgeries during the $T1$ shift on weekdays and an arrival rate of surgeries during the $T1$ shift on weekends. We will determine two different arrival rates, because the $T1$ shift during Saturday or Sunday is a 24h shift and the $T1$ shift during a weekday is from 18:00 o'clock till 8:00 o'clock. We will count the number of surgeries performed during the time moments the $T1$ shift is scheduled and divide by the number of days. This is done for the days during the week $T1$ shift and for the days during the weekend.

$$\lambda_{T1,d} = \begin{cases} 0.29 & \text{if } d \bmod 7 \in \{1,2,3,4,5\} \\ 0.46 & \text{otherwise} \end{cases}$$

8.6.5 Arrival rate surgeries during Td

We will determine the arrival rate of surgeries during the Td shift the same as done for the $T1$ shift in Section 8.6.4. Now we take into account the surgeries during weekdays from 8:00 o'clock till 18:00 o'clock and we will only take into account the Monday and Tuesday. Otherwise we also count the surgeries during the OR.

$$\lambda_{Td,d} = \begin{cases} 0.69 & \text{if } d \bmod 7 \in \{1,2,3,4,5\} \\ 0 & \text{otherwise} \end{cases}$$

8.6.6 Arrival rate surgeries during $T2$

We will assume that the surgeon with the $T2$ shift performs all the surgeries that the surgeons with $T1$ and Td shift perform. Therefore, we can add these arrival rates.

$$\lambda_{T2,d} = \begin{cases} 0.99 & \text{if } d \bmod 7 \in \{1,2,3,4,5\} \\ 0.46 & \text{otherwise} \end{cases}$$

8.6.7 Availability surgeons

All surgeons except for surgeon 11 work full time. Surgeon 11 is not available on Thursday and Friday. However, surgeon 11 can be scheduled at the organ removal shift the first Friday of the organ removal shift week. This surgeon also has a whole non-clinical day per week, but the end of the organ removal shift can be scheduled during this day.

$$AV_s = \begin{cases} \frac{1}{2} & \text{if } s = 11 \\ 1 & \text{otherwise} \end{cases}$$

8.7 Case-study results

The model described in Section 8.5 is programmed in AIMMS and solved with GUROBI, where we use the parameters as described in Section 8.6. We stopped the solver after one day when the best solution had an objective value of $28\,020\,722.65$, the optimality gap is 0.13% and the LP bound is equal to $27\,982\,897.90$. This means that it is possible that a better solution can be found. However, the value of the objective function of the optimal solution is not better than $27\,982\,897.90$. The quarter schedule is shown in Figure 8.1, the shortages at the different tasks in Figure 8.2, the shortages for the MOD certified surgeon in Figure 8.3, the shortages for the senior surgeon in Figure 8.4, the shortages for the hpb surgeon in Figure 8.5 and the distribution of the different tasks over the surgeons in Figure 8.6. The expected number of surgeries per surgeon during the schedule are shown in Table 8.2.

All surgeons work full time, except for surgeon 11. This surgeon has an availability factor of one half. Therefore, we should multiply the expected number of surgeries of surgeon 11 by two when we want to compare this number with the other surgeons. Also, we do not take into account surgeons who can only be scheduled at the organ removal shift (surgeon 13) and surgeons who can only be scheduled at the organ removal shift, outpatient clinic and the operating room (surgeon 12). We see that the expected number of surgeries performed is equal to zero for surgeon 9. This is because this surgeon is not available during the schedule made. We also see that the expected number of surgeries for surgeon 3 is much lower than for other surgeons. This can be explained by the fact that surgeon 3 can only perform surgeries during the organ removal shift and this shift can only be scheduled every other week for at most three days, because this surgeon has the standard non-clinical day on Wednesday. The organ removal shift can be scheduled on Wednesday morning, but the organ removal shift cannot be scheduled from Monday till Friday for this surgeon. The expected number of surgeries for surgeon 7 also is much lower than for other surgeons. This can be explained by the fact that this surgeon is only available till day 31. The expected number of surgeries of all other surgeons varies between 27.36 and 29.16.

Table 8.2: The expected number of surgeries performed in the fourth quarter per surgeon.

Surgeon	Expected number of surgeries
1	29.16
2	27.36
3	5.16
4	28.56
5	29.06
6	29.15
7	9.15
8	29.14
9	0.00
10	29.16
11	14.17
12	13.76
13	0.00

Surgeon\Day	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	V	V	V	V	V	T2	T2	T2/Hpb	Hpb	G/Ch/Hp	Td/Ch	NCD				V	V	V	V	V	V
2	T2	T2/Hpb	or	NCD	Td			t1	t1	t1	t1	t1			T2	T2	V	V	V	V	
3	Hpb	NCD	Hpb	Hpb	Hpb			M	M	NCD/M	Hpb	Hpb					NCD	V	V	V	V
4	Td	Td	or	or	NCD			OC	T2	T2/OR	V	V	V		t1	t1	t1	t1	t1		
5	G/Pali	NCD	T2	T2	T2/OR	t1	t1	FromT1	NCD	M	M	G/M	G	G	G/Td	NCD	Td	Td	T2/OR	T2	T2
6	V	V	V	V	V	V	V	Ch	Ch	or	NCD	Td/Ch	t1	t1	FromT1	Ch	NCD	Ch	or	G	G
7	t1	t1	t1	t1	t1			Td	NCD	M	M	M	T2	T2	Ch	NCD	Ch/Hpb	or	Ch/Hpb		V
8	FromT1	NCD	Td	Td	M	M	M	M		NCD	T2/OR	T2/OR			Hpb	Hpb	NCD	Hpb	Td	V	V
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	Ch	NCD	Ch	Ch	Ch			OC	NCD	Td	V	V	G		OC	NCD	T2/OR	T2	V	t1	t1
11	OC	Ch	NCD	V	V/M	M	M	M	G/Td	V	V	V	G	G	OC	Td	NCD	V	V/M	M	M
12	V	V		or	V			M	M	M	or	V			V	V	V	or	V		
13																					

Surgeon\Day	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42
1	M	M	M	M	M	T2	T2	T2	Ch	NCD	or	Td/Ch			T2/Hpb	T2	T2/Hpb	Td	NCD	T2	T2
2	T2/Hpb	T2	V	V	V	t1	t1	FromT1	T2	T2/OR	NCD	V	V	V	V	Td	V	V	V	V	
3	V	V	V	V	V	V	V	Hpb	Hpb	NCD	Hpb	Hpb			M	M	NCD/M	Hpb	Hpb		
4	M	M	M	Ch	V	V	V	V	V	V	V	V	t1	t1	V	Ch	Td/Ch	Ch	NCD	t1	t1
5	V	Hpb	T2/Hpb	T2/Hpb	V	G	G	G/Ch	NCD	Ch/Hpb	Ch	M	M	M	M	NCD	M	M	G/M	G	G
6	V	V	V	V	V	V		t1	t1	t1	t1	t1	G	G	Td/Ch	Hpb	NCD	T2/OR	T2/Td/Ch	G	G
7	V	V	V	T1/Td	T1/Td			Td	NCD	or	V	V	V	V	V	V	V	V	V	V	V
8	V	V	V	V	V	V	V	OC		NCD	T2	T2/OR			t1	t1	t1	t1	t1	G	G
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	T1/Td	Ch	M	M	M			V	NCD	Td	Td	M	M	M	M	NCD	M	M	M		
11	M	T1/Td	T1/Td	V	V	G	G	OC	Td	NCD	V	V	T2	T2	M	M	NCD/M	V	V	G	G
12	V	V	V	V	V			V	V	V	or	V			V		V	V	V		
13																					

Surgeon\Day	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
1	t1	t1	t1	t1	t1			Ch	T2/Hpb	T2/Hpb	Ch/Hpb	NCD			t1	t1	t1	t1	t1		
2	T2	T2	T2/OR	NCD	M	M	M	M	V	V	V	V			Td	V	Td	NCD	M	M	M
3	Hpb	Hpb	NCD	Hpb	Hpb			M	M	NCD/M	V	V	V	V	Hpb	Hpb	NCD	Hpb	Hpb		
4	FromT1	Td	or	Td	NCD			t1	t1	t1	t1	t1			T2	T2	T2/OR	or	NCD	V	V
5	G/Pali	NCD	Td	T2	T2/OR	t1	t1	FromT1	NCD	M	M	G/M	G	G	G	NCD	Ch/Hpb	Ch	Ch	t1	t1
6	G/Pali		NCD	or	Td	G	G	G/Td	Ch	NCD	or	Td/Ch	T2	T2	Ch	Ch	NCD	T2	T2/OR		
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	Ch	Ch	NCD	Ch	Ch	T2	T2	T2/Hpb	Td	NCD	T2/OR	T2/Hpb	t1	t1	FromT1	Td	NCD	Td	Td		
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10		NCD	Ch/Hpb	or	M	M	M	M	NCD	Td/Ch	Td	V	V	V	OC	NCD	or	or	M	M	M
11	Td	G	V	V	V	V		M	M	NCD/M	V	V	G	G	OC		NCD	V	V	T2	T2
12	V	V	V		V			V		V	V	V			V	V	V		V		
13																					

Surgeon\Day	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84
1	Td/Ch	Ch	T2/Td/Hp	T2/OR	NCD					NCD	Ch	M	M	M	M	T2	T2/OR	Td	NCD	t1	t1
2	M	Td	V	V	V			Ch	Ch	Ch/Hpb	NCD	Ch	T2	T2	T2/Hpb	Hpb	Ch/Hpb	V	V	V	V
3	M	M	NCD/M	Hpb	Hpb			Hpb	Hpb	NCD	Hpb	Hpb			M	M	NCD/M	Hpb	Hpb	V	V
4	V	V	V	V	V	V	V	t1	t1	t1	t1	t1			Ch	Td	Td	Ch	NCD	V	V
5	FromT1	NCD	M	M	G/M	G	G	G/Td	NCD	Td	or	Td	t1	t1	FromT1	NCD	M	M	G/M	G	G
6	t1	t1	t1	t1	t1			OC		NCD	or	or			t1	t1	t1	t1	t1	V	V
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	T2/Hpb	T2/Hpb	NCD	Ch	Td/Ch	t1	t1	FromT1	Td	NCD	Td	M	M	M	M	Ch	NCD	T2/OR	T2/Td/Ch	V	V
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	M	NCD	V	Td	T2/OR	T2	T2	OC	NCD	T2/OR	T2	T2/OR			Td	NCD	M	M	M	G	G
11	M	M	NCD/M	V	V	G	G	T2	T2	NCD	V	V			M	M	NCD/M	V	V	G	G
12	V		V	V	V			V	V	V		V			OC	V	or	or	V		
13																					

Surgeon\Day	85	86	87	88	89	90	91	92	93	94	95	96	97	98
1	V	V	V	V	V	V	V	OC		T1/Td	T1/Td	NCD		
2	V	V	V	V	V	V	V	T2/Hpb	T2	T2/Hpb	NCD	T1/Td	t1	t1
3	V	V	V	V	V	V	V	M	M	NCD/M	Hpb	Hpb		
4	V	V	V	V	V	T2	T2	M	M	M	or	NCD	T2	T2
5	G/Ch		t1	T1/Td	T1/Td			V	V	V	V	V	V	V
6	V	V	V	V	V	V	V	T1/Td	t1	NCD	T2	T2/OR		
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	V	V	V	V	V	V	V	Ch		NCD	Ch	Ch		
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	T2/Hpb	T2		T2/Hpb	T2/Hpb			V	V	V	V	V	V	V
11	T1/Td	t1		V	V	t1	t1	V	V	V	V	V	V	V
12	V	V	V	or	V			OC		or	or	or		
13														

Figure 8.1: The schedule of the different tasks during quarter 4 when we take into account the number of surgeries performed per surgeon.

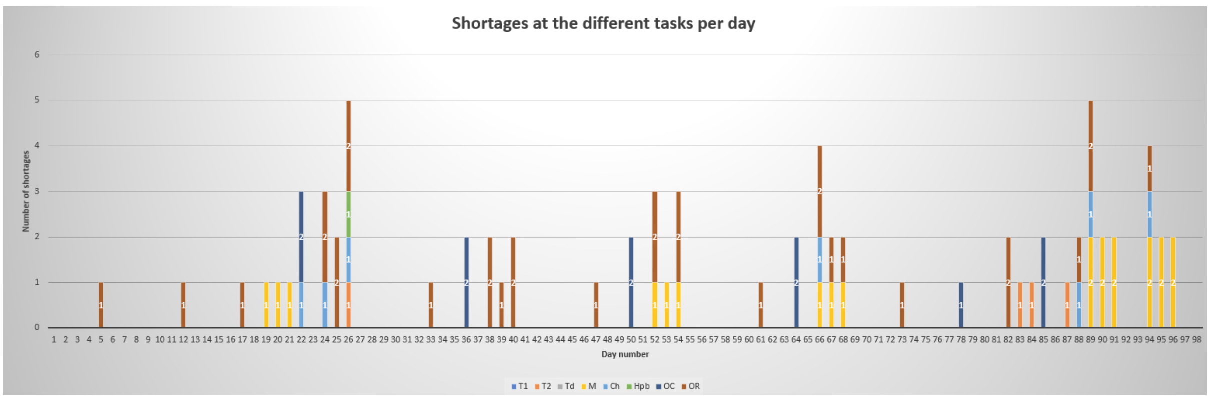

Figure 8.2: The generated shortages at the different tasks in the quarter schedule per day.

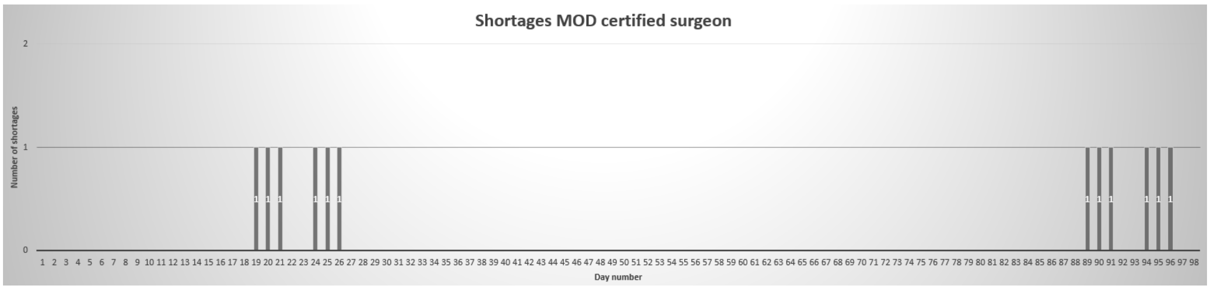

Figure 8.3: The shortages of the MOD certified surgeon in the quarter schedule per day when we take into account the number of surgeries performed per surgeon.

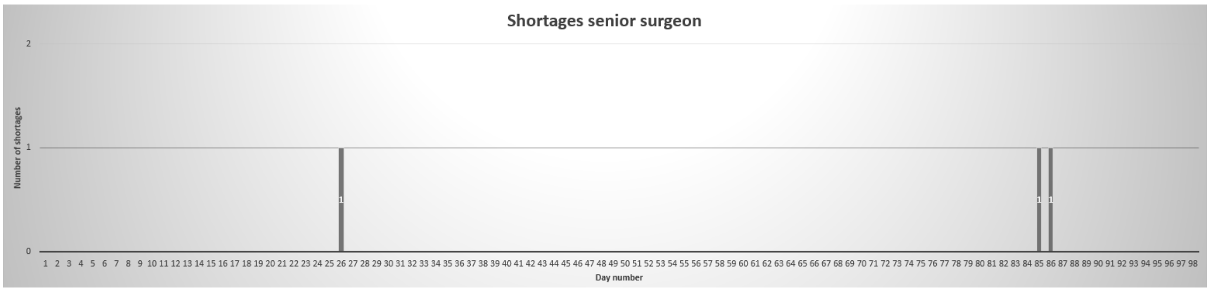

Figure 8.4: The shortages of the senior surgeon in the quarter schedule per day when we take into account the number of surgeries performed per surgeon.

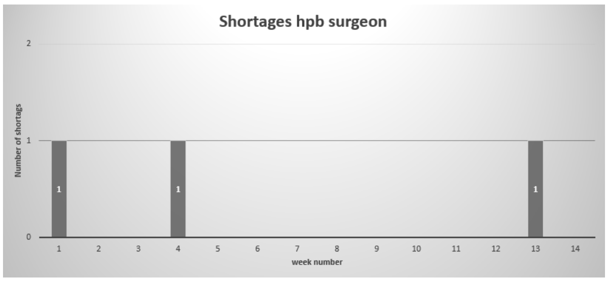

Figure 8.5: The shortages of the hpb surgeon in the quarter schedule per week when we take into account the number of surgeries performed per surgeon.

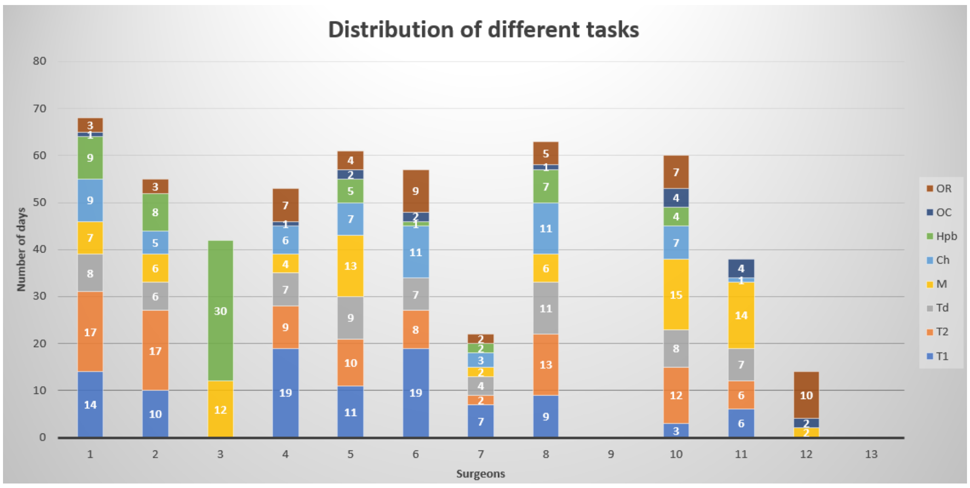

Figure 8.6: The distribution of the different tasks over the surgeon when we take into account the number of surgeries performed per surgeon.

8.8 Compare results with quarter schedule without taken into account number of surgeries

In this section, we will compare the results shown in Section 8.7 (where we do take into account the number of surgeries performed per surgeon) with the result shown in Section 5.3 (where we do not take into account the number of surgeries performed per surgeon). We will look at the expected number of surgeries per surgeon. The expected number of surgeries performed in the fourth quarter per surgeon when we do take into account the number of surgeries performed per surgeon are shown in Table 8.2. The expected number of surgeries performed in the fourth quarter per surgeon when we do not take into account the number of surgeries performed per surgeon are shown in Table 8.3. All surgeons work full time, except for surgeon 11. This surgeon has an availability factor of one half. Therefore, we should multiply the expected number of surgeries of surgeon 11 by two when we want to compare this number with the other surgeons. We see that the maximum expected number of surgeries of Table 8.2 is equal to 29.16, while the maximum expected number of surgeries of Table 8.3 is equal to 36.24. Therefore, the maximum expected number of surgeries is decreased by 20% when we use the model described in Section 8.5 instead of the model described in Chapter 4.

When we compare Figures 5.2 – 5.5 (shortages when we do not take into account the number of surgeries) with Figures 8.2 – 8.5 (shortages when we do take into account the number of surgeries) we see that the shortages do not change. Also, the combined tasks, extended duration of tasks and the number of different surgeons on the Hpb/Ch task per week do not change. This is because these categories have higher importance factors than the category equally dividing the number of surgeries performed. When we compare Figure 5.6 (distribution of the different tasks over the surgeons when we do not take into account the number of surgeries) with Figure 8.6 (distribution of the different tasks over the surgeons when we do take into account the number of surgeries) we see that the distribution of the different tasks over the surgeons does change. The tasks are divided more equally when we do not take into account the number of surgeries performed. This can be explained by the fact that the distribution of the tasks over the surgeons has a lower importance factor than the distribution of the surgeries over the surgeons. However, it is possible that the tasks can be distributed more equally over the surgeons in the case that we want to distribute the number of surgeries equally over the surgeons, because it is possible that the optimal solution is better than the solution shown in Section 8.7. This is because we stopped the solver and we had a optimality gap of 0.13%.

Table 8.3: The expected number of surgeries performed in the fourth quarter per surgeon when we use the model described in Chapter 4.

Surgeon	Expected number of surgeries
1	31.4
2	23.88
3	5.16
4	25.01
5	31.23
6	24.54
7	6.57
8	36.24
9	0.00
10	29.8
11	16.24
12	13.76
13	0.00

Chapter 9

Discussion

Section 9.1 will discuss the model for scheduling tasks, Section 9.2 the generic model for equally dividing the orders, Section 9.3 the model including the number of surgeries done per surgeon and Section 9.4 will discuss further research.

9.1 Model for scheduling tasks

In the model for scheduling the different tasks, we want to equally divide certain tasks and the total number of tasks scheduled over the transplant surgeons. This is done by minimizing the maximum number of days scheduled at this task (or total number tasks scheduled). However, it is possible that the tasks could be divided more equally than only minimizing the maximum number. When for example one surgeon is scheduled many days at a task and this number cannot be reduced, but it is possible to divide the tasks of the other surgeons more equally. This is not included in the model, but it could be done. However, the solving time will increase when we do this. We will explain this with an example.

We have three surgeons and one task should be scheduled over these three surgeons for six days. Surgeon 2 and 3 are not available during day 1,2 and 3. Therefore, surgeon 1 must be schedule at the task on day 1, 2 and 3. When we minimize the maximum number of days scheduled at the task per surgeon during this schedule, we do not schedule surgeon 1 during day 4, 5 and 6. However, it does not matter how we schedule the task during day 4, 5 and 6 over surgeon 2 and 3. Two possibilities are shown in Figure 9.1. While both schedules have the same value when we minimize the maximum number of days scheduled at the task per surgeon, we prefer the second one.

Surgeon\Day	1	2	3	4	5	6
1	Task	Task	Task			
2	V	V	V	Task	Task	Task
3	V	V	V			

Surgeon\Day	1	2	3	4	5	6
1	Task	Task	Task			
2	V	V	V	Task	Task	
3	V	V	V			Task

Figure 9.1: Two possible schedules when one task is scheduled over three surgeons.

9.2 Generic mathematical model for equally dividing orders

We hoped to find a structure in the results of small instances of the generic mathematical model for equally dividing the number of order over the persons that we could extend to bigger instances. Most of the states have an optimal action that can be defined as the expected new state as close as possible to the line where the number of orders is equally divided over the persons. However, this is not always the case. When during one schedule a big task (high arrival rate) must be scheduled this structure does not hold. Also when we have the same arrival rates during the schedules we have some exceptions that we could not explain.

9.3 Model including the number of surgeries done per surgeon

In this section the SDP model will be discussed in Section 9.3.1 and the ILP model will be discussed in Section 9.3.2.

9.3.1 SDP

In the SDP model for including the number of surgeries done per surgeon we assume a deterministic distribution for the number of surgeries done during the OR task. However, the OR task is scheduled every Wednesday, Thursday and Friday and once in the two weeks one OR day expires due to the operating room reduction. Also, the number of surgeries done during the OR task depends on the surgeries scheduled. One surgery takes less time than another one and therefore during some days fewer surgeries can be scheduled than during other days. To say something about the distribution of the number of surgeries done during an OR tasks data of these numbers during the OR task is required. At the shifts we assumed a Poisson process as arrival process, because the surgeries arrive independent of each other. To check if this assumption is correct data of the number on surgeries done during the different shifts is required.

We also assumed that a surgeon scheduled at multiple tasks simultaneously is always available at both tasks. Practically this is not possible. When a surgeon is scheduled at the OR and at the $T2$ shift and the surgeon is performing a surgery for the OR task, this surgeon cannot at the same time perform another surgery of the $T2$ task. Therefore, this should be taken into account in the arrival process of surgeries for this surgeon.

We do not take into account the differences between first and second surgeon during a surgery. However, for the surgical experience this is a difference. Therefore, this should be taken into account.

Last, we assumed that the new quarter schedule can be made after the end of the previous schedule. This assumption is made, because in this case we know the number of surgeries done after the previous schedule before we make the new schedule. In practice, the schedules must be made earlier. This is because the surgeons must know earlier when they have to work during the weekend or when they have night shifts. Therefore, the rule is that the schedules are known six weeks before the start of the schedule. This should be included in the model. This can be done by using the number of surgeries done six weeks before the start of the new schedule. The number of surgeries done during the last six weeks can be added to the transition probabilities. We used the transition probabilities for a number of surgeries during the schedule. However, in this case the last six weeks of the old schedule should be taken into account.

9.3.2 ILP

We approximate the optimal solution of the SDP by minimizing the maximum number of expected surgeries per surgeon at the end of the schedule in an ILP. This is done because in the generic SDP most of the states have the optimal action equal to equally dividing the expected number of orders after the schedule. However, we do not know how good this approximation is.

To add the objective to minimize the maximum number of expected surgeries per surgeon at the end of the schedule to the ILP model for making the schedules without taken into account the number of surgeries completed, we have to know the arrival rate of surgeries during the different tasks. We assumed that these arrival rates do not depend on the day, time of the year etcetera. We only take into account the duration of the shift. This is done because we only have the data of the surgeries performed, but we do not know during which task the surgery occurred. However, when we would take this into account the expectations of the number of surgeries would be more accurate.

Because we do not know during which task the surgeries are done, we approximated the arrival rates. For the organ removal shift we do have the data of the number of surgeries done during this shift. However, because we do not have the data for the other tasks we only used the mean number of surgeries and did not look at differences between days etcetera. For the OR task we determined the mean number of surgeries done on Wednesday, Thursday and Friday and subtract the mean number of surgeries done on Monday and Tuesday. However, we did not take into account the public holidays. It is also possible that during some days no OR task was scheduled, because for example no surgeons were available. We also did not take this into account. Therefore, it is possible that the mean number of surgeries during the OR is higher. It is also possible that during shifts on Monday and Thursday fewer or more surgeries arrive than during Wednesday, Thursday and Friday. The mean number of surgeries during the OR task can be different because of this. For the arrival rate of surgeries during the $T1$ shift during weekdays and the arrival rate of the Td shift we only take into account the Monday and Tuesday. Therefore, it is possible that the arrival rates are different because it is possible that fewer or more surgeries are done during the other days. We assumed that the surgeon scheduled at the $T2$ shift performed all the surgeries done during the $T1$ and Td shift. This is not the case, because some surgeries can be done by one surgeon. Therefore, the arrival rate of surgeries will be lower in practice.

9.4 Further research

Further research can be done to the risks of combining certain tasks for the surgeons. For example if we combine the $T2$ task with the OR task, what can we say about the probability that the surgeon performs a scheduled surgery from the OR task, while this surgeon should also be available for the $T2$ task and this surgeon is not?

Second, research can be done to schedule the surgeons differently. Now the transplant surgeons are scheduled at different shifts and during each shift a surgery can be performed. However, also a group of surgeons can be scheduled at shifts and not specified which shift. In this group a rank order can be determined. The first surgeon is called first when a surgery during one of the shifts should be performed. By using this structure it is possible that fewer surgeons are needed for the shifts, because the arrival rate of surgeries during shifts is not very high.

Also, research could be done to take into account the type of surgery done instead of only takeing into account the number of surgeries to equally divide the surgical experience. Some surgeries are easier than other surgeries. This could be done by equally dividing the time spent on surgeries. However, one surgeon may be faster than another surgeon. Therefore, this should also be taken into account. Also the difference between first and second surgeon can be taken into account.

The ILP model that takes into account the number of surgeries performed could be tested. This can for example be done by simulation.

Last, research could be done to develop an ADP to overcome the curse of dimensionality and approximate the optimal solution of the SDP. This may give better solutions than the approximation of the ILP.

Chapter 10

Conclusion and recommendations

This chapter gives the conclusion and recommendations of this research. First, Section 10.1 shows the conclusion. Then, Section 10.2 shows the recommendations.

10.1 Conclusion

The transplant surgeons must be scheduled over different tasks. Different transplant shift must be scheduled where transplants can be done that are not scheduled. The different transplant shifts are the shift during office hours (Td), the shift outside office hours ($T1$) and the backup transplant shift ($T2$). The organ removal shift (M) must also be scheduled every other week. Also, the chef ward (Ch), chef hpb (Hpb), outpatient clinic (OC) and operating room (OR) tasks must be scheduled during office hours.

We first developed an ILP for scheduling these tasks over the surgeons per quarter. By using this model the optimal schedule can be found given the surgeons and availability of the surgeons. Using this model the schedule for the fourth quarter of 2018 is made. When we use the availability of the surgeons as input, we have shortages at different tasks during the schedule. Even the organ removal shift that is the most important to be scheduled has shortages. This is because this shift must be scheduled by a fixed structure. Two surgeons must be scheduled every other week from Friday 12:00 0'clock till Monday 12:00, two other surgeons are scheduled from Monday 12:00 o'clock till Wednesday 12:00 o'clock and two surgeons must be scheduled from Wednesday 12:00 o'clock till Friday 12:00 o'clock. Therefore, two surgeons must be available during the whole block (consecutive days). When the organ removal shift is scheduled with another structure, fewer shortages occur at the organ removal shift. Three other structures are investigated: old structure (no further constraints), $T2$ structure (two to four consecutive days for the same surgeon) and cooperate with the Erasmus MC (the shift must be scheduled each week and one surgeon must be scheduled). All cases have fewer shortages at the organ removal shift than the current structure.

By developing the ILP model for scheduling tasks over surgeons, we know how to schedule the transplant surgeons. Therefore, the next step is to take into account the surgical experience. This is done with the number of surgeries performed. To do this an SDP model is developed for a more generic case. In this case different tasks are scheduled over persons and during the tasks orders can arrive. The goal is to equally divide the number of completed orders at the end of the horizon. This SDP is solved using DP for various small instances. The goal was to find a structure in the optimal actions that can be extended to bigger instances. Most of the optimal actions are such that the expected new state after the schedule is divided as equally as possible. However, sometimes the optimal action differs a little from this. This SDP is adjusted to the LUMC context. However, due to the curse of dimensionality this SDP cannot be solved using DP. Therefore, we approximated the optimal action by using the ILP model and add the condition that we want to divide the expected number of surgeries equally over the surgeons at the end of the schedule. By doing this the maximal expected number of surgeries at the end of quarter 4 per surgeon is decreased by 20% in comparison with the ILP for scheduling the different tasks. While the shortages, combined tasks, extended duration of tasks and the number of different surgeons per week at the Hpb/Ch task do not change.

10.2 Recommendations

We will recommend the LUMC to use the model for scheduling the transplant surgeons and take into account the number of surgeries performed per surgeon. However, to do this a user friendly application should be developed. By doing this optimal schedules, given the availability of the surgeons, can be made that fulfill all constraints, the program is impartial and the surgeons do not have to spend time at scheduling the different tasks over the surgeons. Also, the mean number of surgeries during a task should be determined. This model can be used to first indicate the shortages that will occur when the requests for the days off have to be approved/rejected. When the days off are fixed the model can be used to make the quarter schedule.

To schedule the organ removal shift more easily it is recommended to use a different structure for scheduling the organ removal shift. One of the following options can be chosen:

- Use the old structure. This means that no further constraints are used for scheduling the organ removal shift. Using this structure the organ removal shift can be scheduled the easiest of the different cases investigated.
- Use the $T2$ structure for scheduling the organ removal shift. This means that the shift is scheduled at least two and at most four consecutive days for one surgeon. By using this structure it can be harder to schedule the organ removal shift than using the old structure. However, the surgeons will be fitter during this shift because it cannot be scheduled longer than four days for the same surgeon.
- Cooperate with Rotterdam and keep the fixed structure. This is harder to schedule because of the fixed structure. However, the differences between the number of surgeons needed per week are zero while this is equal to four at some days with the current structure.

Bibliography

Nederlandse Transplantatie Stichting. Alle transplantatiecentra, 2018. URL https://www.transplantatiestichting.nl/alle-transplantatiecentra. Visit at June 2018.

Marion van der Hoeven. Uitname van organen verloopt beter dankzij zut, 2013. URL https://www.transplantatiestichting.nl/interviews/uitname-van-organen-verloopt-beter-dankzij-zut. Visit at June 2018.

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

Jorne Van den Bergh, Jeroen Beliën, Philippe De Bruecker, Erik Demeulemeester, and Liesje De Boeck. Personnel scheduling: A literature review. *European Journal of Operational Research*, 226(3):367 – 385, 2013.

Mohammad Akbari, M. Zandieh, and Behrouz Dorri. Scheduling part-time and mixed-skilled workers to maximize employee satisfaction. *The International Journal of Advanced Manufacturing Technology*, 64(5):1017–1027, 2013.

T. Majozi and X.X. Zhu. A combined fuzzy set theory and milp approach in integration of planning and scheduling of batch plants—personnel evaluation and allocation. *Computers & Chemical Engineering*, 29(9):2029 – 2047, 2005.

Viktor Dück, Lucian Ionescu, Natalia Kliewer, and Leena Suhl. Increasing stability of crew and aircraft schedules. *Transportation Research Part C: Emerging Technologies*, 20(1):47 – 61, 2012.

E. K. H. Chow, Massie A. B., Muzaale A. D., Singer A. L., Kucirka L. M., Montgomery R. A., Lehmann H. P., and Segev D. L. Identifying appropriate recipients for cdc infectious risk donor kidneys. *American Journal of Transplantation*, 13(5):1227–1234, 2012.

Kyle J van Arendonk, Eric K H Chow, Nathan T James, Babak J. Orandi, Trevor A Ellison, Jodi M. Smith, Paul M. Colombani, and Dorry L Segev. Choosing the order of deceased donor and living donor kidney transplantation in pediatric recipients: a markov decision process model. *Transplantation*, 99 (2):360–366, 2015.

Derya Sever, Lei Zhao, Nico Dellaert, Emrah Demir, Tom Van Woensel, and Ton De Kok. The dynamic shortest path problem with time-dependent stochastic disruptions. *Transportation Research Part C: Emerging Technologies*, 92:42 – 57, 2018.

Eline R. Tsai. Optimal time allocation of an orthopedic surgeon. Master thesis, University of Twente, August 2017.

Peter J.H. Hulshof, Martijn R.K. Mes, Richard J. Boucherie, and Erwin W. Hans. Patient admission planning using approximate dynamic programming. *Flexible Services And Manufacturing Journal*, 28 (1):30–61, 2016.

Peter J. H. Hulshof, Richard J. Boucherie, Erwin W. Hans, and Johann L. Hurink. Tactical resource allocation and elective patient admission planning in care processes. *Health Care Management Science*, 16(2):152–166, 2013.

Warren B. Powell. *Approximate Dynamic Programming, Solving the Curses of Dimensionality*. John Wiley & Sons, USA, NJ, 2011.

H. Zhang, B. de Saporta, F. Dufour, D. Laneuville, and A. Nègre. Stochastic control of observer trajectories in passive tracking with acoustic signal propagation optimisation. *IET Radar, Sonar Navigation*, 12(1):112–120, 2018.

Dimitris Bertsimas and John N. Tsitsiklis. *Introduction to linear optimization*. Athena Scientific, Dynamic Ideas, 1998.

Martin L. Puterman. *Markov Decision Processes Discrete Stochastic Dynamic Programming*. John Wiley & Sons, USA, NY, 2005.

Sheldon M. Ross. *Introduction to Probability Models*. Academic Press, USA, MA, 11th edition, 2010.

Richard Durrett. *Essentials of Stochastic Processes*. Springer, Switzerland, 2nd edition, 2012.

Ralph Grimaldi. *Discrete and Combinatorial Mathematics: An Applied Introduction*. Pearson, England, 5th edition, 2003.

Lloyd N. Trefethen and David Bau III. *Numerical linear algebra*. Siam, USA, PA, 5th edition, 1997.

Appendix A

Model for scheduling tasks

The model for scheduling the different tasks over the surgeons is shown in this appendix.

$$\begin{aligned}
\text{minimize}\quad & 210\,000 \cdot \sum_{d:d \bmod 14 = 5+7\cdot MW} (d_{M,d} + dm_d) + 140\,000 \cdot \sum_{\substack{d:d \bmod 14 \in \{3+7\cdot(1-MW),\\ 1+7\cdot(1-MW)\}}} (d_{M,d} + dm_d) \\
& + 80\,000 \cdot \sum_{d:d \bmod 7 = 6} d_{T1,d} + 40\,000 \cdot \Big(\sum_{\substack{d:d \bmod 7 \neq 6,\\ COMB_{\lceil \frac{d}{7} \rceil}=1}} d_{T1,d} + 5 \cdot \sum_{\substack{d:d \bmod 7 \neq 6,\\ COMB_{\lceil \frac{d}{7} \rceil}=0}} d_{T1,d} + \sum_w dhpb_w \\
& + \sum_d ds_d \Big) + 60\,000 \cdot \sum_{d:d \bmod 7 = 6} d_{T2,d} + 30\,000 \cdot \sum_{d:d \bmod 7 \neq 6} d_{T2,d} + 20\,000 \cdot \sum_d d_{Td,d} \\
& + 10\,000 \cdot \sum_d \sum_{t \in \{Ch,Hpb\}} d_{t,d} + 5\,000 \cdot \sum_d \sum_{t \in \{OC,OR\}} d_{t,d} + 3\,000 \cdot \sum_d u_d + 2\,000 \cdot \sum_d z_d \\
& + 2\,000 \cdot \sum_d y_d + 1\,000 \cdot \sum_d w_d + 300 \cdot \sum_w wk_w + 200 \cdot \sum_w lm_w + 100 \cdot \Big(\sum_d fd_d + fdm + fdtu \Big) \\
& + 10 \cdot \sum_t \sum_w (ih_{s,w} + ic_{s,w}) + \frac{\sum_s TS_{s,M}}{DTS_M + \sum_s DT_{s,M}} \cdot (mds_M) + \frac{\sum_s TS_{s,T1}}{DTS_{T1} + \sum_s DT_{s,T1}} \cdot \\
& (mds_{T1} + \frac{1}{2} \sum_{\substack{d:d \bmod 7 \notin \{6,0\},\\ COMB_{\lceil \frac{d}{7} \rceil}=0}} \sum_s FWT1_s \cdot x_{s,T1,d}) + \frac{\sum_s TS_{s,T2}}{DTS_{T2} + \sum_s DT_{s,T2}} \cdot mds_{T2} \\
& + \frac{\sum_s TS_{s,OR}}{DTS_{OR} + \sum_s DT_{s,OR}} \cdot mds_{OR} + \frac{|S|}{\sum_t DTS_t + \sum_t \sum_s DT_{s,t}} \cdot md
\end{aligned}$$

subject to

$$\begin{aligned}
ih_{s,w} &= 0 && \forall s,w \quad \text{st} \quad TS_{s,Hpb} = 0 \\
ic_{s,w} &= 0 && \forall s,w \quad \text{st} \quad TS_{s,Ch} = 0 \\
mds_t &= 0 && \forall t \in \{Td, Hpb, Ch, OC\} \\
y_d &= 0 && \forall d \quad \text{st} \quad d \bmod 7 \in \{0,6\} \\
w_d &= 0 && \forall d \quad \text{st} \quad d \bmod 14 \neq 5 + 7 \cdot MW \\
z_d &= 0 && \forall d \quad \text{st} \quad d \bmod 7 \in \{0,1,2,6\} \\
u_d &= 0 && \forall d \quad \text{st} \quad d \bmod 7 \in \{0,6\} \\
mc_{d,t,t'} &= 0 && \forall d,t,t' \quad \text{st} \quad d \bmod 7 \in \{0,6\} \\
mc_{d,t,t'} &= 0 && \forall d,t,t' \quad \text{st} \quad t \in \{T1, M, OC, OR\} \\
mc_{d,t,t'} &= 0 && \forall d,t,t' \quad \text{st} \quad t' \in \{T1, M, OC, OR\} \\
mc_{d,t,t'} &= 0 && \forall d,t,t' \quad \text{st} \quad t \prec t' \\
mc_{d,t,t'} &= 0 && \forall d,t,t' \quad \text{st} \quad t' = Hpb, t \in \{T2, Ch\} \\
mcd_d &= 0 && \forall d \quad \text{st} \quad d \bmod 7 \in \{0,6\}
\end{aligned}$$

$$
\begin{aligned}
x_{s,T1,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 7 = 0 \\
x_{s,T1,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 7 \in \{2,3,4,5\}, COMB_{\lceil \frac{d}{7} \rceil} = 0 \\
x_{s,T2,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 7 = 0 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 \notin \{1 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW), \\
& && \quad 5 + 7 \cdot MW\} \\
x_{s,t,d} &= 0 && \forall s,t,d \quad \text{st} \quad t \in \{Td, Ch, Hpb\}, d \bmod 7 \in \{0,6\} \\
x_{s,OC,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 7 \in \{0,2,3,4,5,6\} \\
x_{s,OR,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 7 \in \{0,1,2,6\} \\
x_{s,t,d} &= 0 && \forall s,t,d \quad \text{st} \quad t \notin \{T1, M\}, d \bmod 7 = NCD_s, NNCD_{s,d} = 0 \\
x_{s,t,d} &= 0 && \forall s,t,d \quad \text{st} \quad t \notin \{T1, M\}, ENCD_{s,d} = 1 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1 + 7 \cdot (1 - MW), \\
& && \quad 3 + 7 \cdot (1 - MW)\}, (d+1) \bmod 7 = NCD_s, NNCD_{s,d+1} = 0 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW)\}, \\
& && \quad ENCD_{s,d+1} = 1 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW)\}, \\
& && \quad (d+2) \bmod 7 = NCD_s, NNCD_{s,d+2} = 0, MONCD_{s,d+2} = 0 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW)\}, \\
& && \quad ENCD_{s,d+2} = 1, MONCD_{s,d+2} = 0 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 = 5 + 7 \cdot MW, (d+3) \bmod 7 = NCD_s, \\
& && \quad NNCD_{s,d+3} = 0, MONCD_{s,d+3} = 0 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 = 5 + 7 \cdot MW, ENCD_{s,d+3} = 1, MONCD_{s,d+3} = 0 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW), 5 + 7 \cdot MW\}, \\
& && \quad d \bmod 7 = NCD_s, NNCD_{s,d} = 0, MNCD_{s,d} = 0 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW), 5 + 7 \cdot MW\}, \\
& && \quad ENCD_{s,d} = 1, MNCD_{s,d} = 0 \\
x_{s,t,d} &= 0 && \forall s,t,d \quad \text{st} \quad TS_{s,t} = 0 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{5 + 7 \cdot MW, 1 + 7 \cdot (1 - MW), \\
& && \quad 3 + 7 \cdot (1 - MW)\}, V_{s,d} = 1, MV_{s,d} = 0 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{5 + 7 \cdot MW, 1 + 7 \cdot (1 - MW), \\
& && \quad 3 + 7 \cdot (1 - MW)\}, V_{s,d+1} = 1 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 = 5 + 7 \cdot MW, V_{s,d+2} = 1 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{1 + 7 \cdot (1 - MW), \\
& && \quad 3 + 7 \cdot (1 - MW)\}, V_{s,d+2} = 1, MOV_{s,d+2} = 0 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 = 5 + 7 \cdot MW, V_{s,d+3} = 1, MOV_{s,d+3} = 0 \\
x_{s,t,d} &= 0 && \forall s,t,d \quad \text{st} \quad t \neq M, d \bmod 7 \in \{1,2,3,4,5\}, V_{s,d} = 1 \\
x_{s,T1,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 7 = 1, V_{s,d+1} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \\
x_{s,T1,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 7 = 1, V_{s,d+2} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \\
x_{s,T1,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 7 = 1, V_{s,d+3} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \\
x_{s,T1,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 7 = 1, V_{s,d+4} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \\
x_{s,t,d} &= 0 && \forall s,t,d \quad \text{st} \quad t \in \{T1, T2\}, d \bmod 7 = 6, V_{s,d} + V_{s,d+1} \geq 1 \\
x_{s,t,d} &= 0 && \forall d,t \quad \text{st} \quad t \in \{Td, Ch, Hpb, OC, OR\}, d \bmod 7 \in \{1,2,3,4,5\}, PH_d = 1 \\
x_{s,t,d} &= 0 && \forall s,t,d \quad \text{st} \quad t \in \{T1, T2, M\}, d \bmod 7 \in \{1,2,3,4,5\}, G_{s,d} = 1 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 \in \{5 + 7 \cdot MW, 1 + 7 \cdot (1 - MW), \\
& && \quad 3 + 7 \cdot (1 - MW)\}, G_{s,d+1} = 1 \\
x_{s,M,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 14 = 5 + 7 \cdot MW, G_{s,d+2} = 1 \\
x_{s,T1,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 7 = 1, G_{s,d+1} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \\
x_{s,T1,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 7 = 1, G_{s,d+2} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \\
x_{s,T1,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 7 = 1, G_{s,d+3} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \\
x_{s,T1,d} &= 0 && \forall s,d \quad \text{st} \quad d \bmod 7 = 1, G_{s,d+4} = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0 \\
x_{s,t,d} &= 0 && \forall s,t,d \quad \text{st} \quad t \in \{T1, T2\}, d \bmod 7 = 6, G_{s,d} + G_{s,d+1} \geq 1
\end{aligned}
$$

$$x_{s,M,d} = 0 \qquad \forall s,d \quad \text{st} \quad \sum_t TS_{s,t} = TS_{s,M} + TS_{s,OC} + TS_{s,OR}, X_{s,M,d} = 0$$

$$d_{t,d} = 0 \qquad \forall t,d \quad \text{st} \quad t \in \{T1,T2\}, d \bmod 7 = 0$$

$$d_{T1,d} = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{2,3,4,5\}, COMB_{\lceil \frac{d}{7} \rceil} = 0$$

$$d_{M,d} = 0 \qquad \forall d \quad \text{st} \quad d \bmod 14 \notin \{1+7\cdot(1-MW), 3+7\cdot(1-MW), 5+7\cdot MW\}$$

$$d_{t,d} = 0 \qquad \forall t,d \quad \text{st} \quad t \in \{Td, Ch, Hpb\}, d \bmod 7 \in \{0,6\}$$

$$d_{OC,d} = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{0,2,3,4,5,6\}$$

$$d_{OR,d} = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{0,1,2,6\}$$

$$d_{t,d} = 0 \qquad \forall t,d \quad \text{st} \quad t \in \{Td, Ch, Hpb, OC, OR\}, PH_d = 1$$

$$dm_d = 0 \qquad \forall d \quad \text{st} \quad d \bmod 14 \notin \{5+7\cdot MW, 1+7\cdot(1-MW), 3+7\cdot(1-MW)\}$$

$$lm_w = 0 \qquad \forall w \quad \text{st} \quad w \bmod 2 = 1 - MW$$

$$fd_d = 0 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{0,3\}$$

$$\sum_s (MOD_s + MODC_{s,d}) \cdot x_{s,M,d} \geq 1 - dm_d \qquad \forall d \quad \text{st} \quad d \bmod 14 \in \{7\cdot MW+5, 7\cdot(1-MW)+1, 7\cdot(1-MW)+3\}$$

$$\sum_s (1-JR_s) \cdot (x_{s,T1,d} + x_{s,T2,d}) \geq 1 - ds_d \qquad \forall d \quad \text{st} \quad d \bmod 7 = 6$$

$$\sum_s (1-JR_s) \cdot (x_{s,T1,d-5} + x_{s,T2,d}) \geq 1 - ds_d \qquad \forall d \quad \text{st} \quad d \bmod 7 = 6, COMB_{\lceil \frac{d}{7} \rceil} = 0$$

$$\sum_s (1-JR_s) \cdot (x_{s,T1,d} + x_{s,T2,d+2}) \geq 1 - ds_{d+2} \qquad \forall d \quad \text{st} \quad d \bmod 7 = 6 \quad COMB_{\lceil \frac{d+2}{7} \rceil} = 0$$

$$\sum_s (1-JR_s) \cdot (x_{s,T1,d} + x_{s,T2,d'}) \geq 1 - ds_{d'} \qquad \forall d,d' \quad \text{st} \quad d \bmod 7 = 1, COMB_{\lceil \frac{d}{7} \rceil} = 0, d' \in \{d, d+1, d+2, d+3, d+4\}$$

$$\sum_s (1-JR_s) \cdot (x_{s,T1,d} + x_{s,T2,d}) \geq 1 - ds_d \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5\}, COMB_{\lceil \frac{d}{7} \rceil} = 1$$

$$\sum_s (1-JR_s) \cdot (x_{s,Td,d} + x_{s,T2,d}) \geq 1 - ds_d \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5\}, COMB_{\lceil \frac{d}{7} \rceil} = 0$$

$$\sum_{d'=d}^{d+2} \sum_s HPB_s \cdot x_{s,OR,d'} \geq 1 - dhpb_{\lceil \frac{d}{7} \rceil} \qquad \forall d \quad \text{st} \quad d \bmod 7 = 3$$

$$\sum_s x_{s,t,d} + d_{t,d} = 1 \qquad \forall t,d \quad \text{st} \quad t \in \{Td, Ch, Hpb\}, d \bmod 7 \in \{1,2,3,4,5\},\ PH_d = 0$$

$$\sum_s x_{s,T1,d} + d_{T1,d} = 1 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{1,6\}$$

$$\sum_s x_{s,T1,d} + d_{T1,d} = 1 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{2,3,4,5\}, COMB_{\lceil \frac{d}{7} \rceil} = 1$$

$$\sum_s x_{s,T2,d} + d_{T2,d} = 1 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5,6\}$$

$$\sum_s x_{s,M,d} + d_{M,d} = 2 \qquad \forall d \quad \text{st} \quad d \bmod 14 \in \{5+7\cdot MW, 7\cdot(1-MW)+1, 7\cdot(1-MW)+3\}$$

$$\sum_s x_{s,OC,d} + d_{OC,d} = 2 \qquad \forall d \quad \text{st} \quad d \bmod 7 = 1, PH_d = 0$$

$$\sum_s x_{s,OR,d} + d_{OR,d} = 2 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{3,4,5\}, PH_d = 0$$

$$x_{s,t,d'} \leq 1 - x_{s,T1,d} \qquad \forall s,t,d,d' \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 0, t \neq T1, d \bmod 7 = 1, d' \in \{d, d+1, d+2, d+3, d+4\}$$

$$x_{s,T1,d} \leq 1 - x_{s,T1,d+2} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 0, d \bmod 7 = 6$$

$$x_{s,T2,d} \leq 1 - x_{s,T1,d+2} + wk_{\lceil \frac{d+2}{7} \rceil} \qquad \forall s,d,w \quad \text{st} \quad COMB_{\lceil \frac{d+2}{7} \rceil} = 0, d \bmod 7 = 6$$

$$\sum_{t \in \{T1,T2\}} x_{s,t,d+5} + x_{s,T1,d} \leq 1 \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 0, d \bmod 7 = 1$$

$$\sum_t x_{s,t,d+2} + 5 \cdot x_{s,T1,d} \leq 5 \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d+2}{7} \rceil} = 0, d \bmod 7 = 6$$

$$x_{s,T1,d} + x_{s,T1,d+7} \leq 1 \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 0, COMB_{\lceil \frac{d+7}{7} \rceil} = 0, d \bmod 7 = 1$$

$$x_{s,M,d-1} + x_{s,M,d+6} + x_{s,M,d+13} + \sum_{t \in \{T1,T2\}} x_{s,t,d} + x_{s,t,d+7} + x_{s,t,d+14} \leq 2 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 6$$

$$x_{s,t,d} \leq 1 - x_{s,OC,d} \qquad \forall s,t,d \quad \text{st} \quad d \bmod 7 = 1, t \neq OC$$

$$x_{s,t,d} \leq 1 - x_{s,OR,d} \qquad \forall s,t,d \quad \text{st} \quad d \bmod 7 \in \{3,4,5\}, t \notin \{T2, OR\}$$

$$x_{s,Hpb,d} + x_{s,T2,d} \leq 1 + y_d \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5\}$$

$$x_{s,Hpb,d} + x_{s,M,d} \leq 1 + w_d \qquad \forall s,d \quad \text{st} \quad d \bmod 14 \in \{7 \cdot MW + 5, 7 \cdot (1 - MW) + 1, 7 \cdot (1 - MW) + 3\}$$

$$x_{s,OR,d} + x_{s,T2,d} \leq 1 + z_d \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{3,4,5\}$$

$$x_{s,Ch,d} + x_{s,Hpb,d} \leq 1 + u_d \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5\}$$

$$y_d + z_d \leq 1 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{3,4,5\}$$

$$y_d + w_d + u_d \leq 1 \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5\}$$

$$x_{s,t,d} + x_{s,t',d} \leq 1 + mc_{d,t,t'} \qquad \forall s,t,t',d \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5\}, t, t' \in \{T2, Td, Ch, Hpb\}, t' \prec t, \neg(t \in \{T2, Ch\}, t' = Hpb)$$

$$x_{s,t,d} = X_{s,t,d} \qquad \forall s,t,d \quad \text{st} \quad X_{s,t,d} = 1, d \bmod 7 \neq 0$$

$$x_{s,T2,d+1} \geq x_{s,T2,d} - x_{s,T2,d-2} \qquad \forall d \quad \text{st} \quad d \bmod 7 = 1$$

$$x_{s,T2,d+1} \geq x_{s,T2,d} - x_{s,T2,d-1} \qquad \forall d \quad \text{st} \quad d \bmod 7 \in \{2,3,4,5\}$$

$$\sum_{d'=d}^{d+3} x_{s,T2,d'} \leq 3 + fd_d \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,2\}$$

$$\sum_{d'=d}^{d+3} x_{s,T2,d'} \leq 3 \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 3$$

$$\sum_{d'=d}^{d+2} x_{s,T2,d'} \leq 2 + fd_d \qquad \forall s,d \quad \text{st} \quad d \bmod 7 = 4$$

$$\sum_{d'=d}^{d+3} x_{s,T2,d'} \leq 2 + fd_d \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{5,6\}$$

$$x_{s,T2,d+4} \leq 2 - fd_d - x_{s,T2,d} \qquad \forall s,d \quad \text{st} \quad d \bmod 7 \in \{1,2,4,5,6\}$$

$$x_{s,t,d} \leq 1 - x_{s,M,d} \qquad \forall s,t,d \quad \text{st} \quad t \in \{T2, Td, OR, Ch\}, d \bmod 14 = 5 + 7 \cdot MW$$

$$x_{s,t,d+1} \leq 1 - x_{s,M,d} \qquad \forall s,t,d \quad \text{st} \quad t \in \{T1, T2\}, d \bmod 14 = 5 + 7 \cdot MW$$

$$x_{s,t,d+3} \leq 1 - x_{s,M,d} \qquad \forall s,t,d \quad \text{st} \quad d \bmod 14 = 5 + 7 \cdot MW$$

$$x_{s,t,d'} \leq 1 - x_{s,M,d} \qquad \forall s,t,d,d' \quad \text{st} \quad t \in \{T1, T2, Td, OC, Ch, Hpb\}, d \bmod 14 = 1 + 7 \cdot (1 - MW), d' \in \{d, d+1\}$$

$$x_{s,t,d+2} \leq 1 - x_{s,M,d} \qquad \forall s,t,d \quad \text{st} \quad t \neq M, d \bmod 14 = 1 + 7 \cdot (1 - MW)$$

$$x_{s,M,d+2} \leq 1 - x_{s,M,d} + lm_{\lceil \frac{d}{7} \rceil} \qquad \forall s,d \quad \text{st} \quad d \bmod 14 = 1 + 7 \cdot (1 - MW)$$

$$x_{s,t,d'} \leq 1 - x_{s,M,d} \qquad \forall s,t,d,d' \quad \text{st} \quad t \in \{T2, Td, OR, Ch, Hpb\}, d \bmod 14 = 3 + 7 \cdot (1 - MW), d' \in \{d, d+1, d+2\}$$

$$x_{s,Td,d} = x_{s,T1,d} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 1, PH_d = 0, d \bmod 7 \neq 6$$

$$x_{s,t,d} \leq 1 - x_{s,T1,d} \qquad \forall s,t,d \quad \text{st} \quad t \notin \{T1, Td\}, COMB_{\lceil \frac{d}{7} \rceil} = 1$$

$$x_{s,T1,d+1} \leq 1 - x_{s,T2,d} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d+1}{7} \rceil} = 1, (d+1) \bmod 7 \neq 1$$

$$x_{s,T1,d+2} \leq 1 - x_{s,T2,d} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d+2}{7} \rceil} = 1, (d+2) \bmod 7 = 1$$

$$x_{s,T2,d+1} \leq 1 - x_{s,T1,d} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 1, (d+1) \bmod 7 \neq 1$$

$$x_{s,T2,d+2} \leq 1 - x_{s,T1,d} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 1, (d+2) \bmod 7 = 1$$

$$x_{s,T1,d+1} \geq x_{s,T1,d} - x_{s,T1,d-1} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 1, d \bmod 7 \in \{2,3,4,5\}$$

$$x_{s,T1,d+1} \geq x_{s,T1,d} - x_{s,T1,d-2} \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 1, d \bmod 7 = 1$$

$$\sum_{d'=d}^{d+3} x_{s,T1,d'} \leq 3 \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 1, d \bmod 7 \in \{1,2,3\}$$

$$\sum_{d'=d}^{d+2} x_{s,T1,d'} \leq 2 \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d}{7} \rceil} = 1, d \bmod 7 = 4$$

$$\sum_{d'=d}^{d+3} x_{s,T1,d'} \leq 2 \qquad \forall s,d \quad \text{st} \quad COMB_{\lceil \frac{d+3}{7} \rceil} = 1, d \bmod 7 \in \{5,6\}$$

$$\begin{aligned} 3 \cdot \sum_{d:d \bmod 14 = 5 + 7 \cdot MW} x_{s,M,d} &+ \\ 2 \cdot \sum_{d:d \bmod 14 = 1 + 7 \cdot (1-MW)} x_{s,M,d} &+ \\ 2 \cdot \sum_{d:d \bmod 14 = 3 + 7 \cdot (1-MW)} x_{s,M,d} + DT_{s,M} &\leq mds_M \end{aligned} \qquad \forall s \quad \text{st} \quad \sum_{t \in \{T1,T2,M\}} TS_{t,s} \neq TS_{M,s}$$

$$\begin{aligned} \sum_{d:d \bmod 7 \neq 6} x_{s,T2,d} &+ \\ 2 \cdot \sum_{d:d \bmod 7 = 6} x_{s,T2,d} + DT_{s,T2} &\leq mds_{T2} \end{aligned} \qquad \forall s$$

$$\begin{aligned} 3 \cdot \sum_{d:d \bmod 14 = 5 + 7 \cdot MW} x_{s,M,d} &+ \\ 2 \cdot \sum_{d:d \bmod 14 = 1 + 7 \cdot (1-MW)} x_{s,M,d} &+ \\ 2 \cdot \sum_{d:d \bmod 14 = 3 + 7 \cdot (1-MW)} x_{s,M,d} &+ \\ 2 \cdot \sum_{d:d \bmod 7 = 6} \sum_{t \in \{T1,T2\}} x_{s,t,d} &+ \\ \sum_{d:d \bmod 7 \neq 6} \sum_{t \notin \{M,T1\}} x_{s,t,d} &+ \\ \sum_{d:d \bmod 7 \notin \{1,6\}} x_{s,T1,d} &+ \end{aligned}$$

$$5 \cdot \sum_{d: d \bmod 7=1, COMB_{\lceil \frac{d}{7} \rceil}=0} x_{s,T1,d} +$$
$$\sum_{d: d \bmod 7=1, COMB_{\lceil \frac{d}{7} \rceil}=1} x_{s,T1,d} +$$
$$2 \cdot \sum_{d: d \bmod 7=6} x_{s,T1,d} + \sum_{t} DT_{s,t} \leq md \qquad \forall s$$
$$\sum_{d: d \bmod 7 \notin \{1,6\}} x_{s,T1,d} +$$
$$5 \cdot \sum_{d: d \bmod 7=1, COMB_{\lceil \frac{d}{7} \rceil}=0} x_{s,T1,d} +$$
$$\sum_{d: d \bmod 7=1, COMB_{\lceil \frac{d}{7} \rceil}=1} x_{s,T1,d} +$$
$$2 \cdot \sum_{d: d \bmod 7=6} x_{s,T1,d} + DT_{s,T1} \leq mds_{T1} \qquad \forall s$$
$$\sum_{d} x_{s,OR,d} + DT_{s,OR} \leq mds_{OR} \qquad \forall s$$
$$x_{s,M,d} = X_{s,M,d} \qquad \forall s, d \quad \text{st} \quad \sum_{t} TS_{t,s} = \sum_{t \in \{M,OR,OD\}} TS_{t,s},$$
$$d \bmod 14 \in \{5 + 7 \cdot WM, 1 + 7 \cdot (1 - MW), 3 + 7 \cdot (1 - MW)\}$$
$$3 \cdot \sum_{d: d \bmod 14=5+7 \cdot MW} x_{s,M,d} +$$
$$2 \cdot \sum_{d: d \bmod 14=1+7 \cdot (1-MW)} x_{s,M,d} +$$
$$2 \cdot \sum_{d: d \bmod 14=3+7 \cdot (1-MW)} x_{s,M,d} \leq \frac{|D|}{5} \qquad \forall s \quad \text{st} \quad \sum_{t \in \{T1,T2,M\}} TS_{t,s} = TS_{M,s}$$
$$\sum_{d=7 \cdot (w-1)+1}^{7 \cdot w} x_{s,Hpb,d} \leq 5 \cdot ih_{s,w} \qquad \forall s, w \quad \text{st} \quad TS_{s,Hpb} = 1$$
$$\sum_{d=7 \cdot (w-1)+1}^{7 \cdot w} x_{s,Ch,d} \leq 5 \cdot ic_{s,w} \qquad \forall s, w \quad \text{st} \quad TS_{s,Ch} = 1$$
$$mcd_d \geq mc_{d,t,t'} \qquad \forall d, t, t' \quad \text{st} \quad d \bmod 7 \in \{1,2,3,4,5\}, t \prec t'$$
$$t, t' \in \{T2, Td, Ch, Hpb\}$$
$$\sum_{d: d \bmod 7=1, COMB_{\lceil \frac{d}{7} \rceil}=0} x_{s,T1,d} \leq 1 \qquad \forall s \quad \text{st} \quad FWT1_s = 1$$
$$x_{s,T2,d} + x_{s,T2,d+1} \leq 1 \qquad \forall s, d \quad \text{st} \quad CHRISTMAS_d = 1,$$
$$CHRISTMAS_{d+1} = 1$$
$$x_{s,T1,d} + x_{s,T1,d+1} \leq 1 \qquad \forall s, d \quad \text{st} \quad CHRISTMAS_d = 1,$$
$$CHRISTMAS_{d+1} = 1$$
$$\sum_{t} x_{s,t,1} \leq 5 - 5 \cdot TW_s \qquad \forall s \quad \text{st} \quad COMB_1 = 0$$
$$x_{s,T1,1} \leq 1 - TWB_s \qquad \forall s \quad \text{st} \quad COMB_1 = 0$$
$$x_{s,M,5} + \sum_{t \in \{T1,T2\}} x_{s,t,6} \leq 2 - WB_s - WB_s \cdot WBT_s \qquad \forall s$$
$$x_{s,M,5} + \sum_{t \in \{T1,T2\}} x_{s,t,6} +$$
$$x_{s,M,12} + \sum_{t \in \{T1,T2\}} x_{s,t,13} \leq 2 - WB_s \qquad \forall s$$
$$x_{s,T2,1} = 0 \qquad \forall s \quad \text{st} \quad ND_s = 4$$
$$x_{s,T2,1} \leq fdm \qquad \forall s \quad \text{st} \quad ND_s = 3$$
$$x_{s,T2,2} \leq 1 - x_{s,T2,1} + fdtu \qquad \forall s \quad \text{st} \quad ND_s = 2$$
$$x_{s,T1,1} = 0 \qquad \forall s \quad \text{st} \quad COMB_1 = 1, NDT_s = 3$$

$$
\begin{aligned}
x_{s,T1,2} &\leq 1 - x_{s,T1,1} && \forall s \quad \text{st} \quad COMB_1 = 1, NDT_s = 2 \\
x_{s,T1,1} &\leq 1 - TW_s && \forall s \quad \text{st} \quad COMB_1 = 0 \\
x_{s,T1,1} &\leq wk_1 && \forall s \quad \text{st} \quad COMB_1 = 0, ND_s > 0 \\
x_{s,t,1} &\leq 1 - WBM_S && \forall s, t \quad \text{st} \quad MW = 1 \\
\sum_s (1 - JR_s)(TW_s + x_{s,T2,1}) &\geq 1 - ds_1 && \quad \text{st} \quad COMB_1 = 0
\end{aligned}
$$

Appendix B

Procedure making schedule after solving ILP

When we solve the ILP as described in Chapter 4, we made a schedule, but we scheduled only the starting moments of certain tasks and days. Therefore, we have to execute the procedure shown in Algorithm 2 to make a schedule from the output of the ILP.

```
for all surgeons, tasks and days do
    if the day number corresponds to a Saturday then
        The tasks scheduled on Saturday are also scheduled on Sunday
    else if the day number does not correspond to a Sunday then
        if the task is the organ removal shift then
            if the day number corresponds to a Friday and in the weekend the organ removal shift must
              be scheduled then
                if The organ removal shift is scheduled during this day then
                    The organ removal shift is scheduled during day numbers d, d+1, d+2, d+3
                end
            else if the day number corresponds to a Monday during a week that the organ removal shift
              must be scheduled then
                if The organ removal shift is scheduled during this day then
                    The organ removal shift is scheduled during day numbers d, d+1, d+2
                end
            else if the day number corresponds to a Wednesday during a week that the organ removal
              shift must be scheduled then
                if The organ removal shift is scheduled during this day then
                    The organ removal shift is scheduled during day numbers d, d+1, d+2
                end
            end
        else if the task is the T1 task and the T1 and Td shifts are not combined then
            if the T1 task is scheduled on Monday then
                This task is scheduled for the whole week
            end
        else
            A task is scheduled during the day if the task starts that day for this surgeon
        end
    end
end
for all surgeons and days do
    if the day number corresponds to the non-clinical day of the surgeon and no T1 shift is scheduled
      during this day then
        The surgeon has a non-clinical day during this day
    end
    if the day number corresponds to a day off for the surgeon then
        Schedule the day off for this surgeon
    end
    if the day number corresponds to a day the surgeon does not want to do a shift then
        Indicate that this surgeon does not want to do a shift in the schedule
    end
end
for all surgeons do
    if the surgeon has done the organ removal shift the last weekend before the schedule then
        Schedule the organ removal shift on the first day of the schedule
    end
end
```

Algorithm 2: The algorithm to output the schedule made by the ILP.

Appendix C

Input schedule

In this chapter the input for the case-study is shown. Table C.1 shows the characteristics of the different surgeons. Table C.2 shows the tasks that can be scheduled for the different surgeons. Table C.3 shows the weeks in which the $T1$ and Td shifts are combined. Table C.4 shows the public holidays and Christmas days during the schedule. Table C.5 shows the possibilities to schedule half a day organ removal shift during a day off or non-clinical day. Table C.6 shows which surgeon has done which shift before the start date of the schedule made. Table C.7 shows the number of days a task is scheduled. Figure C.1 shows the days off, non-clinical days, when no shift should be scheduled and the tasks which are already scheduled.

Table C.1: The characteristics of the different surgeons.

Surgeon	Junior	MOD certified	Hpb surgeon	weekend $T1$ preferred	NCD
1	0	0	1	0	5
2	0	1	0	0	4
3	0	1	1	0	3
4	0	1	0	0	5
5	0	1	0	0	2
6	0	1	1	0	3
7	1	0	0	0	2
8	0	1	0	1	3
9	0	0	0	0	4
10	1	0	0	0	2
11	1	0	0	0	3
12	0	1	0	0	-1
13	0	1	0	0	-1

Table C.2: The tasks that can be scheduled for the different surgeons.

Surgeon	$T1$	$T2$	Td	M	Ch	Hpb	OC	OR
1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1
3	0	0	0	1	0	1	0	0
4	1	1	1	1	1	0	1	1
5	1	1	1	1	1	1	1	1
6	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1
9	0	0	1	0	1	1	1	1
10	1	1	1	1	1	1	1	1
11	1	1	1	1	1	1	1	1
12	0	0	0	1	0	0	1	1
13	0	0	0	1	0	0	0	0

Table C.3: The weeks the $T1$ and Td shifts are combined.

Week	$T1, Td$ combined
1	0
2	0
3	0
4	1
5	0
6	0
7	0
8	0
9	0
10	0
11	0
12	0
13	1
14	1

Table C.4: The day numbers corresponding to public holidays during the schedule.

Public holiday	Christmas
Day number 86	Day number 86
Day number 87	Day number 87
Day number 93	

Table C.5: The days the organ removal shift can be scheduled for surgeons during a day off or non-clinical day.

Surgeon	Morning during NCD	Afternoon during NCD	Morning during day off	Afternoon during day off
1				
2				
3	Wednesday			
4				
5				
6				
7				
8				
9				
10				
11	Wednesday			Friday
12				
13				

Table C.6: The value of the boundary parameters.

Surgeon	Boundary parameters						
	Last weekend	Second last weekend	$T1$ week	$T1$ weekend	M weekend	Days $T2$	Days $T1$
1	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0
4	0	1	0	0	0	0	0
5	0	1	0	0	0	0	0
6	0	1	0	0	0	0	0
7	0	0	0	0	0	0	0
8	1	0	0	1	0	0	2
9	0	0	0	0	0	0	0
10	1	0	0	0	0	2	0
11	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0
13	0	1	0	0	0	0	0

Table C.7: The number of days a task should be scheduled

Task	Number of days
$T1$	98
$T2$	98
Td	70
M	98
Ch	70
Hpb	70
OC	14
OR	42

Surgeon\Day	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	V	V	V	V	V					G		NCD				V	V	V	V	V	V
2				NCD							NCD						V	V	V	V	
3		NCD								NCD							NCD	V	V	V	V
4					NCD						V	V	V						NCD		
5	G	NCD							NCD			G	G	G	G	NCD					
6	V	V	V	V	V	V	V				NCD						NCD			G	G
7		NCD							NCD							NCD					V
8	FromT1	NCD								NCD							NCD			V	V
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10		NCD							NCD		V	V	G			NCD			V		
11			NCD	V	V				G	V	V	V	G	G			NCD	V	V		
12	V	V			V							V			V	V	V		V		
13																					

Surgeon\Day	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42
1										NCD									NCD		
2			V	V	V						NCD	V	V	V	V		V	V	V	V	
3	V	V	V	V	V	V	V			NCD							NCD				
4					V	V	V	V	V	V	V	V			V				NCD		
5	V				V	G	G	G	NCD							NCD			G	G	G
6	V	V	V	V	V	V				NCD			G	G	G		NCD			G	G
7	V	V	V						NCD		V	V	V	V	V	V	V	V	V	V	V
8	V	V	V	V	V	V	V			NCD							NCD			G	G
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10								V	NCD							NCD					
11				V	V	G	G			NCD	V	V					NCD	V	V	G	G
12	V	V	V	V	V			V	V	V		V			V		V	V	V		
13																					

Surgeon\Day	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
1			NCD									NCD					NCD				
2				NCD					V	V	V	V				V		NCD			
3			NCD							NCD	V	V	V	V			NCD				
4					NCD							NCD							NCD	V	V
5	G	NCD							NCD			G	G	G	G	NCD					
6	G		NCD			G	G	G		NCD							NCD				
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8			NCD							NCD							NCD				
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10		NCD							NCD			V	V	V		NCD					
11		G	V	V	V	V				NCD	V	V	G	G			NCD	V	V		
12	V	V	V		V			V		V	V	V			V	V	V		V		
13																					

Surgeon\Day	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84
1					NCD					NCD									NCD		
2			V	V	V						NCD							V	V	V	V
3			NCD							NCD							NCD			V	V
4	V	V	V	V	V	V	V					NCD							NCD	V	V
5		NCD			G	G	G	G	NCD							NCD			G	G	G
6			NCD							NCD							NCD			V	V
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8			NCD							NCD							NCD			V	V
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10		NCD	V						NCD							NCD				G	G
11			NCD	V	V	G	G			NCD	V	V					NCD	V	V	G	G
12	V		V	V	V			V	V	V		V				V			V		
13																					

Surgeon\Day	85	86	87	88	89	90	91	92	93	94	95	96	97	98
1	V	V	V	V	V	V	V					NCD		
2	V	V	V	V	V	V	V				NCD			
3	V	V	V	V	V	V	V			NCD				
4	V	V	V	V	V							NCD		
5	G							V	V	V	V	V	V	V
6	V	V	V	V	V	V	V			NCD				
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	V	V	V	V	V	V	V			NCD				
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10								V	V	V	V	V	V	V
11				V	V			V	V	V	V	V	V	V
12	V	V	V		V									
13														

Figure C.1: The input of the days off, days without shifts and already scheduled tasks.

Appendix D

Schedules with different organ removal shift structure

In this appendix the schedules made with the different organ removal shift structures are shown. Section D.1 shows the quarter schedule when the LUMC coorperates with Erasmus MC, Section D.2 shows the quarter schedule when the organ removal shift is scheduled as was done before the current structure was introduced and Section D.3 shows the quarter schedule when the organ removal shift has the same structure as the $T2$ shift.

D.1 Cooperate with Erasmus MC

Figure D.1 shows the quarter schedule and Figure D.2 shows the shortages at the different tasks when the LUMC cooperates with the Erasmus MC.

Surgeon\Day	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	V	V	V	V	V					G/OR	Td	NCD			OC	V	V	V	V	V	V
2	T2	T2/Hpb	Td	NCD	Ch			Ch	Ch	Ch	NCD	Ch	t1	t1	FromT1	Td	V	V	V	V	
3	Hpb	NCD	Hpb	Hpb	Hpb			Hpb	Hpb	NCD	Hpb	M/Hpb	M	M	M	Hpb	NCD	V	V	V	V
4	Ch	Td/Ch	Ch	Ch	NCD	T2	T2	OC		Td	V	V	V		t1	t1	t1	t1	t1		
5	G/Td	NCD	M	M	M	t1	t1	FromT1	NCD	M	M	G/M	G	G	G/Hpb	NCD	T2/Hpb	T2/Hpb	T2/OR	t1	t1
6	V	V	V	V	V	V	V	t1	t1	t1	t1	t1			T2	T2	NCD	or	Td	G	G
7	t1	t1	t1	t1	t1			Td	NCD	T2/Hpb	T2/OR	T2/OR			Td	NCD	M	M	M		V
8	FromT1	NCD	T2	T2/OR	T2/OR			T2	T2	NCD	Ch	Td	T2	T2	Ch	Ch	NCD	Td	Ch/Hpb	V	V
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	OC	NCD	or	Td	Td			OC	NCD	or	V	V	G		OC	NCD	Td/Ch	Ch	V	T2	T2
11	M	M	NCD/M	V	V/M	M	M	M	G/Td	V	V	V	G	G	M	M	NCD/M	V	V/M	M	M
12	V	V	or	or	V			M	M	M	or	V			V	V	V	or	V		
13																					

Surgeon\Day	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42
1	Ch	Ch	or	T1/Td	T1/Td			T2	T2	NCD	Td	T2/OR	T2	T2	T2/Hpb	Hpb	Ch/Hpb	Ch	NCD	T2	T2
2	T2/Hpb	T2	V	V	V	T2	T2	Td	Ch	Ch/Hpb	NCD	V	V	V	V	Td	V	V	V	V	
3	V	V	V	V	V	V	V	Hpb	Hpb	NCD	Hpb	Hpb			M	M	NCD/M	Hpb	Hpb		
4	T1/Td	T1/Td	T1/Td	or	V	V	V	V	V	V	V	V	t1	t1	V	T2	T2/OR	or	NCD	t1	t1
5	V		T2/OR	T2/OR	V	G	G	G/Pali	NCD	T2/OR	T2/OR	Td			Td	NCD	M	M	G/M	G	G
6	V	V	V	V	V	V		t1	t1	t1	t1	t1	G	G	G/Ch	Ch	NCD	T2/OR	T2/Td/Ch	G	G
7	V	V	V	Ch/Hpb	M/Hpb	M	M	M	NCD	Td	V	V	V	V	V	V	V	V	V	V	V
8	V	V	V	V	V	V	V	Ch	Td	NCD	Ch	Ch			t1	t1	t1	t1	t1	G	G
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	M	M	M	M	M	t1	t1	V	NCD	M	M	M			OC	NCD	Td	Td	M	M	M
11	M	Hpb	Ch/Hpb	V	V	G	G	M	M	NCD/M	V	V/M	M	M	M		NCD	V	V	G	G
12	V	V	V	V	V			V	V	V	or	V			V		V	V	V		
13																					

Surgeon\Day	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
1	t1	t1	t1	t1	t1			T2	T2	Td/Ch	or	NCD			t1	t1	t1	t1	t1		
2	Ch	Hpb	Ch/Hpb	NCD	Ch	t1	t1	FromT1	V	V	V	V	t1	t1	FromT1	V	Td	NCD	Td		
3	M	M	NCD/M	Hpb	Hpb			Hpb	Hpb	NCD	V	V	V	V	Hpb	Hpb	NCD	Hpb	Hpb		
4	FromT1	Ch	Td	Ch	NCD			t1	t1	t1	t1	t1			T2	T2	T2/OR	or	NCD	V	V
5	G/Pali	NCD	T2/OR	T2	T2/OR			OC	NCD	M	M	G/M	G	G	G/Td	NCD	M	M	M	T2	T2
6	G/Td		NCD	or	Td	G	G	G/Ch	Ch	NCD	Ch/Hpb	T2/Td/Ch	T2	T2	OC	Ch	NCD	Td	or		
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	T2/Hpb	T2	NCD	Td	M	M	M	M	Td	NCD	Td	M/Hpb	M	M	M	Td	NCD	T2	T2/OR		
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	M	NCD	M	M	M	T2	T2	Td	NCD	T2/Hpb	T2/OR	V	V	V	Ch	NCD	Ch/Hpb	Ch	Ch	t1	t1
11	OC	G/Td	V	V	V	V		M	M	NCD/M	V	V	G	G	M	M	NCD/M	V	V/M	M	M
12	V	V	V	or	V			V		V	V	V			V	V	V	or	V		
13																					

Surgeon\Day	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84
1	OC	Hpb	T2/Td/Hp	T2/OR	NCD			t1	t1	t1	t1	t1			Td		Td	or	NCD	t1	t1
2	T2/Hpb	T2	V	V	V			T2/Hpb	T2	Ch/Hpb	NCD	Ch			Hpb	Hpb	Ch/Hpb	V	V	V	V
3	M	M	NCD/M	Hpb	M/Hpb	M	M	M	Hpb	NCD	Hpb	Hpb			M	M	NCD/M	Hpb	Hpb	V	V
4	V	V	V	V	V	V	V	Td	Td	T2/OR	T2	NCD			t1	t1	t1	t1	t1	V	V
5	Td	NCD	M	M	G/M	G	G	G/Pali	NCD	Td	Td	Td	t1	t1	FromT1	NCD	M	M	G/M	G	G
6	t1	t1	t1	t1	t1			Ch	Ch	NCD	Ch	T2/OR	T2	T2	Ch	Ch	NCD	Ch	Ch	V	V
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	Ch	Ch	NCD	Ch	Td/Ch	t1	t1	FromT1		NCD	or	or			T2	T2	NCD	Td	Td	V	V
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	FromT1	NCD	V	Td	T2/OR	T2	T2	OC	NCD	M	M	M			OC	NCD	T2/OR	T2	T2/OR	G	G
11	M	Td	NCD	V	V	G	G	M	M	NCD/M	V	V/M	M	M	M	Td	NCD	V	V	G	G
12	V		V	V	V			V	V	V	or	V			OC	V	or	or	V		
13																					

Surgeon\Day	85	86	87	88	89	90	91	92	93	94	95	96	97	98
1	V	V	V	V	V	V	V	Hpb		Ch/Hpb	or	NCD		
2	V	V	V	V	V	V	V	T1/Td	t1	T1/Td	NCD	T2/OR	T2	T2
3	V	V	V	V	V	V	V	M	M	NCD/M	Hpb	M/Hpb	M	M
4	V	V	V	V	V	T2	T2	OC		T2/OR	T2	NCD		
5	G/Ch/Hpb		t1	T1/Td	T1/Td			V	V	V	V	V	V	V
6	V	V	V	V	V	V	V	T2	T2	NCD	T1/Td	T1/Td		
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	V	V	V	V	V	V	V	Ch		NCD	Ch	Ch	t1	t1
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	T1/Td	t1	M	M	M			V	V	V	V	V	V	V
11	M	M	M	V	V	t1	t1	V	V	V	V	V	V	V
12	V	V	V	or	V			OC		or	or	or		
13														

Figure D.1: The generated schedule of the different tasks during quarter 4 when the LUMC cooperates with the Erasmus MC.

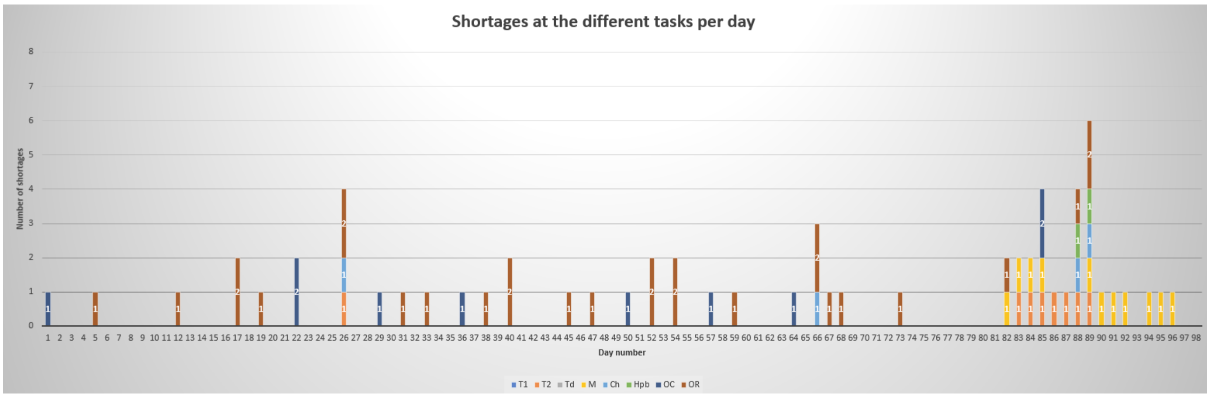

Figure D.2: The shortages at the different tasks in the quarter schedule per day when the LUMC cooperates with the Erasmus MC.

D.2 Old situation

Figure D.3 shows the quarter schedule and Figure D.4 shows the shortages at the different tasks when the organ removal shift is scheduled using the old structure.

Surgeon\Day	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	V	V	V	V	V			OC		G/OR	Td	NCD			Ch	V	V	V	V	V	V
2	T2	T2/Hpb	or	NCD	Ch	M	M	T2	T2	T2/OR	NCD	M	T2	T2	T2	Td	V	V	V	V	
3	Hpb	NCD	Hpb	Hpb	M/Hpb			Hpb	Hpb	NCD	M	M			Hpb	Hpb	NCD	V	V	V	V
4	Ch	Ch	or	or	NCD	T2	T2	Td	Ch	M	V	V	V		Td	T2	T2/OR	Ch	NCD	M	M
5	G/Td	NCD	T2	T2	T2/OR			Ch	NCD	Ch	or	G/Td	G	G	G	NCD	Ch/Hpb	Hpb	M/Hpb	T2	T2
6	V	V	V	V	V	V	V	t1	t1	t1	t1	t1			OC		NCD	or	M	G	G
7	t1	t1	t1	t1	t1			M	NCD	Hpb	Ch/Hpb	Ch/Hpb			t1	t1	t1	t1	t1		V
8	FromT1	NCD	Ch	Td	M			OC	M	NCD	T2/OR	T2/OR	t1	t1	FromT1		NCD	T2	T2/Td/Ch	V	V
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	OC	NCD	Td	Ch	Td				NCD	Td	V	V	G			NCD	Td	Td	V	M	M
11	OC	Td	NCD	V	V	t1	t1	FromT1	G/Td	V	V	V	G	G	OC	Ch	NCD	V	V	t1	t1
12	V	V		or	V	M	M	M	M	M	M	V			V	V	V	or	V		
13																					

Surgeon\Day	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42
1	M	Hpb	Ch/Hpb	T2/OR	T2/Hpb			T2	T2	NCD	or	Td	M	M	T2/Hpb	T2	T2/Hpb	Ch/Hpb	NCD		
2	T1/Td	T1/Td	V	V	V	T2	T2	OC		Ch/Hpb	NCD	V	V	V	V	M	V	V	V	V	
3	V	V	V	V	V	V	V	Hpb	Hpb	NCD	Hpb	M/Hpb	M	M	M	M	NCD	M	M		
4	M	M	T1/Td	T1/Td	V	V	V	V	V	V	V	V	t1	t1	V	Ch	Td/Ch	M	NCD	T2	T2
5	V	Ch	M	M	V	G	G	G	NCD	Td	Td	Ch			Ch	NCD	M	or	Td/Ch	G	G
6	V	V	V	V	V	V		t1	t1	t1	t1	t1	G	G	G/Td	Hpb	NCD	T2/OR	T2/Hpb	G	G
7	V	V	V	M	M			OC	NCD	or	V	V	V	V	V	V	V	V	V	V	V
8	V	V	V	V	V	V	V	Td	Ch	NCD	Ch	M			t1	t1	t1	t1	t1	G	G
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	T2/Hpb	T2	T2/OR	Ch/Hpb	T1/Td	t1	t1	V	NCD	T2/OR	T2	T2/OR			OC	NCD	M	Td	M	t1	t1
11	Ch	M	M	V	V	G	G	Ch	Td	NCD	V	V	T2	T2	M	Td	NCD	V	V	G	G
12	V	V	V	V	V			V	V	V	or	V			V		V	V	V		
13																					

Surgeon\Day	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
1	t1	t1	t1	t1	t1			Ch	T2/Hpb	T2/Td/Hp	T2/OR	NCD			t1	t1	t1	t1	t1		
2	Ch		Ch/Hpb	NCD	Td	M	M	Td	V	V	V	V			OC	V	Ch/Hpb	NCD	or	M	M
3	Hpb	Hpb	NCD	Hpb	M/Hpb			M	M	NCD	V	V	V	V	Hpb	Hpb	NCD	Hpb	M/Hpb		
4	OC		or	Ch	NCD			t1	t1	t1	t1	t1			OC	T2	T2/OR	or	NCD	V	V
5	G	NCD	Td	Td	or	T2	T2	T2/Hpb	NCD	M	Ch/Hpb	T2/Ch/H	G	G	G/Td	NCD	Td	Ch	Ch	T2	T2
6	G/Poli	Td	NCD	or	M	G	G	G/Poli	Ch	NCD	M	M	T2	T2	T2	Td	NCD	T2	T2/OR		
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	T2	T2	NCD	or	Ch	M	M	M	Td	NCD	M	M	t1	t1	FromT1		NCD	or	M		
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	FromT1	NCD	T2/OR	T2	T2/OR	t1	t1	FromT1	NCD	M	Td	V	V	V		NCD	or	Td	Td	t1	t1
11	Td	G/Ch	V	V	V	V		OC	M	NCD	V	V	G	G	Ch	Ch	NCD	V	V	M	M
12	V	V	V		V			V		V	V	V			V	V	V		V		
13																					

Surgeon\Day	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84
1	M	Td	T2/Td/Hp	T2/OR	NCD			t1	t1	t1	t1	t1			OC	M	M	or	NCD	t1	t1
2	Td	Ch	V	V	V	t1	t1	FromT1	Td	Ch/Hpb	NCD	or			Td	T2	T2/OR	V	V	V	V
3	M	M	NCD	M	M			Hpb	Hpb	NCD	Hpb	M/Hpb			Hpb	Hpb	NCD	Hpb	M	V	V
4	V	V	V	V	V	V	V	T2	T2	T2/OR	Ch	NCD			t1	t1	t1	t1	t1	V	V
5	Ch	NCD	M	Td	Td/Ch	G	G	G	NCD	or	or	M			Ch	NCD	Td	Ch	G/Td	G	G
6	t1	t1	t1	t1	t1			OC	Ch	NCD	T2	T2/OR	M	M	M	M	NCD	M	M	V	V
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	T2/Hpb	T2/Hpb	NCD	Ch/Hpb	T2/Hpb	T2	T2	Td		NCD	Td	Td	t1	t1	FromT1	Ch	NCD	T2/OR	T2/OR	V	V
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	FromT1	NCD	V	M	M			Ch	NCD	Td	or	Ch			OC	NCD	Ch/Hpb	Td	Ch/Hpb	G	G
11	OC	M	NCD	V	V	G	G	OC		NCD	V	V	T2	T2	T2	Td	NCD	V	V	G	G
12	V		V	V	V			V	V	V		V	M	M	M	V	M	M	V		
13																					

Surgeon\Day	85	86	87	88	89	90	91	92	93	94	95	96	97	98
1	V	V	V	V	V	V	V	M		M	Ch/Hpb	NCD		
2	V	V	V	V	V	V	V	T1/Td	t1	T1/Td	NCD	M	t1	t1
3	V	V	V	V	V	V	V	M	M	NCD	M	M		
4	V	V	V	V	V	t1	t1	OC		M	M	NCD	T2	T2
5	G/Ch		t1	T1/Td	T1/Td	M	M	V	V	V	V	V	V	V
6	V	V	V	V	V	V	V	T2/Hpb	T2	NCD	T2/OR	T2/Hpb		
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	V	V	V	V	V	V	V	Ch	M	NCD	T1/Td	T1/Td		
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	T2/Hpb	T2		T2/Hpb	T2/Hpb	M	M	V	V	V	V	V	V	V
11	T1/Td	t1		V	V	T2	T2	V	V	V	V	V	V	V
12	V	V	V	or	V			OC		or	or	or		
13														

Figure D.3: The generated schedule of the different tasks during quarter 4 when the organ removal shift is scheduled using the old structure

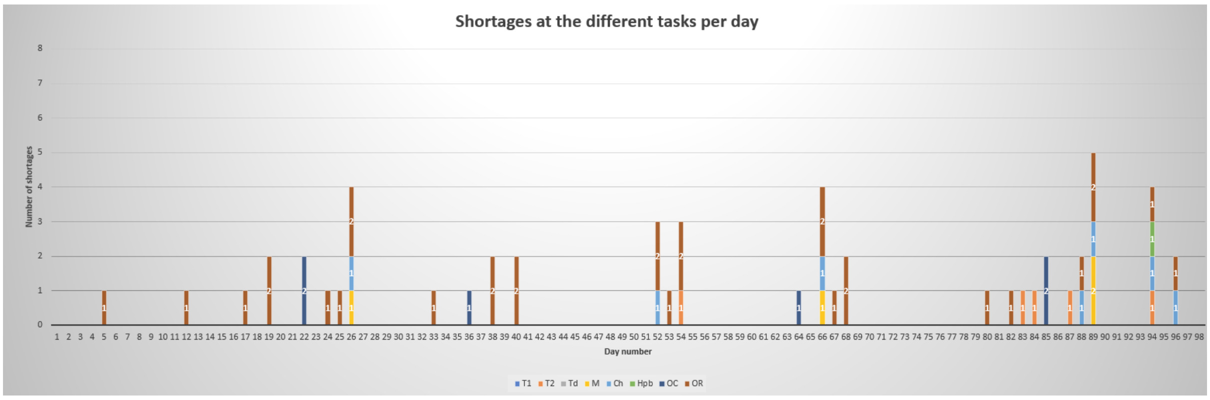

Figure D.4: The shortages at the different tasks in the quarter schedule per day when the organ removal shift is scheduled using the old structure.

D.3 Same structure as the T2 shift

Figure D.5 shows the quarter schedule and Figure D.6 shows the shortages at the different tasks when the organ removal shift is scheduled with the same structure as the $T2$ shift.

Surgeon\Day	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	V	V	V	V	V			T2	T2	G/Td	or	NCD	T2	T2	OC	V	V	V	V	V	V
2	T2	T2/Hpb	or	NCD	M	M	M	Td		or	NCD	or	t1	t1	FromT1		V	V	V	V	
3	Hpb	NCD	Hpb	Hpb	M/Hpb	M	M	Hpb	Hpb	NCD	M	M			Hpb	Hpb	NCD	V	V	V	V
4	OC	Td	or	or	NCD			M	M	M	V	V	V		T2	T2	T2/OR	or	NCD	T2	T2
5	G/Td	NCD	T2	T2	Td				NCD	Hpb	Ch/Hpb	G/Ch/Hp	G	G	G	NCD	Ch/Hpb	Hpb	M/Hpb	M	M
6	V	V	V	V	V	V	V	t1	t1	t1	t1	t1			Ch		NCD	T2	T2/OR	G	G
7	t1	t1	t1	t1	t1			OC	NCD	T2/OR	T2/OR	T2/OR			t1	t1	t1	t1	t1		V
8	FromT1	NCD	Ch	Td	T2/OR	T2	T2	OC	Ch	NCD	Td	Td			Td	Td	NCD	Ch	Td/Ch	V	V
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	OC	NCD	Td	Ch	Ch			Ch	NCD	Ch	V	V	G			NCD	Td	Td	V	t1	t1
11	Ch	Ch	NCD	V	V	t1	t1	FromT1	G/Td	V	V	V	G	G	OC	Ch	NCD	V	V	M	M
12	V	V		or	V			M	M	M	M	V			V	V	V	or	V		
13																					

Surgeon\Day	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42
1	T2/Hpb	T2	T2/OR	Ch/Hpb	T1/Td	t1	t1	FromT1	Ch	NCD	or	Ch			M	M	M	or	NCD	t1	t1
2	M	M	V	V	V			T2	T2	T2/OR	NCD	V	V	V	V	Ch	V	V	V	V	
3	V	V	V	V	V	V	V	Hpb	Hpb	NCD	Hpb	M/Hpb	M	M	M	Hpb	NCD	M	M		
4	Ch	M	M	M	V	V	V	V	V	V	V	V	t1	t1	V	M	M	Td	NCD		
5	V	T1/Td	T1/Td	T1/Td	V	G	G	G/Pali	NCD	Td	Td	M	M	M	Td	NCD	T2/Hpb	T2/OR	Td/Ch	G	G
6	V	V	V	V	V	V		t1	t1	t1	t1	t1	G	G	G/Pali	Td	NCD	M	M	G	G
7	V	V	V	T2/OR	T2/Hpb			Ch	NCD	Ch/Hpb	V	V	V	V	V	V	V	V	V	V	V
8	V	V	V	V	V	V	V	Td	Td	NCD	T2	T2/OR			t1	t1	t1	t1	t1	G	G
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	T1/Td	Ch	M	M	M	T2	T2	V	NCD	or	Ch	Td			Ch	NCD	Td/Ch	Ch/Hpb	T2/Hpb	T2	T2
11	M	Hpb	Ch/Hpb	V	V	G	G	OC		NCD	V	V	T2	T2	T2/Hpb	T2	NCD	V	V	G	G
12	V	V	V	V	V			V	V	V	or	V			V		V	V	V		
13																					

Surgeon\Day	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
1	FromT1		NCD	Ch	Ch			t1	t1	t1	t1	t1			OC		NCD	Td	T2/OR	T2	T2
2	T2	T2	T2/OR	NCD	Td	T2	T2	Td	V	V	V	V			Td	V	Ch/Hpb	NCD	M	M	M
3	Hpb	Hpb	NCD	Hpb	Hpb			M	M	NCD	V	V	V	V	Hpb	Hpb	NCD	Hpb	Hpb		
4	t1	t1	t1	t1	t1			Ch	M	M	Td	NCD			t1	t1	t1	t1	t1	V	V
5	G	NCD	Td	Td	or	t1	t1	FromT1	NCD	T2/T1/Hp	T2/OR	T2/Ch/H	G	G	G	NCD	Td	or	Td	t1	t1
6	G/Pali	Ch	NCD	T2	T2/OR	G	G	G/Pali	Td	NCD	Ch/Hpb	T2/OR	T2	T2	Ch	Ch	NCD	Ch	or		
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	Td		NCD	or	M	M	M	M	Ch	NCD	M	M	t1	t1	FromT1	Td	NCD	or	Ch		
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	Ch	NCD	Ch/Hpb	or	M	M	M	OC	NCD	M	M	V	V	V	OC	NCD	T2/OR	T2	M	M	M
11	OC	G/Td	V	V	V	V		T2/Hpb	T2/Hpb	NCD	V	V	G	G	T2	T2	NCD	V	V		
12	V	V	V		V			V		V	V	V			V	V	V		V		
13																					

Surgeon\Day	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84
1	OC	Ch	T2/Td/Hp	T2/OR	NCD			t1	t1	t1	t1	t1			M	M	M	or	NCD	t1	t1
2	Td	Td	V	V	V			OC	Td	Ch/Hpb	NCD	Td	t1	t1	FromT1	T2	T2/OR	V	V	V	V
3	M	M	NCD	M	M			Hpb	Hpb	NCD	Hpb	M/Hpb	M	M	Hpb	Hpb	NCD	Hpb	Hpb	V	V
4	V	V	V	V	V	V	V	T2	T2	T2/OR	or	NCD			t1	t1	t1	t1	t1	V	V
5	FromT1	NCD	M	M	G/Td	G	G	G	NCD	Td	Ch	or			Td	NCD	Td	Ch	G/Td	G	G
6	t1	t1	t1	t1	t1			Td		NCD	or	M	M	M	Ch	Ch	NCD	M	M	V	V
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	T2/Hpb	T2/Hpb	NCD	Ch/Hpb	Ch/Hpb	t1	t1	FromT1		NCD	T2	T2/OR			M	M	NCD	T2/OR	T2/OR	V	V
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	Ch	NCD	V	Td	T2/OR	T2	T2	Ch	NCD	or	Td	Ch			OC	NCD	Ch/Hpb	Td	Ch	G	G
11	M	M	NCD	V	V	G	G	OC	Ch	NCD	V	V	T2	T2	T2	Td	NCD	V	V	G	G
12	V		V	V	V			V	V	V		V			OC	V	M	M	V		
13																					

Surgeon\Day	85	86	87	88	89	90	91	92	93	94	95	96	97	98
1	V	V	V	V	V	V	V	M	M	M	Ch/Hpb	NCD		
2	V	V	V	V	V	V	V	M	M	M	NCD	T1/Td	t1	t1
3	V	V	V	V	V	V	V	Hpb		NCD	M	M		
4	V	V	V	V	V	T2	T2	Ch		T1/Td	T1/Td	NCD	T2	T2
5	G/Ch		T2	T2/OR	M/Hpb	M	M	V	V	V	V	V	V	V
6	V	V	V	V	V	V	V	T2	T2	NCD	T2/OR	T2/Hpb		
7	V	V	V	V	V	V	V	V	V	V	V	V	V	V
8	V	V	V	V	V	V	V	T1/Td	t1	NCD	M	M		
9	V	V	V	V	V	V	V	V	V	V	V	V	V	V
10	T2/Hpb	T2		Ch/Hpb	T1/Td	t1	t1	V	V	V	V	V	V	V
11	T1/Td	t1		V	V	M	M	V	V	V	V	V	V	V
12	V	V	V	or	V			OC		or	or	or		
13														

Figure D.5: The generated schedule of the different tasks during quarter 4 when the organ removal shift is scheduled with the same structure as the $T2$ shift.

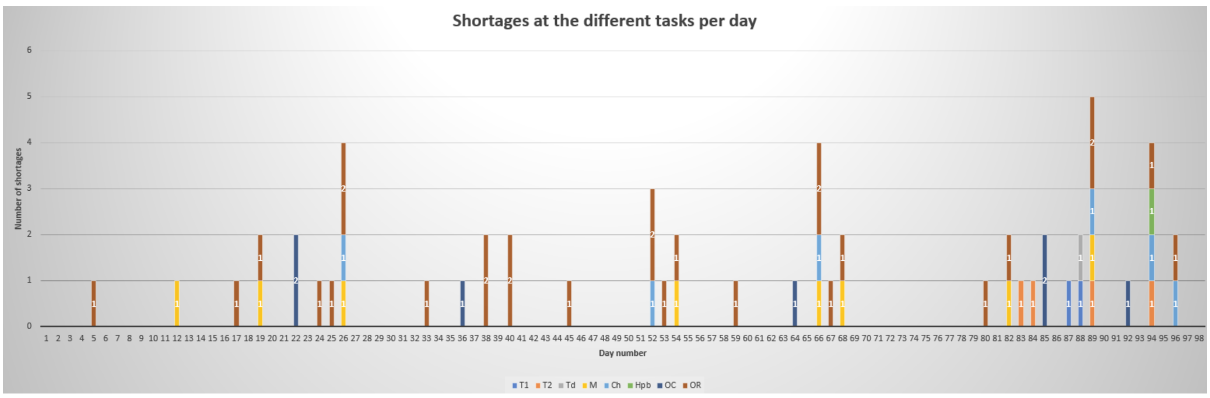

Figure D.6: The shortages at the different tasks in the quarter schedule per day when the organ removal shift is scheduled with the same structure as the $T2$ shift.

Appendix E

Output SDP

In this appendix the output of the SDP for different instances is shown. In Section E.1 results are shown with the original state space and in Section E.2 results are shown for a bigger instance and states lexicographically ordered.

E.1 Original states

The results from the DP with three persons who can be scheduled over three tasks with one time block and when two schedules are made, are shown in Table E.1. This table shows an optimal action given the state and decision epoch. A 1 corresponds to task one with λ_1, a 2 corresponds to task two with λ_2 and a 3 corresponds to task three with λ_3. When '$-$' is in the table, the corresponding state is not part of the state space at the corresponding decision epoch. The parameters used in this case are shown below.

$$
\begin{aligned}
\lambda_1 &= 0.01 \\
\lambda_2 &= 2.02 \\
\lambda_3 &= 1.00 \\
q &= 0.01 \\
N &= 3 \\
J &= 3 \\
M &= 3 \\
B &= 1
\end{aligned}
$$

Table E.1: The DP results for a small instance.

State	$n=1$			$n=2$		
	$p=1$	$p=2$	$p=3$	$p=1$	$p=2$	$p=3$
$(0,0,0)$	3	2	1	2	3	1
$(0,0,1)$	-	-	-	2	3	1
$(0,0,2)$	-	-	-	3	2	1
$(0,0,3)$	-	-	-	3	2	1
$(0,0,4)$	-	-	-	3	2	1
$(0,0,5)$	-	-	-	2	3	1
$(0,0,6)$	-	-	-	3	2	1
$(0,1,0)$	-	-	-	2	1	3
$(0,1,1)$	-	-	-	2	3	1
$(0,1,2)$	-	-	-	2	3	1
$(0,1,3)$	-	-	-	2	3	1
$(0,1,4)$	-	-	-	2	3	1
$(0,1,5)$	-	-	-	2	3	1
$(0,1,6)$	-	-	-	2	3	1
$(0,2,0)$	-	-	-	3	1	2
$(0,2,1)$	-	-	-	2	1	3
$(0,2,2)$	-	-	-	2	3	1

$(0,2,3)$	-	-	-	2	3	1
$(0,2,4)$	-	-	-	2	3	1
$(0,2,5)$	-	-	-	2	3	1
$(0,2,6)$	-	-	-	2	3	1
$(0,3,0)$	-	-	-	3	1	2
$(0,3,1)$	-	-	-	2	1	3
$(0,3,2)$	-	-	-	2	1	3
$(0,3,3)$	-	-	-	2	3	1
$(0,3,4)$	-	-	-	2	3	1
$(0,3,5)$	-	-	-	2	3	1
$(0,3,6)$	-	-	-	2	3	1
$(0,4,0)$	-	-	-	3	1	2
$(0,4,1)$	-	-	-	2	1	3
$(0,4,2)$	-	-	-	2	1	3
$(0,4,3)$	-	-	-	2	1	3
$(0,4,4)$	-	-	-	2	1	3
$(0,4,5)$	-	-	-	2	3	1
$(0,4,6)$	-	-	-	2	3	1
$(0,5,0)$	-	-	-	2	1	3
$(0,5,1)$	-	-	-	2	1	3
$(0,5,2)$	-	-	-	2	1	3
$(0,5,3)$	-	-	-	2	1	3
$(0,5,4)$	-	-	-	2	1	3
$(0,5,5)$	-	-	-	2	1	3
$(0,5,6)$	-	-	-	2	3	1
$(0,6,0)$	-	-	-	3	1	2
$(0,6,1)$	-	-	-	2	1	3
$(0,6,2)$	-	-	-	2	1	3
$(0,6,3)$	-	-	-	2	1	3
$(0,6,4)$	-	-	-	2	1	3
$(0,6,5)$	-	-	-	2	3	1
$(0,6,6)$	-	-	-	2	3	1
$(1,0,0)$	-	-	-	1	3	2
$(1,0,1)$	-	-	-	3	2	1
$(1,0,2)$	-	-	-	3	2	1
$(1,0,3)$	-	-	-	3	2	1
$(1,0,4)$	-	-	-	3	2	1
$(1,0,5)$	-	-	-	3	2	1
$(1,0,6)$	-	-	-	3	2	1
$(1,1,0)$	-	-	-	3	1	2
$(1,1,1)$	-	-	-	3	1	2
$(1,1,2)$	-	-	-	2	3	1
$(1,1,3)$	-	-	-	3	2	1
$(1,1,4)$	-	-	-	3	2	1
$(1,1,5)$	-	-	-	2	3	1
$(1,1,6)$	-	-	-	2	3	1
$(1,2,0)$	-	-	-	3	1	2
$(1,2,1)$	-	-	-	2	1	3
$(1,2,2)$	-	-	-	2	3	1
$(1,2,3)$	-	-	-	2	3	1
$(1,2,4)$	-	-	-	2	3	1
$(1,2,5)$	-	-	-	2	3	1
$(1,2,6)$	-	-	-	2	3	1
$(1,3,0)$	-	-	-	3	1	2
$(1,3,1)$	-	-	-	3	1	2
$(1,3,2)$	-	-	-	2	1	3
$(1,3,3)$	-	-	-	2	3	1
$(1,3,4)$	-	-	-	2	3	1
$(1,3,5)$	-	-	-	2	3	1
$(1,3,6)$	-	-	-	2	3	1
$(1,4,0)$	-	-	-	3	1	2
$(1,4,1)$	-	-	-	3	1	2
$(1,4,2)$	-	-	-	2	1	3
$(1,4,3)$	-	-	-	2	1	3

$(1,4,4)$	-	-	-	2	3	1
$(1,4,5)$	-	-	-	2	3	1
$(1,4,6)$	-	-	-	2	3	1
$(1,5,0)$	-	-	-	3	1	2
$(1,5,1)$	-	-	-	2	1	3
$(1,5,2)$	-	-	-	2	1	3
$(1,5,3)$	-	-	-	2	1	3
$(1,5,4)$	-	-	-	2	1	3
$(1,5,5)$	-	-	-	2	1	3
$(1,5,6)$	-	-	-	2	3	1
$(1,6,0)$	-	-	-	3	1	2
$(1,6,1)$	-	-	-	2	1	3
$(1,6,2)$	-	-	-	2	1	3
$(1,6,3)$	-	-	-	2	1	3
$(1,6,4)$	-	-	-	2	1	3
$(1,6,5)$	-	-	-	2	1	3
$(1,6,6)$	-	-	-	2	1	3
$(2,0,0)$	-	-	-	1	3	2
$(2,0,1)$	-	-	-	1	2	3
$(2,0,2)$	-	-	-	1	2	3
$(2,0,3)$	-	-	-	3	2	1
$(2,0,4)$	-	-	-	3	2	1
$(2,0,5)$	-	-	-	3	2	1
$(2,0,6)$	-	-	-	3	2	1
$(2,1,0)$	-	-	-	1	3	2
$(2,1,1)$	-	-	-	1	2	3
$(2,1,2)$	-	-	-	3	2	1
$(2,1,3)$	-	-	-	3	2	1
$(2,1,4)$	-	-	-	3	2	1
$(2,1,5)$	-	-	-	3	2	1
$(2,1,6)$	-	-	-	3	2	1
$(2,2,0)$	-	-	-	1	3	2
$(2,2,1)$	-	-	-	3	1	2
$(2,2,2)$	-	-	-	3	2	1
$(2,2,3)$	-	-	-	2	3	1
$(2,2,4)$	-	-	-	3	2	1
$(2,2,5)$	-	-	-	3	2	1
$(2,2,6)$	-	-	-	3	2	1
$(2,3,0)$	-	-	-	3	1	2
$(2,3,1)$	-	-	-	3	1	2
$(2,3,2)$	-	-	-	2	1	3
$(2,3,3)$	-	-	-	2	3	1
$(2,3,4)$	-	-	-	2	3	1
$(2,3,5)$	-	-	-	2	3	1
$(2,3,6)$	-	-	-	2	3	1
$(2,4,0)$	-	-	-	3	1	2
$(2,4,1)$	-	-	-	3	1	2
$(2,4,2)$	-	-	-	3	1	2
$(2,4,3)$	-	-	-	2	1	3
$(2,4,4)$	-	-	-	2	3	1
$(2,4,5)$	-	-	-	2	3	1
$(2,4,6)$	-	-	-	2	3	1
$(2,5,0)$	-	-	-	3	1	2
$(2,5,1)$	-	-	-	3	1	2
$(2,5,2)$	-	-	-	2	1	3
$(2,5,3)$	-	-	-	2	1	3
$(2,5,4)$	-	-	-	2	1	3
$(2,5,5)$	-	-	-	2	3	1
$(2,5,6)$	-	-	-	2	3	1
$(2,6,0)$	-	-	-	3	1	2
$(2,6,1)$	-	-	-	3	1	2
$(2,6,2)$	-	-	-	3	1	2
$(2,6,3)$	-	-	-	2	1	3
$(2,6,4)$	-	-	-	2	1	3

$(2,6,5)$	-	-	-	2	1	3
$(2,6,6)$	-	-	-	2	1	3
$(3,0,0)$	-	-	-	1	3	2
$(3,0,1)$	-	-	-	1	2	3
$(3,0,2)$	-	-	-	1	2	3
$(3,0,3)$	-	-	-	3	2	1
$(3,0,4)$	-	-	-	3	2	1
$(3,0,5)$	-	-	-	3	2	1
$(3,0,6)$	-	-	-	3	2	1
$(3,1,0)$	-	-	-	1	3	2
$(3,1,1)$	-	-	-	1	2	3
$(3,1,2)$	-	-	-	1	2	3
$(3,1,3)$	-	-	-	1	2	3
$(3,1,4)$	-	-	-	3	2	1
$(3,1,5)$	-	-	-	3	2	1
$(3,1,6)$	-	-	-	3	2	1
$(3,2,0)$	-	-	-	1	3	2
$(3,2,1)$	-	-	-	1	3	2
$(3,2,2)$	-	-	-	1	3	2
$(3,2,3)$	-	-	-	3	2	1
$(3,2,4)$	-	-	-	3	2	1
$(3,2,5)$	-	-	-	3	2	1
$(3,2,6)$	-	-	-	3	2	1
$(3,3,0)$	-	-	-	3	1	2
$(3,3,1)$	-	-	-	1	3	2
$(3,3,2)$	-	-	-	3	1	2
$(3,3,3)$	-	-	-	3	2	1
$(3,3,4)$	-	-	-	2	3	1
$(3,3,5)$	-	-	-	2	3	1
$(3,3,6)$	-	-	-	3	2	1
$(3,4,0)$	-	-	-	3	1	2
$(3,4,1)$	-	-	-	3	1	2
$(3,4,2)$	-	-	-	3	1	2
$(3,4,3)$	-	-	-	2	1	3
$(3,4,4)$	-	-	-	2	3	1
$(3,4,5)$	-	-	-	2	3	1
$(3,4,6)$	-	-	-	2	3	1
$(3,5,0)$	-	-	-	3	1	2
$(3,5,1)$	-	-	-	3	1	2
$(3,5,2)$	-	-	-	3	1	2
$(3,5,3)$	-	-	-	2	1	3
$(3,5,4)$	-	-	-	2	1	3
$(3,5,5)$	-	-	-	2	3	1
$(3,5,6)$	-	-	-	2	3	1
$(3,6,0)$	-	-	-	3	1	2
$(3,6,1)$	-	-	-	3	1	2
$(3,6,2)$	-	-	-	3	1	2
$(3,6,3)$	-	-	-	2	1	3
$(3,6,4)$	-	-	-	2	1	3
$(3,6,5)$	-	-	-	2	1	3
$(3,6,6)$	-	-	-	2	3	1
$(4,0,0)$	-	-	-	1	2	3
$(4,0,1)$	-	-	-	1	2	3
$(4,0,2)$	-	-	-	1	2	3
$(4,0,3)$	-	-	-	1	2	3
$(4,0,4)$	-	-	-	1	2	3
$(4,0,5)$	-	-	-	3	2	1
$(4,0,6)$	-	-	-	3	2	1
$(4,1,0)$	-	-	-	1	3	2
$(4,1,1)$	-	-	-	1	3	2
$(4,1,2)$	-	-	-	1	2	3
$(4,1,3)$	-	-	-	1	2	3
$(4,1,4)$	-	-	-	3	2	1
$(4,1,5)$	-	-	-	3	2	1

$(4,1,6)$	-	-	-	3	2	1
$(4,2,0)$	-	-	-	1	3	2
$(4,2,1)$	-	-	-	1	3	2
$(4,2,2)$	-	-	-	1	3	2
$(4,2,3)$	-	-	-	1	2	3
$(4,2,4)$	-	-	-	3	2	1
$(4,2,5)$	-	-	-	3	2	1
$(4,2,6)$	-	-	-	3	2	1
$(4,3,0)$	-	-	-	1	3	2
$(4,3,1)$	-	-	-	1	3	2
$(4,3,2)$	-	-	-	1	3	2
$(4,3,3)$	-	-	-	1	3	2
$(4,3,4)$	-	-	-	3	2	1
$(4,3,5)$	-	-	-	3	2	1
$(4,3,6)$	-	-	-	3	2	1
$(4,4,0)$	-	-	-	1	3	2
$(4,4,1)$	-	-	-	3	1	2
$(4,4,2)$	-	-	-	1	3	2
$(4,4,3)$	-	-	-	3	1	2
$(4,4,4)$	-	-	-	3	2	1
$(4,4,5)$	-	-	-	2	3	1
$(4,4,6)$	-	-	-	3	2	1
$(4,0,0)$	-	-	-	1	2	3
$(4,0,1)$	-	-	-	1	2	3
$(4,0,2)$	-	-	-	1	2	3
$(4,0,3)$	-	-	-	1	2	3
$(4,0,4)$	-	-	-	1	2	3
$(4,0,5)$	-	-	-	3	2	1
$(4,0,6)$	-	-	-	3	2	1
$(4,1,0)$	-	-	-	1	3	2
$(4,1,1)$	-	-	-	1	3	2
$(4,1,2)$	-	-	-	1	2	3
$(4,1,3)$	-	-	-	1	2	3
$(4,1,4)$	-	-	-	3	2	1
$(4,1,5)$	-	-	-	3	2	1
$(4,1,6)$	-	-	-	3	2	1
$(4,2,0)$	-	-	-	1	3	2
$(4,2,1)$	-	-	-	1	3	2
$(4,2,2)$	-	-	-	1	3	2
$(4,2,3)$	-	-	-	1	2	3
$(4,2,4)$	-	-	-	3	2	1
$(4,2,5)$	-	-	-	3	2	1
$(4,2,6)$	-	-	-	3	2	1
$(4,3,0)$	-	-	-	1	3	2
$(4,3,1)$	-	-	-	1	3	2
$(4,3,2)$	-	-	-	1	3	2
$(4,3,3)$	-	-	-	1	3	2
$(4,3,4)$	-	-	-	3	2	1
$(4,3,5)$	-	-	-	3	2	1
$(4,3,6)$	-	-	-	3	2	1
$(4,4,0)$	-	-	-	1	3	2
$(4,4,1)$	-	-	-	3	1	2
$(4,4,2)$	-	-	-	1	3	2
$(4,4,3)$	-	-	-	3	1	2
$(4,4,4)$	-	-	-	3	2	1
$(4,4,5)$	-	-	-	2	3	1
$(4,4,6)$	-	-	-	3	2	1
$(4,5,0)$	-	-	-	3	1	2
$(4,5,1)$	-	-	-	3	1	2
$(4,5,2)$	-	-	-	3	1	2
$(4,5,3)$	-	-	-	3	1	2
$(4,5,4)$	-	-	-	2	1	3
$(4,5,5)$	-	-	-	2	3	1
$(4,5,6)$	-	-	-	2	3	1

$(4,6,0)$	-	-	-	3	1	2
$(4,6,1)$	-	-	-	3	1	2
$(4,6,2)$	-	-	-	3	1	2
$(4,6,3)$	-	-	-	3	1	2
$(4,6,4)$	-	-	-	3	1	2
$(4,6,5)$	-	-	-	2	1	3
$(4,6,6)$	-	-	-	2	1	3
$(5,0,0)$	-	-	-	1	3	2
$(5,0,1)$	-	-	-	1	2	3
$(5,0,2)$	-	-	-	1	2	3
$(5,0,3)$	-	-	-	1	2	3
$(5,0,4)$	-	-	-	1	2	3
$(5,0,5)$	-	-	-	1	2	3
$(5,0,6)$	-	-	-	3	2	1
$(5,1,0)$	-	-	-	1	3	2
$(5,1,1)$	-	-	-	1	2	3
$(5,1,2)$	-	-	-	1	2	3
$(5,1,3)$	-	-	-	1	2	3
$(5,1,4)$	-	-	-	1	2	3
$(5,1,5)$	-	-	-	1	2	3
$(5,1,6)$	-	-	-	3	2	1
$(5,2,0)$	-	-	-	1	3	2
$(5,2,1)$	-	-	-	1	3	2
$(5,2,2)$	-	-	-	1	3	2
$(5,2,3)$	-	-	-	1	2	3
$(5,2,4)$	-	-	-	3	2	1
$(5,2,5)$	-	-	-	3	2	1
$(5,2,6)$	-	-	-	3	2	1
$(5,3,0)$	-	-	-	1	3	2
$(5,3,1)$	-	-	-	1	3	2
$(5,3,2)$	-	-	-	1	3	2
$(5,3,3)$	-	-	-	1	3	2
$(5,3,4)$	-	-	-	2	2	3
$(5,3,5)$	-	-	-	3	2	1
$(5,3,6)$	-	-	-	3	2	1
$(5,4,0)$	-	-	-	1	3	2
$(5,4,1)$	-	-	-	1	3	2
$(5,4,2)$	-	-	-	1	3	2
$(5,4,3)$	-	-	-	1	3	2
$(5,4,4)$	-	-	-	1	2	3
$(5,4,5)$	-	-	-	3	2	1
$(5,4,6)$	-	-	-	3	2	1
$(5,5,0)$	-	-	-	1	2	2
$(5,5,1)$	-	-	-	1	2	2
$(5,5,2)$	-	-	-	3	1	2
$(5,5,3)$	-	-	-	3	1	2
$(5,5,4)$	-	-	-	3	1	2
$(5,5,5)$	-	-	-	2	3	1
$(5,5,6)$	-	-	-	2	3	1
$(5,6,0)$	-	-	-	3	1	2
$(5,6,1)$	-	-	-	3	1	2
$(5,6,2)$	-	-	-	3	1	2
$(5,6,3)$	-	-	-	3	1	2
$(5,6,4)$	-	-	-	3	1	2
$(5,6,5)$	-	-	-	2	1	3
$(5,6,6)$	-	-	-	2	1	3
$(6,0,0)$	-	-	-	1	3	2
$(6,0,1)$	-	-	-	1	2	3
$(6,0,2)$	-	-	-	1	2	3
$(6,0,3)$	-	-	-	1	2	3
$(6,0,4)$	-	-	-	1	2	3
$(6,0,5)$	-	-	-	1	2	3
$(6,0,6)$	-	-	-	1	2	3
$(6,1,0)$	-	-	-	1	3	2

$(6,1,1)$	-	-	-	1	2	3
$(6,1,2)$	-	-	-	1	2	3
$(6,1,3)$	-	-	-	1	2	3
$(6,1,4)$	-	-	-	1	2	3
$(6,1,5)$	-	-	-	1	2	3
$(6,1,6)$	-	-	-	1	2	3
$(6,2,0)$	-	-	-	1	3	2
$(6,2,1)$	-	-	-	1	3	2
$(6,2,2)$	-	-	-	1	2	3
$(6,2,3)$	-	-	-	1	2	3
$(6,2,4)$	-	-	-	1	2	3
$(6,2,5)$	-	-	-	1	2	3
$(6,2,6)$	-	-	-	3	2	1
$(6,3,0)$	-	-	-	1	3	2
$(6,3,1)$	-	-	-	1	3	2
$(6,3,2)$	-	-	-	1	3	2
$(6,3,3)$	-	-	-	1	3	2
$(6,3,4)$	-	-	-	1	2	3
$(6,3,5)$	-	-	-	1	2	3
$(6,3,6)$	-	-	-	3	2	1
$(6,4,0)$	-	-	-	1	3	2
$(6,4,1)$	-	-	-	1	3	2
$(6,4,2)$	-	-	-	1	3	2
$(6,4,3)$	-	-	-	1	3	2
$(6,4,4)$	-	-	-	1	3	2
$(6,4,5)$	-	-	-	1	2	3
$(6,4,6)$	-	-	-	1	2	3
$(6,5,0)$	-	-	-	1	3	2
$(6,5,1)$	-	-	-	1	3	2
$(6,5,2)$	-	-	-	1	3	2
$(6,5,3)$	-	-	-	1	3	2
$(6,5,4)$	-	-	-	1	3	2
$(6,5,5)$	-	-	-	1	2	3
$(6,5,6)$	-	-	-	3	2	1
$(6,6,0)$	-	-	-	1	3	2
$(6,6,1)$	-	-	-	1	3	2
$(6,6,2)$	-	-	-	1	3	2
$(6,6,3)$	-	-	-	3	1	2
$(6,6,4)$	-	-	-	1	3	2
$(6,6,5)$	-	-	-	1	3	2
$(6,6,6)$	-	-	-	2	3	1

E.2 States lexicographically ordered

In this section we will look at the results from the DP when the states are ordered lexicographically. First, the results are shown when the arrival rates during the different schedules are the same. The results are shown in Section E.2.1 when the number of time blocks is equal to the number of different tasks and in Section E.2.2 when the number of time blocks is not equal to the number of different tasks. Then, the results are shown when the arrival rates differ per schedule. The results are shown in Section E.2.3 when the schedule with different arrival rates is in schedule three and in total four schedules are made and in Section E.2.4 when the schedule with different arrival rates is in schedule five and in total five schedules are made. In Section E.2.5 the results are shown when we have bigger differences between the arrival rates of the different tasks.

E.2.1 Same arrival rate during schedules and number of time blocks equal to number of different tasks

The results from the DP with three persons who can be scheduled over three tasks with three time blocks and when five schedules are made, are shown in Tables E.2 (fifth schedule) – E.6 (first schedule). This tables show an optimal action given the state and decision epoch. A 1 corresponds to task one with λ_1, a 2 corresponds to task two with λ_2 and a 3 corresponds to task three with λ_3. The parameters used in this case are shown below.

$$
\begin{aligned}
\lambda_1 &= 0.05 \\
\lambda_2 &= 0.5 \\
\lambda_3 &= 0.3 \\
q &= 0.01 \\
N &= 6 \\
J &= 3 \\
M &= 3 \\
B &= 3
\end{aligned}
$$

Table E.2: The DP results for the fifth schedule of a small instance when the states are ordered lexicographically.

State	$p=1$			$p=2$			$p=3$			Expected new state
	$d=1$	$d=2$	$d=3$	$d=1$	$d=2$	$d=3$	$d=1$	$d=2$	$d=3$	
$(0,0,0)$	1	3	2	2	1	3	3	2	1	$(0.85,0.85,0.85)$
$(1,0,0)$	1	1	1	2	2	3	3	3	2	$(1.15,1.3,1.1)$
$(1,1,0)$	1	1	3	3	3	1	2	2	2	$(1.4,1.65,1.5)$
$(2,0,0)$	1	1	1	2	3	3	3	2	2	$(2.15,1.1,1.3)$
$(2,1,0)$	1	1	1	3	3	3	2	2	2	$(2.15,1.9,1.5)$
$(2,2,0)$	1	1	3	3	3	1	2	2	2	$(2.4,2.65,1.5)$
$(3,0,0)$	1	1	1	2	2	3	3	3	2	$(3.15,1.3,1.1)$
$(3,1,0)$	1	1	1	3	3	3	2	2	2	$(3.15,1.9,1.5)$
$(3,2,0)$	1	1	1	3	3	3	2	2	2	$(3.15,2.9,1.5)$
$(3,3,0)$	1	1	3	3	3	1	2	2	2	$(3.4,3.65,1.5)$
$(4,0,0)$	1	1	1	2	2	3	3	3	2	$(4.15,1.3,1.1)$
$(4,1,0)$	1	1	1	3	3	3	2	2	2	$(4.15,1.9,1.5)$
$(4,2,0)$	1	1	1	3	3	3	2	2	2	$(4.15,2.9,1.5)$
$(4,3,0)$	1	1	1	3	3	3	2	2	2	$(4.15,3.9,1.5)$
$(4,4,0)$	1	1	3	3	3	1	2	2	2	$(4.4,4.65,1.5)$
$(5,0,0)$	1	1	1	2	2	3	3	3	2	$(5.15,1.3,1.1)$
$(5,1,0)$	1	1	1	3	3	3	2	2	2	$(5.15,1.9,1.5)$
$(5,2,0)$	1	1	1	3	3	3	2	2	2	$(5.15,2.9,1.5)$
$(5,3,0)$	1	1	1	3	3	3	2	2	2	$(5.15,3.9,1.5)$
$(5,4,0)$	1	1	1	3	3	3	2	2	2	$(5.15,4.9,1.5)$
$(5,5,0)$	1	1	3	3	3	1	2	2	2	$(5.4,5.65,1.5)$
$(6,0,0)$	1	1	1	2	2	3	3	3	2	$(6.15,1.3,1.1)$
$(6,1,0)$	1	1	1	3	3	3	2	2	2	$(6.15,1.9,1.5)$
$(6,2,0)$	1	1	1	3	3	3	2	2	2	$(6.15,2.9,1.5)$
$(6,3,0)$	1	1	1	3	3	3	2	2	2	$(6.15,3.9,1.5)$

$(6,4,0)$	1	1	1	3	3	3	2	2	2	$(6.15,4.9,1.5)$
$(6,5,0)$	1	1	1	3	3	3	2	2	2	$(6.15,5.9,1.5)$
$(6,6,0)$	1	1	3	3	3	1	2	2	2	$(6.4,6.65,1.5)$
$(7,0,0)$	1	1	1	2	2	3	3	3	2	$(7.15,1.3,1.1)$
$(7,1,0)$	1	1	1	3	3	3	2	2	2	$(7.15,1.9,1.5)$
$(7,2,0)$	1	1	1	3	3	3	2	2	2	$(7.15,2.9,1.5)$
$(7,3,0)$	1	1	1	3	3	3	2	2	2	$(7.15,3.9,1.5)$
$(7,4,0)$	1	1	1	3	3	3	2	2	2	$(7.15,4.9,1.5)$
$(7,5,0)$	1	1	1	3	3	3	2	2	2	$(7.15,5.9,1.5)$
$(7,6,0)$	1	1	1	3	3	3	2	2	2	$(7.15,6.9,1.5)$
$(7,7,0)$	1	1	3	3	3	1	2	2	2	$(7.4,7.65,1.5)$
$(8,0,0)$	1	1	1	2	2	3	3	3	2	$(8.15,1.3,1.1)$
$(8,1,0)$	1	1	1	3	3	3	2	2	2	$(8.15,1.9,1.5)$
$(8,2,0)$	1	1	1	3	3	3	2	2	2	$(8.15,2.9,1.5)$
$(8,3,0)$	1	1	1	3	3	3	2	2	2	$(8.15,3.9,1.5)$
$(8,4,0)$	1	1	1	3	3	3	2	2	2	$(8.15,4.9,1.5)$
$(8,5,0)$	1	1	1	3	3	3	2	2	2	$(8.15,5.9,1.5)$
$(8,6,0)$	1	1	1	3	3	3	2	2	2	$(8.15,6.9,1.5)$
$(8,7,0)$	1	1	1	3	3	3	2	2	2	$(8.15,7.9,1.5)$
$(8,8,0)$	1	1	3	3	3	1	2	2	2	$(8.4,8.65,1.5)$
$(9,0,0)$	1	1	1	2	2	3	3	3	2	$(9.15,1.3,1.1)$
$(9,1,0)$	1	1	1	3	3	3	2	2	2	$(9.15,1.9,1.5)$
$(9,2,0)$	1	1	1	3	3	3	2	2	2	$(9.15,2.9,1.5)$
$(9,3,0)$	1	1	1	3	3	3	2	2	2	$(9.15,3.9,1.5)$
$(9,4,0)$	1	1	1	3	3	3	2	2	2	$(9.15,4.9,1.5)$
$(9,5,0)$	1	1	1	3	3	3	2	2	2	$(9.15,5.9,1.5)$
$(9,6,0)$	1	1	1	3	3	3	2	2	2	$(9.15,6.9,1.5)$
$(9,7,0)$	1	1	1	3	3	3	2	2	2	$(9.15,7.9,1.5)$
$(9,8,0)$	1	1	1	3	3	3	2	2	2	$(9.15,8.9,1.5)$
$(9,9,0)$	1	1	3	3	3	1	2	2	2	$(9.4,9.65,1.5)$
$(10,0,0)$	1	1	1	2	2	3	3	3	2	$(10.15,1.3,1.1)$
$(10,1,0)$	1	1	1	3	3	3	2	2	2	$(10.15,1.9,1.5)$
$(10,2,0)$	1	1	1	3	3	3	2	2	2	$(10.15,2.9,1.5)$
$(10,3,0)$	1	1	1	3	3	3	2	2	2	$(10.15,3.9,1.5)$
$(10,4,0)$	1	1	1	3	3	3	2	2	2	$(10.15,4.9,1.5)$
$(10,5,0)$	1	1	1	3	3	3	2	2	2	$(10.15,5.9,1.5)$
$(10,6,0)$	1	1	1	3	3	3	2	2	2	$(10.15,6.9,1.5)$
$(10,7,0)$	1	1	1	3	3	3	2	2	2	$(10.15,7.9,1.5)$
$(10,8,0)$	1	1	1	3	3	3	2	2	2	$(10.15,8.9,1.5)$
$(10,9,0)$	1	1	1	3	3	3	2	2	2	$(10.15,9.9,1.5)$
$(10,10,0)$	1	1	3	3	3	1	2	2	2	$(10.4,10.65,1.5)$
$(11,0,0)$	1	1	1	2	2	3	3	3	2	$(11.15,1.3,1.1)$
$(11,1,0)$	1	1	1	3	3	3	2	2	2	$(11.15,1.9,1.5)$
$(11,2,0)$	1	1	1	3	3	3	2	2	2	$(11.15,2.9,1.5)$
$(11,3,0)$	1	1	1	3	3	3	2	2	2	$(11.15,3.9,1.5)$
$(11,4,0)$	1	1	1	3	3	3	2	2	2	$(11.15,4.9,1.5)$
$(11,5,0)$	1	1	1	3	3	3	2	2	2	$(11.15,5.9,1.5)$
$(11,6,0)$	1	1	1	3	3	3	2	2	2	$(11.15,6.9,1.5)$
$(11,7,0)$	1	1	1	3	3	3	2	2	2	$(11.15,7.9,1.5)$
$(11,8,0)$	1	1	1	3	3	3	2	2	2	$(11.15,8.9,1.5)$
$(11,9,0)$	1	1	1	3	3	3	2	2	2	$(11.15,9.9,1.5)$
$(11,10,0)$	1	1	1	3	3	3	2	2	2	$(11.15,10.9,1.5)$
$(11,11,0)$	1	1	3	3	3	1	2	2	2	$(11.4,11.65,1.5)$
$(12,0,0)$	1	1	1	2	2	3	3	3	2	$(12.15,1.3,1.1)$
$(12,1,0)$	1	1	1	3	3	3	2	2	2	$(12.15,1.9,1.5)$
$(12,2,0)$	1	1	1	3	3	3	2	2	2	$(12.15,2.9,1.5)$
$(12,3,0)$	1	1	1	3	3	3	2	2	2	$(12.15,3.9,1.5)$
$(12,4,0)$	1	1	1	3	3	3	2	2	2	$(12.15,4.9,1.5)$
$(12,5,0)$	1	1	1	3	3	3	2	2	2	$(12.15,5.9,1.5)$
$(12,6,0)$	1	1	1	3	3	3	2	2	2	$(12.15,6.9,1.5)$
$(12,7,0)$	1	1	1	3	3	3	2	2	2	$(12.15,7.9,1.5)$
$(12,8,0)$	1	1	1	3	3	3	2	2	2	$(12.15,8.9,1.5)$
$(12,9,0)$	1	1	1	3	3	3	2	2	2	$(12.15,9.9,1.5)$
$(12,10,0)$	1	1	1	3	3	3	2	2	2	$(12.15,10.9,1.5)$

$(12,11,0)$	1	1	1	3	3	3	2	2	2	$(12.15,11.9,1.5)$
$(12,12,0)$	1	1	1	3	3	3	2	2	2	$(12.15,12.9,1.5)$

Table E.3: The DP results for the fourth schedule of a small instance when the states are ordered lexicographically.

State	$p=1$			$p=2$			$p=3$			Expected new state
	$d=1$	$d=2$	$d=3$	$d=1$	$d=2$	$d=3$	$d=1$	$d=2$	$d=3$	
$(0,0,0)$	1	3	2	2	1	3	3	2	1	$(0.85,0.85,0.85)$
$(1,0,0)$	1	1	3	2	2	1	3	3	2	$(1.4,1.05,1.1)$
$(1,1,0)$	1	1	2	3	3	1	2	2	3	$(1.6,1.65,1.3)$
$(2,0,0)$	1	1	1	2	2	3	3	3	2	$(2.15,1.3,1.1)$
$(2,1,0)$	1	1	1	3	3	3	2	2	2	$(2.15,1.9,1.5)$
$(2,2,0)$	1	3	1	3	1	3	2	2	2	$(2.4,2.65,1.5)$
$(3,0,0)$	1	1	1	2	2	3	3	3	2	$(3.15,1.3,1.1)$
$(3,1,0)$	1	1	1	3	3	3	2	2	2	$(3.15,1.9,1.5)$
$(3,2,0)$	1	1	1	3	3	3	2	2	2	$(3.15,2.9,1.5)$
$(3,3,0)$	1	1	3	3	3	1	2	2	2	$(3.4,3.65,1.5)$
$(4,0,0)$	1	1	1	2	2	3	3	3	2	$(4.15,1.3,1.1)$
$(4,1,0)$	1	1	1	3	3	3	2	2	2	$(4.15,1.9,1.5)$
$(4,2,0)$	1	1	1	3	3	3	2	2	2	$(4.15,2.9,1.5)$
$(4,3,0)$	1	1	1	3	3	3	2	2	2	$(4.15,3.9,1.5)$
$(4,4,0)$	1	1	3	3	3	1	2	2	2	$(4.4,4.65,1.5)$
$(5,0,0)$	1	1	1	2	2	3	3	3	2	$(5.15,1.3,1.1)$
$(5,1,0)$	1	1	1	3	3	3	2	2	2	$(5.15,1.9,1.5)$
$(5,2,0)$	1	1	1	3	3	3	2	2	2	$(5.15,2.9,1.5)$
$(5,3,0)$	1	1	1	3	3	3	2	2	2	$(5.15,3.9,1.5)$
$(5,4,0)$	1	1	1	3	3	3	2	2	2	$(5.15,4.9,1.5)$
$(5,5,0)$	1	1	3	3	3	1	2	2	2	$(5.4,5.65,1.5)$
$(6,0,0)$	1	1	1	2	3	3	3	2	2	$(6.15,1.1,1.3)$
$(6,1,0)$	1	1	1	3	3	3	2	2	2	$(6.15,1.9,1.5)$
$(6,2,0)$	1	1	1	3	3	3	2	2	2	$(6.15,2.9,1.5)$
$(6,3,0)$	1	1	1	3	3	3	2	2	2	$(6.15,3.9,1.5)$
$(6,4,0)$	1	1	1	3	3	3	2	2	2	$(6.15,4.9,1.5)$
$(6,5,0)$	1	1	1	3	3	3	2	2	2	$(6.15,5.9,1.5)$
$(6,6,0)$	1	1	3	3	3	1	2	2	2	$(6.4,6.65,1.5)$
$(7,0,0)$	1	1	1	2	2	3	3	3	2	$(7.15,1.3,1.1)$
$(7,1,0)$	1	1	1	3	3	3	2	2	2	$(7.15,1.9,1.5)$
$(7,2,0)$	1	1	1	3	3	3	2	2	2	$(7.15,2.9,1.5)$
$(7,3,0)$	1	1	1	3	3	3	2	2	2	$(7.15,3.9,1.5)$
$(7,4,0)$	1	1	1	3	3	3	2	2	2	$(7.15,4.9,1.5)$
$(7,5,0)$	1	1	1	3	3	3	2	2	2	$(7.15,5.9,1.5)$
$(7,6,0)$	1	1	1	3	3	3	2	2	2	$(7.15,6.9,1.5)$
$(7,7,0)$	1	1	3	3	3	1	2	2	2	$(7.4,7.65,1.5)$
$(8,0,0)$	1	1	1	2	2	3	3	3	2	$(8.15,1.3,1.1)$
$(8,1,0)$	1	1	1	3	3	3	2	2	2	$(8.15,1.9,1.5)$
$(8,2,0)$	1	1	1	3	3	3	2	2	2	$(8.15,2.9,1.5)$
$(8,3,0)$	1	1	1	3	3	3	2	2	2	$(8.15,3.9,1.5)$
$(8,4,0)$	1	1	1	3	3	3	2	2	2	$(8.15,4.9,1.5)$
$(8,5,0)$	1	1	1	3	3	3	2	2	2	$(8.15,5.9,1.5)$
$(8,6,0)$	1	1	1	3	3	3	2	2	2	$(8.15,6.9,1.5)$
$(8,7,0)$	1	1	1	3	3	3	2	2	2	$(8.15,7.9,1.5)$
$(8,8,0)$	1	1	3	3	3	1	2	2	2	$(8.4,8.65,1.5)$
$(9,0,0)$	1	1	1	2	2	3	3	3	2	$(9.15,1.3,1.1)$
$(9,1,0)$	1	1	1	3	3	3	2	2	2	$(9.15,1.9,1.5)$
$(9,2,0)$	1	1	1	3	3	3	2	2	2	$(9.15,2.9,1.5)$
$(9,3,0)$	1	1	1	3	3	3	2	2	2	$(9.15,3.9,1.5)$
$(9,4,0)$	1	1	1	3	3	3	2	2	2	$(9.15,4.9,1.5)$
$(9,5,0)$	1	1	1	3	3	3	2	2	2	$(9.15,5.9,1.5)$
$(9,6,0)$	1	1	1	3	3	3	2	2	2	$(9.15,6.9,1.5)$
$(9,7,0)$	1	1	1	3	3	3	2	2	2	$(9.15,7.9,1.5)$
$(9,8,0)$	1	1	1	3	3	3	2	2	2	$(9.15,8.9,1.5)$

$(9,9,0)$	1	1	3	3	3	1	2	2	2	$(9.4,9.65,1.5)$
$(10,0,0)$	1	1	1	2	2	3	3	3	2	$(10.15,1.3,1.1)$
$(10,1,0)$	1	1	1	3	3	3	2	2	2	$(10.15,1.9,1.5)$
$(10,2,0)$	1	1	1	3	3	3	2	2	2	$(10.15,2.9,1.5)$
$(10,3,0)$	1	1	1	3	3	3	2	2	2	$(10.15,3.9,1.5)$
$(10,4,0)$	1	1	1	3	3	3	2	2	2	$(10.15,4.9,1.5)$
$(10,5,0)$	1	1	1	3	3	3	2	2	2	$(10.15,5.9,1.5)$
$(10,6,0)$	1	1	1	3	3	3	2	2	2	$(10.15,6.9,1.5)$
$(10,7,0)$	1	1	1	3	3	3	2	2	2	$(10.15,7.9,1.5)$
$(10,8,0)$	1	1	1	3	3	3	2	2	2	$(10.15,8.9,1.5)$
$(10,9,0)$	1	1	1	3	3	3	2	2	2	$(10.15,9.9,1.5)$
$(10,10,0)$	1	1	1	3	3	3	2	2	2	$(10.15,10.9,1.5)$

Table E.4: The DP results for the third schedule of a small instance when the states are ordered lexicographically.

State	$p=1$			$p=2$			$p=3$			Expected new state
	$d=1$	$d=2$	$d=3$	$d=1$	$d=2$	$d=3$	$d=1$	$d=2$	$d=3$	
$(0,0,0)$	1	3	2	2	1	3	3	2	1	$(0.85,0.85,0.85)$
$(1,0,0)$	1	1	3	3	3	2	2	2	1	$(1.4,1.1,1.05)$
$(1,1,0)$	1	1	2	3	3	1	2	2	3	$(1.6,1.65,1.3)$
$(2,0,0)$	1	1	1	2	2	3	3	3	2	$(2.15,1.3,1.1)$
$(2,1,0)$	1	1	1	3	3	3	2	2	2	$(2.15,1.9,1.5)$
$(2,2,0)$	1	1	3	3	3	1	2	2	2	$(2.4,2.65,1.5)$
$(3,0,0)$	1	1	1	2	2	3	3	3	2	$(3.15,1.3,1.1)$
$(3,1,0)$	1	1	1	3	3	3	2	2	2	$(3.15,1.9,1.5)$
$(3,2,0)$	1	1	1	3	3	3	2	2	2	$(3.15,2.9,1.5)$
$(3,3,0)$	1	1	3	3	3	1	2	2	2	$(3.4,3.65,1.5)$
$(4,0,0)$	1	1	1	2	2	3	3	3	2	$(4.15,1.3,1.1)$
$(4,1,0)$	1	1	1	3	3	3	2	2	2	$(4.15,1.9,1.5)$
$(4,2,0)$	1	1	1	3	3	3	2	2	2	$(4.15,2.9,1.5)$
$(4,3,0)$	1	1	1	3	3	3	2	2	2	$(4.15,3.9,1.5)$
$(4,4,0)$	1	1	3	3	3	1	2	2	2	$(4.4,4.65,1.5)$
$(5,0,0)$	1	1	1	2	2	3	3	3	2	$(5.15,1.3,1.1)$
$(5,1,0)$	1	1	1	3	3	3	2	2	2	$(5.15,1.9,1.5)$
$(5,2,0)$	1	1	1	3	3	3	2	2	2	$(5.15,2.9,1.5)$
$(5,3,0)$	1	1	1	3	3	3	2	2	2	$(5.15,3.9,1.5)$
$(5,4,0)$	1	1	1	3	3	3	2	2	2	$(5.15,4.9,1.5)$
$(5,5,0)$	1	1	3	3	3	1	2	2	2	$(5.4,5.65,1.5)$
$(6,0,0)$	1	1	1	2	2	3	3	3	2	$(6.15,1.3,1.1)$
$(6,1,0)$	1	1	1	3	3	3	2	2	2	$(6.15,1.9,1.5)$
$(6,2,0)$	1	1	1	3	3	3	2	2	2	$(6.15,2.9,1.5)$
$(6,3,0)$	1	1	1	3	3	3	2	2	2	$(6.15,3.9,1.5)$
$(6,4,0)$	1	1	1	3	3	3	2	2	2	$(6.15,4.9,1.5)$
$(6,5,0)$	1	1	1	3	3	3	2	2	2	$(6.15,5.9,1.5)$
$(6,6,0)$	1	1	3	3	3	1	2	2	2	$(6.4,6.65,1.5)$
$(7,0,0)$	1	1	1	2	3	3	3	2	2	$(7.15,1.1,1.3)$
$(7,1,0)$	1	1	1	3	3	3	2	2	2	$(7.15,1.9,1.5)$
$(7,2,0)$	1	1	1	3	3	3	2	2	2	$(7.15,2.9,1.5)$
$(7,3,0)$	1	1	1	3	3	3	2	2	2	$(7.15,3.9,1.5)$
$(7,4,0)$	1	1	1	3	3	3	2	2	2	$(7.15,4.9,1.5)$
$(7,5,0)$	1	1	1	3	3	3	2	2	2	$(7.15,5.9,1.5)$
$(7,6,0)$	1	1	1	3	3	3	2	2	2	$(7.15,6.9,1.5)$
$(7,7,0)$	1	1	3	3	3	1	2	2	2	$(7.4,7.65,1.5)$

Table E.5: The DP results for the second schedule of a small instance when the states are ordered lexicographically.

State	$p=1$			$p=2$			$p=3$			Expected new state
	$d=1$	$d=2$	$d=3$	$d=1$	$d=2$	$d=3$	$d=1$	$d=2$	$d=3$	
$(0,0,0)$	1	3	2	2	1	3	3	2	1	$(0.85,0.85,0.85)$
$(1,0,0)$	1	1	3	3	3	2	2	2	1	$(1.4,1.1,1.05)$
$(1,1,0)$	1	1	2	3	3	1	2	2	3	$(1.6,1.65,1.3)$
$(2,0,0)$	1	1	1	2	2	3	3	3	2	$(2.15,1.3,1.1)$
$(2,1,0)$	1	1	1	3	3	3	2	2	2	$(2.15,1.9,1.5)$
$(2,2,0)$	1	1	3	3	3	1	2	2	2	$(2.4,2.65,1.5)$
$(3,0,0)$	1	1	1	2	2	3	3	3	2	$(3.15,1.3,1.1)$
$(3,1,0)$	1	1	1	3	3	3	2	2	2	$(3.15,1.9,1.5)$
$(3,2,0)$	1	1	1	3	3	3	2	2	2	$(3.15,2.9,1.5)$
$(3,3,0)$	1	1	3	3	3	1	2	2	2	$(3.4,3.65,1.5)$
$(4,0,0)$	1	1	1	2	2	3	3	3	2	$(4.15,1.3,1.1)$
$(4,1,0)$	1	1	1	3	3	3	2	2	2	$(4.15,1.9,1.5)$
$(4,2,0)$	1	1	1	3	3	3	2	2	2	$(4.15,2.9,1.5)$
$(4,3,0)$	1	1	1	3	3	3	2	2	2	$(4.15,3.9,1.5)$
$(4,4,0)$	1	1	3	3	3	1	2	2	2	$(4.4,4.65,1.5)$
$(5,0,0)$	1	1	1	2	2	3	3	3	2	$(5.15,1.3,1.1)$
$(5,1,0)$	1	1	1	3	3	3	2	2	2	$(5.15,1.9,1.5)$
$(5,2,0)$	1	1	1	3	3	3	2	2	2	$(5.15,2.9,1.5)$
$(5,3,0)$	1	1	1	3	3	3	2	2	2	$(5.15,3.9,1.5)$
$(5,4,0)$	1	1	1	3	3	3	2	2	2	$(5.15,4.9,1.5)$
$(5,5,0)$	1	1	1	3	3	3	2	2	2	$(5.15,5.9,1.5)$

Table E.6: The DP results for the first schedule of a small instance when the states are ordered lexicographically.

State	$p=1$			$p=2$			$p=3$			Expected new state
	$d=1$	$d=2$	$d=3$	$d=1$	$d=2$	$d=3$	$d=1$	$d=2$	$d=3$	
$(0,0,0)$	1	3	2	2	1	3	3	2	1	$(0.85,0.85,0.85)$

E.2.2 Same arrival rate during schedules and number of time blocks not equal to number of different tasks

The results from the DP with three persons who can be scheduled over three tasks with two time blocks and when five schedules are made, are shown in Tables E.7 (fifth schedule) – E.11 (first schedule). These tables show an optimal action given the state and decision epoch. A 1 corresponds to task one with λ_1, a 2 corresponds to task two with λ_2 and a 3 corresponds to task three with λ_3. The parameters used in this case are shown below.

$$\begin{aligned} \lambda_1 &= 0.05 \\ \lambda_2 &= 0.5 \\ \lambda_3 &= 0.3 \\ q &= 0.01 \\ N &= 6 \\ J &= 3 \\ M &= 3 \\ B &= 2 \end{aligned}$$

Table E.7: The DP results for the fifth schedule of a small instance when the states are ordered lexicographically and the number of time blocks is not equal to the number of different tasks.

State	$p=1$		$p=2$		$p=3$		Expected new state
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$	
$(0,0,0)$	1	2	2	1	3	3	$(0.55,0.55,0.6)$
$(1,0,0)$	1	1	2	3	3	1	$(1.1,0.8,0.8)$
$(1,1,0)$	1	3	3	1	2	2	$(1.35,1.35,1.0)$
$(2,0,0)$	1	1	2	3	3	1	$(2.1,0.8,0.8)$
$(2,1,0)$	1	1	3	3	2	2	$(2.1,1.6,1.0)$
$(2,2,0)$	1	3	3	1	2	2	$(2.35,2.35,1.0)$
$(3,0,0)$	1	1	2	3	3	1	$(3.1,0.8,0.8)$
$(3,1,0)$	1	1	3	3	2	2	$(3.1,1.6,1.0)$
$(3,2,0)$	1	1	3	3	2	2	$(3.1,2.6,1.0)$
$(3,3,0)$	1	3	3	1	2	2	$(3.35,3.35,1.0)$
$(4,0,0)$	1	1	2	3	3	1	$(4.1,0.8,0.8)$
$(4,1,0)$	1	1	3	3	2	2	$(4.1,1.6,1.0)$
$(4,2,0)$	1	1	3	3	2	2	$(4.1,2.6,1.0)$
$(4,3,0)$	1	1	3	3	2	2	$(4.1,2.6,1.0)$
$(4,4,0)$	1	3	3	1	2	2	$(4.35,4.35,1.0)$
$(5,0,0)$	1	1	2	3	3	1	$(5.1,0.8,0.8)$
$(5,1,0)$	1	1	3	3	2	2	$(5.1,1.6,1.0)$
$(5,2,0)$	1	1	3	3	2	2	$(5.1,2.6,1.0)$
$(5,3,0)$	1	1	3	3	2	2	$(5.1,3.6,1.0)$
$(5,4,0)$	1	1	3	3	2	2	$(5.1,4.6,1.0)$
$(5,5,0)$	1	3	3	1	2	2	$(5.35,5.35,1.0)$
$(6,0,0)$	1	1	2	3	3	1	$(6.1,0.8,0.8)$
$(6,1,0)$	1	1	3	3	2	2	$(6.1,1.6,1.0)$
$(6,2,0)$	1	1	3	3	2	2	$(6.1,2.6,1.0)$
$(6,3,0)$	1	1	3	3	2	2	$(6.1,3.6,1.0)$
$(6,4,0)$	1	1	3	3	2	2	$(6.1,4.6,1.0)$
$(6,5,0)$	1	1	3	3	2	2	$(6.1,5.6,1.0)$
$(6,6,0)$	1	3	3	1	2	2	$(6.35,6.35,1.0)$
$(7,0,0)$	1	1	2	3	3	1	$(7.1,0.8,0.8)$
$(7,1,0)$	1	1	3	3	2	2	$(7.1,1.6,1.0)$
$(7,2,0)$	1	1	3	3	2	2	$(7.1,2.6,1.0)$
$(7,3,0)$	1	1	3	3	2	2	$(7.1,3.6,1.0)$
$(7,4,0)$	1	1	3	3	2	2	$(7.1,4.6,1.0)$
$(7,5,0)$	1	1	3	3	2	2	$(7.1,5.6,1.0)$
$(7,6,0)$	1	1	3	3	2	2	$(7.1,6.6,1.0)$
$(7,7,0)$	1	3	3	1	2	2	$(7.35,7.35,1.0)$
$(8,0,0)$	1	1	2	3	3	1	$(8.1,0.8,0.8)$
$(8,1,0)$	1	1	3	3	2	2	$(8.1,1.6,1.0)$
$(8,2,0)$	1	1	3	3	2	2	$(8.1,2.6,1.0)$

$(8,3,0)$	1	1	3	3	2	2	$(8.1,3.6,1.0)$
$(8,4,0)$	1	1	3	3	2	2	$(8.1,4.6,1.0)$
$(8,5,0)$	1	1	3	3	2	2	$(8.1,5.6,1.0)$
$(8,6,0)$	1	1	3	3	2	2	$(8.1,6.6,1.0)$
$(8,7,0)$	1	1	3	3	2	2	$(8.1,7.6,1.0)$
$(8,8,0)$	1	1	3	3	2	2	$(8.1,8.6,1.0)$
$(9,0,0)$	1	1	2	3	3	2	$(9.1,0.8,0.8)$
$(9,1,0)$	1	1	3	3	2	2	$(9.1,1.6,1.0)$
$(9,2,0)$	1	1	3	3	2	2	$(9.1,2.6,1.0)$
$(9,3,0)$	1	1	3	3	2	2	$(9.1,3.6,1.0)$
$(9,4,0)$	1	1	3	3	2	2	$(9.1,4.6,1.0)$
$(9,5,0)$	1	1	3	3	2	2	$(9.1,5.6,1.0)$
$(9,6,0)$	1	1	3	3	2	2	$(9.1,6.6,1.0)$
$(9,7,0)$	1	1	3	3	2	2	$(9.1,7.6,1.0)$
$(9,8,0)$	1	1	3	3	2	2	$(9.1,8.6,1.0)$
$(9,9,0)$	1	1	3	3	2	2	$(9.1,9.6,1.0)$

Table E.8: The DP results for the fourth schedule of a small instance when the states are ordered lexicographically and the number of time blocks is not equal to the number of different tasks.

State	$p=1$		$p=2$		$p=3$		Expected new state
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$	
$(0,0,0)$	1	2	2	1	3	3	$(0.55,0.55,0.6)$
$(1,0,0)$	1	1	2	3	3	2	$(1.1,0.8,0.8)$
$(1,1,0)$	1	3	3	1	2	2	$(1.35,1.35,1.0)$
$(2,0,0)$	1	1	2	3	3	2	$(2.1,0.8,0.8)$
$(2,1,0)$	1	1	3	3	2	2	$(2.1,1.6,1.0)$
$(2,2,0)$	1	3	3	1	2	2	$(2.35,2.35,1.0)$
$(3,0,0)$	1	1	2	3	3	2	$(3.1,0.8,0.8)$
$(3,1,0)$	1	1	3	3	2	2	$(3.1,1.6,1.0)$
$(3,2,0)$	1	1	3	3	2	2	$(3.1,2.6,1.0)$
$(3,3,0)$	1	3	3	1	2	2	$(3.35,3.35,1.0)$
$(4,0,0)$	1	1	2	3	3	2	$(4.1,0.8,0.8)$
$(4,1,0)$	1	1	3	3	2	2	$(4.1,1.6,1.0)$
$(4,2,0)$	1	1	3	3	2	2	$(4.1,2.6,1.0)$
$(4,3,0)$	1	1	3	3	2	2	$(4.1,2.6,1.0)$
$(4,4,0)$	1	3	3	1	2	2	$(4.35,4.35,1.0)$
$(5,0,0)$	1	1	2	3	3	2	$(5.1,0.8,0.8)$
$(5,1,0)$	1	1	3	3	2	2	$(5.1,1.6,1.0)$
$(5,2,0)$	1	1	3	3	2	2	$(5.1,2.6,1.0)$
$(5,3,0)$	1	1	3	3	2	2	$(5.1,3.6,1.0)$
$(5,4,0)$	1	1	3	3	2	2	$(5.1,4.6,1.0)$
$(5,5,0)$	1	3	3	1	2	2	$(5.35,5.35,1.0)$
$(6,0,0)$	1	1	2	3	3	2	$(6.1,0.8,0.8)$
$(6,1,0)$	1	1	3	3	2	2	$(6.1,1.6,1.0)$
$(6,2,0)$	1	1	3	3	2	2	$(6.1,2.6,1.0)$
$(6,3,0)$	1	1	3	3	2	2	$(6.1,3.6,1.0)$
$(6,4,0)$	1	1	3	3	2	2	$(6.1,4.6,1.0)$
$(6,5,0)$	1	1	3	3	2	2	$(6.1,5.6,1.0)$
$(6,6,0)$	1	3	3	1	2	2	$(6.35,6.35,1.0)$
$(7,0,0)$	1	1	2	3	3	2	$(7.1,0.8,0.8)$
$(7,1,0)$	1	1	3	3	2	2	$(7.1,1.6,1.0)$
$(7,2,0)$	1	1	3	3	2	2	$(7.1,2.6,1.0)$
$(7,3,0)$	1	1	3	3	2	2	$(7.1,3.6,1.0)$
$(7,4,0)$	1	1	3	3	2	2	$(7.1,4.6,1.0)$
$(7,5,0)$	1	1	3	3	2	2	$(7.1,5.6,1.0)$
$(7,6,0)$	1	1	3	3	2	2	$(7.1,6.6,1.0)$
$(7,7,0)$	1	1	3	3	2	2	$(7.1,7.6,1.0)$

Table E.9: The DP results for the third schedule of a small instance when the states are ordered lexicographically and the number of time blocks is not equal to the number of different tasks.

State	$p=1$		$p=2$		$p=3$		Expected new state
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$	
$(0,0,0)$	1	2	2	1	3	3	$(0.55,0.55,0.6)$
$(1,0,0)$	1	1	2	3	3	2	$(1.1,0.8,0.8)$
$(1,1,0)$	1	3	3	1	2	2	$(1.35,1.35,1.0)$
$(2,0,0)$	1	1	2	3	3	2	$(2.1,0.8,0.8)$
$(2,1,0)$	1	1	3	3	2	2	$(2.1,1.6,1.0)$
$(2,2,0)$	1	3	3	1	2	2	$(2.35,2.35,1.0)$
$(3,0,0)$	1	1	2	3	3	2	$(3.1,0.8,0.8)$
$(3,1,0)$	1	1	3	3	2	2	$(3.1,1.6,1.0)$
$(3,2,0)$	1	1	3	3	2	2	$(3.1,2.6,1.0)$
$(3,3,0)$	1	3	3	1	2	2	$(3.35,3.35,1.0)$
$(4,0,0)$	1	1	2	3	3	2	$(4.1,0.8,0.8)$
$(4,1,0)$	1	1	3	3	2	2	$(4.1,1.6,1.0)$
$(4,2,0)$	1	1	3	3	2	2	$(4.1,2.6,1.0)$
$(4,3,0)$	1	1	3	3	2	2	$(4.1,3.6,1.0)$
$(4,4,0)$	1	3	3	1	2	2	$(4.35,4.35,1.0)$
$(5,0,0)$	1	1	2	3	3	2	$(5.1,0.8,0.8)$
$(5,1,0)$	1	1	3	3	2	2	$(5.1,1.6,1.0)$
$(5,2,0)$	1	1	3	3	2	2	$(5.1,2.6,1.0)$
$(5,3,0)$	1	1	3	3	2	2	$(5.1,3.6,1.0)$
$(5,4,0)$	1	1	3	3	2	2	$(5.1,4.6,1.0)$
$(5,5,0)$	1	1	3	3	2	2	$(5.1,5.6,1.0)$
$(6,0,0)$	1	1	2	3	3	2	$(6.1,0.8,0.8)$
$(6,1,0)$	1	1	3	3	2	2	$(6.1,1.6,1.0)$
$(6,2,0)$	1	1	3	3	2	2	$(6.1,2.6,1.0)$
$(6,3,0)$	1	1	3	3	2	2	$(6.1,3.6,1.0)$
$(6,4,0)$	1	1	3	3	2	2	$(6.1,4.6,1.0)$
$(6,5,0)$	3	3	1	1	2	2	$(6.6,5.1,1.0)$
$(6,6,0)$	1	1	3	3	2	2	$(6.1,6.6,1.0)$

Table E.10: The DP results for the second schedule of a small instance when the states are ordered lexicographically and the number of time blocks is not equal to the number of different tasks.

State	$p=1$		$p=2$		$p=3$		Expected new state
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$	
$(0,0,0)$	1	2	2	1	3	3	$(0.55,0.55,0.6)$
$(1,0,0)$	1	1	2	3	3	2	$(1.1,0.8,0.8)$
$(1,1,0)$	1	3	3	1	2	2	$(1.35,1.35,1.0)$
$(2,0,0)$	1	1	2	3	3	2	$(2.1,0.8,0.8)$
$(2,1,0)$	1	1	3	3	2	2	$(2.1,1.6,1.0)$
$(2,2,0)$	1	3	3	1	2	2	$(2.35,2.35,1.0)$
$(3,0,0)$	1	1	2	3	3	2	$(3.1,0.8,0.8)$
$(3,1,0)$	1	1	3	3	2	2	$(3.1,1.6,1.0)$
$(3,2,0)$	1	1	3	3	2	2	$(3.1,2.6,1.0)$
$(3,3,0)$	1	3	3	1	2	2	$(3.35,3.35,1.0)$
$(4,0,0)$	1	1	2	3	3	2	$(4.1,0.8,0.8)$
$(4,1,0)$	1	1	3	3	2	2	$(4.1,1.6,1.0)$
$(4,2,0)$	1	1	3	3	2	2	$(4.1,2.6,1.0)$
$(4,3,0)$	1	1	3	3	2	2	$(4.1,3.6,1.0)$
$(4,4,0)$	1	1	3	3	2	2	$(4.1,4.6,1.0)$

Table E.11: The DP results for the first schedule of a small instance when the states are ordered lexicographically and the number of time blocks is not equal to the number of different tasks.

State	$p=1$		$p=2$		$p=3$		Expected new state
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$	
$(0,0,0)$	1	2	3	3	2	1	$(0.55, 0.6, 0.55)$

E.2.3 Different arrival rate per schedule

The results from the DP with three persons who can be scheduled over three tasks with two time blocks and four schedules are made, are shown in Tables E.12 (fourth schedule) – E.15 (first schedule). These tables show an optimal action given the state and decision epoch. A 1 corresponds to task 1 with λ_1, a 2 corresponds to task 2 with $\lambda_{2,d}$ and a 3 corresponds to task 3 with λ_3. The parameters used are shown below.

$$
\begin{aligned}
\lambda_1 &= 0.05 \\
\lambda_{2,d} &= \begin{cases} 0.5 & \text{if day } d \text{ is during schedule 1, 2 or 4} \\ 5.0 & \text{if day } d \text{ is during schedule 3} \end{cases} \\
\lambda_3 &= 0.3 \\
q &= 0.01 \\
N &= 5 \\
J &= 3 \\
M &= 3 \\
B &= 2
\end{aligned}
$$

Table E.12: The DP results for the fourth schedule of a small instance when the states are ordered lexicographically and the arrival rates per task can differ per schedule.

State	$p=1$		$p=2$		$p=3$		Expected new state
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$	
$(0,0,0)$	1	2	2	1	3	3	$(0.55,0.55,0.6)$
$(1,0,0)$	1	1	2	3	3	2	$(1.1,0.8,0.8)$
$(1,1,0)$	1	3	3	1	2	2	$(1.35,1.35,1.0)$
$(2,0,0)$	1	1	2	3	3	2	$(2.1,0.8,0.8)$
$(2,1,0)$	1	1	3	3	2	2	$(2.1,1.6,1.0)$
$(2,2,0)$	1	3	3	1	2	2	$(2.35,2.35,1.0)$
$(3,0,0)$	1	1	2	3	3	2	$(3.1,0.8,0.8)$
$(3,1,0)$	1	1	3	3	2	2	$(3.1,1.6,1.0)$
$(3,2,0)$	1	1	3	3	2	2	$(3.1,2.6,1.0)$
$(3,3,0)$	1	3	3	1	2	2	$(3.35,3.35,1.0)$
$(4,0,0)$	1	1	2	3	3	2	$(4.1,0.8,0.8)$
$(4,1,0)$	1	1	3	3	2	2	$(4.1,1.6,1.0)$
$(4,2,0)$	1	1	3	3	2	2	$(4.1,2.6,1.0)$
$(4,3,0)$	1	1	3	3	2	2	$(4.1,3.6,1.0)$
$(4,4,0)$	1	3	3	1	2	2	$(4.35,4.35,1.0)$
$(5,0,0)$	1	1	2	3	3	2	$(5.1,0.8,0.8)$
$(5,1,0)$	1	1	3	3	2	2	$(5.1,1.6,1.0)$
$(5,2,0)$	1	1	3	3	2	2	$(5.1,2.6,1.0)$
$(5,3,0)$	1	1	3	3	2	2	$(5.1,3.6,1.0)$
$(5,4,0)$	1	1	3	3	2	2	$(5.1,4.6,1.0)$
$(5,5,0)$	1	3	3	1	2	2	$(5.35,5.35,1.0)$
$(6,0,0)$	1	1	2	3	3	2	$(6.1,0.8,0.8)$
$(6,1,0)$	1	1	3	3	2	2	$(6.1,1.6,1.0)$
$(6,2,0)$	1	1	3	3	2	2	$(6.1,2.6,1.0)$
$(6,3,0)$	1	1	3	3	2	2	$(6.1,3.6,1.0)$
$(6,4,0)$	1	1	3	3	2	2	$(6.1,4.6,1.0)$
$(6,5,0)$	1	1	3	3	2	2	$(6.1,5.6,1.0)$
$(6,6,0)$	1	3	3	1	2	2	$(6.35,6.35,1.0)$
$(7,0,0)$	1	1	2	3	3	2	$(7.1,0.8,0.8)$
$(7,1,0)$	1	1	3	3	2	2	$(7.1,1.6,1.0)$
$(7,2,0)$	1	1	3	3	2	2	$(7.1,2.6,1.0)$
$(7,3,0)$	1	1	3	3	2	2	$(7.1,3.6,1.0)$
$(7,4,0)$	1	1	3	3	2	2	$(7.1,4.6,1.0)$
$(7,5,0)$	1	1	3	3	2	2	$(7.1,5.6,1.0)$
$(7,6,0)$	1	1	3	3	2	2	$(7.1,6.6,1.0)$
$(7,7,0)$	1	3	3	1	2	2	$(7.35,7.35,1.0)$
$(8,0,0)$	1	1	2	3	3	2	$(8.1,0.8,0.8)$
$(8,1,0)$	1	1	3	3	2	2	$(8.1,1.6,1.0)$

$(8,2,0)$	1	1	3	3	2	2	$(8.1,2.6,1.0)$
$(8,3,0)$	1	1	3	3	2	2	$(8.1,3.6,1.0)$
$(8,4,0)$	1	1	3	3	2	2	$(8.1,4.6,1.0)$
$(8,5,0)$	1	1	3	3	2	2	$(8.1,5.6,1.0)$
$(8,6,0)$	1	1	3	3	2	2	$(8.1,6.6,1.0)$
$(8,7,0)$	1	1	3	3	2	2	$(8.1,7.6,1.0)$
$(8,8,0)$	1	3	3	1	2	2	$(8.35,8.35,1.0)$
$(9,0,0)$	1	1	2	3	3	2	$(9.1,0.8,0.8)$
$(9,1,0)$	1	1	3	3	2	2	$(9.1,1.6,1.0)$
$(9,2,0)$	1	1	3	3	2	2	$(9.1,2.6,1.0)$
$(9,3,0)$	1	1	3	3	2	2	$(9.1,3.6,1.0)$
$(9,4,0)$	1	1	3	3	2	2	$(9.1,4.6,1.0)$
$(9,5,0)$	1	1	3	3	2	2	$(9.1,5.6,1.0)$
$(9,6,0)$	1	1	3	3	2	2	$(9.1,6.6,1.0)$
$(9,7,0)$	1	1	3	3	2	2	$(9.1,7.6,1.0)$
$(9,8,0)$	1	1	3	3	2	2	$(9.1,8.6,1.0)$
$(9,9,0)$	1	3	3	1	2	2	$(9.35,9.35,1.0)$
$(10,0,0)$	1	1	2	3	3	2	$(10.1,0.8,0.8)$
$(10,1,0)$	1	1	3	3	2	2	$(10.1,1.6,1.0)$
$(10,2,0)$	1	1	3	3	2	2	$(10.1,2.6,1.0)$
$(10,3,0)$	1	1	3	3	2	2	$(10.1,3.6,1.0)$
$(10,4,0)$	1	1	3	3	2	2	$(10.1,4.6,1.0)$
$(10,5,0)$	1	1	3	3	2	2	$(10.1,5.6,1.0)$
$(10,6,0)$	1	1	3	3	2	2	$(10.1,6.6,1.0)$
$(10,7,0)$	1	1	3	3	2	2	$(10.1,7.6,1.0)$
$(10,8,0)$	1	1	3	3	2	2	$(10.1,8.6,1.0)$
$(10,9,0)$	1	1	3	3	2	2	$(10.1,9.6,1.0)$
$(10,10,0)$	1	3	3	1	2	2	$(10.35,10.35,1.0)$
$(11,0,0)$	1	1	2	3	3	2	$(11.1,0.8,0.8)$
$(11,1,0)$	1	1	3	3	2	2	$(11.1,1.6,1.0)$
$(11,2,0)$	1	1	3	3	2	2	$(11.1,2.6,1.0)$
$(11,3,0)$	1	1	3	3	2	2	$(11.1,3.6,1.0)$
$(11,4,0)$	1	1	3	3	2	2	$(11.1,4.6,1.0)$
$(11,5,0)$	1	1	3	3	2	2	$(11.1,5.6,1.0)$
$(11,6,0)$	1	1	3	3	2	2	$(11.1,6.6,1.0)$
$(11,7,0)$	1	1	3	3	2	2	$(11.1,7.6,1.0)$
$(11,8,0)$	1	1	3	3	2	2	$(11.1,8.6,1.0)$
$(11,9,0)$	1	1	3	3	2	2	$(11.1,9.6,1.0)$
$(11,10,0)$	1	1	3	3	2	2	$(11.1,10.6,1.0)$
$(11,11,0)$	1	3	3	1	2	2	$(11.35,11.35,1.0)$
$(12,0,0)$	1	1	2	3	3	2	$(12.1,0.8,0.8)$
$(12,1,0)$	1	1	3	3	2	2	$(12.1,1.6,1.0)$
$(12,2,0)$	1	1	3	3	2	2	$(12.1,2.6,1.0)$
$(12,3,0)$	1	1	3	3	2	2	$(12.1,3.6,1.0)$
$(12,4,0)$	1	1	3	3	2	2	$(12.1,4.6,1.0)$
$(12,5,0)$	1	1	3	3	2	2	$(12.1,5.6,1.0)$
$(12,6,0)$	1	1	3	3	2	2	$(12.1,6.6,1.0)$
$(12,7,0)$	1	1	3	3	2	2	$(12.1,7.6,1.0)$
$(12,8,0)$	1	1	3	3	2	2	$(12.1,8.6,1.0)$
$(12,9,0)$	1	1	3	3	2	2	$(12.1,9.6,1.0)$
$(12,10,0)$	1	1	3	3	2	2	$(12.1,10.6,1.0)$
$(12,11,0)$	1	1	3	3	2	2	$(12.1,11.6,1.0)$
$(12,12,0)$	1	3	3	1	2	2	$(12.35,12.35,1.0)$
$(13,0,0)$	1	1	2	3	3	2	$(13.1,0.8,0.8)$
$(13,1,0)$	1	1	3	3	2	2	$(13.1,1.6,1.0)$
$(13,2,0)$	1	1	3	3	2	2	$(13.1,2.6,1.0)$
$(13,3,0)$	1	1	3	3	2	2	$(13.1,3.6,1.0)$
$(13,4,0)$	1	1	3	3	2	2	$(13.1,4.6,1.0)$
$(13,5,0)$	1	1	3	3	2	2	$(13.1,5.6,1.0)$
$(13,6,0)$	1	1	3	3	2	2	$(13.1,6.6,1.0)$
$(13,7,0)$	1	1	3	3	2	2	$(13.1,7.6,1.0)$
$(13,8,0)$	1	1	3	3	2	2	$(13.1,8.6,1.0)$
$(13,9,0)$	1	1	3	3	2	2	$(13.1,9.6,1.0)$
$(13,10,0)$	1	1	3	3	2	2	$(13.1,10.6,1.0)$

$(13,11,0)$	1	1	3	3	2	2	$(13.1, 11.6, 1.0)$
$(13,12,0)$	1	1	3	3	2	2	$(13.1, 12.6, 1.0)$
$(13,13,0)$	1	3	3	1	2	2	$(13.35, 13.35, 1.0)$
$(14,0,0)$	1	1	2	3	3	2	$(14.1, 0.8, 0.8)$
$(14,1,0)$	1	1	3	3	2	2	$(14.1, 1.6, 1.0)$
$(14,2,0)$	1	1	3	3	2	2	$(14.1, 2.6, 1.0)$
$(14,3,0)$	1	1	3	3	2	2	$(14.1, 3.6, 1.0)$
$(14,4,0)$	1	1	3	3	2	2	$(14.1, 4.6, 1.0)$
$(14,5,0)$	1	1	3	3	2	2	$(14.1, 5.6, 1.0)$
$(14,6,0)$	1	1	3	3	2	2	$(14.1, 6.6, 1.0)$
$(14,7,0)$	1	1	3	3	2	2	$(14.1, 7.6, 1.)$
$(14,8,0)$	1	1	3	3	2	2	$(14.1, 8.6, 1.0)$
$(14,9,0)$	1	1	3	3	2	2	$(14.1, 9.6, 1.0)$
$(14,10,0)$	1	1	3	3	2	2	$(14.1, 10.6, 1.0)$
$(14,11,0)$	1	1	3	3	2	2	$(14.1, 11.6, 1.0)$
$(14,12,0)$	1	1	3	3	2	2	$(14.1, 12.6, 1.0)$
$(14,13,0)$	1	1	3	3	2	2	$(14.1, 13.6, 1.0)$
$(14,14,0)$	1	3	3	1	2	2	$(14.35, 14.35, 1.0)$
$(15,0,0)$	1	1	2	3	3	2	$(15.1, 0.8, 0.8)$
$(15,1,0)$	1	1	3	3	2	2	$(15.1, 1.6, 1.0)$
$(15,2,0)$	1	1	3	3	2	2	$(15.1, 2.6, 1.0)$
$(15,3,0)$	1	1	3	3	2	2	$(15.1, 3.6, 1.0)$
$(15,4,0)$	1	1	3	3	2	2	$(15.1, 4.6, 1.0)$
$(15,5,0)$	1	1	3	3	2	2	$(15.1, 5.6, 1.0)$
$(15,6,0)$	1	1	3	3	2	2	$(15.1, 6.6, 1.0)$
$(15,7,0)$	1	1	3	3	2	2	$(15.1, 7.6, 1.0)$
$(15,8,0)$	1	1	3	3	2	2	$(15.1, 8.6, 1.0)$
$(15,9,0)$	1	1	3	3	2	2	$(15.1, 9.6, 1.0)$
$(15,10,0)$	1	1	3	3	2	2	$(15.1, 10.6, 1.0)$
$(15,11,0)$	1	1	3	3	2	2	$(15.1, 11.6, 1.0)$
$(15,12,0)$	1	1	3	3	2	2	$(15.1, 12.6, 1.0)$
$(15,13,0)$	1	1	3	3	2	2	$(15.1, 13.6, 1.0)$
$(15,14,0)$	1	1	3	3	2	2	$(15.1, 14.6, 1.0)$
$(15,15,0)$	1	3	3	1	2	2	$(15.35, 15.35, 1.0)$
$(16,0,0)$	1	1	2	3	3	2	$(16.1, 0.8, 0.8)$
$(16,1,0)$	1	1	3	3	2	2	$(16.1, 1.6, 1.0)$
$(16,2,0)$	1	1	3	3	2	2	$(16.1, 2.6, 1.0)$
$(16,3,0)$	1	1	3	3	2	2	$(16.1, 3.6, 1.0)$
$(16,4,0)$	1	1	3	3	2	2	$(16.1, 4.6, 1.0)$
$(16,5,0)$	1	1	3	3	2	2	$(16.1, 5.6, 1.0)$
$(16,6,0)$	1	1	3	3	2	2	$(16.1, 6.6, 1.0)$
$(16,7,0)$	1	1	3	3	2	2	$(16.1, 7.6, 1.0)$
$(16,8,0)$	1	1	3	3	2	2	$(16.1, 8.6, 1.0)$
$(16,9,0)$	1	1	3	3	2	2	$(16.1, 9.6, 1.0)$
$(16,10,0)$	1	1	3	3	2	2	$(16.1, 10.6, 1.0)$
$(16,11,0)$	1	1	3	3	2	2	$(16.1, 11.6, 1.0)$
$(16,12,0)$	1	1	3	3	2	2	$(16.1, 12.6, 1.0)$
$(16,13,0)$	1	1	3	3	2	2	$(16.1, 13.6, 1.0)$
$(16,14,0)$	1	1	3	3	2	2	$(16.1, 14.6, 1.0)$
$(16,15,0)$	1	1	3	3	2	2	$(16.1, 15.6, 1.0)$
$(16,16,0)$	1	3	3	1	2	2	$(16.35, 16.35, 1.0)$
$(17,0,0)$	1	1	2	3	3	2	$(17.1, 0.8, 0.8)$
$(17,1,0)$	1	1	3	3	2	2	$(17.1, 1.6, 1.0)$
$(17,2,0)$	1	1	3	3	2	2	$(17.1, 2.6, 1.0)$
$(17,3,0)$	1	1	3	3	2	2	$(17.1, 3.6, 1.0)$
$(17,4,0)$	1	1	3	3	2	2	$(17.1, 4.6, 1.0)$
$(17,5,0)$	1	1	3	3	2	2	$(17.1, 5.6, 1.0)$
$(17,6,0)$	1	1	3	3	2	2	$(17.1, 6.6, 1.0)$
$(17,7,0)$	1	1	3	3	2	2	$(17.1, 7.6, 1.0)$
$(17,8,0)$	1	1	3	3	2	2	$(17.1, 8.6, 1.0)$
$(17,9,0)$	1	1	3	3	2	2	$(17.1, 9.6, 1.0)$
$(17,10,0)$	1	1	3	3	2	2	$(17.1, 10.6, 1.0)$
$(17,11,0)$	1	1	3	3	2	2	$(17.1, 11.6, 1.0)$
$(17,12,0)$	1	1	3	3	2	2	$(17.1, 12.6, 1.0)$

$(17,13,0)$	1	1	3	3	2	2	$(17.1,13.6,1.0)$
$(17,14,0)$	1	1	3	3	2	2	$(17.1,14.6,1.0)$
$(17,15,0)$	1	1	3	3	2	2	$(17.1,15.6,1.0)$
$(17,16,0)$	1	1	3	3	2	2	$(17.1,16.6,1.0)$
$(17,17,0)$	1	3	3	1	2	2	$(17.35,17.35,1.0)$
$(18,0,0)$	1	1	2	3	3	2	$(18.1,0.8,0.8)$
$(18,1,0)$	1	1	3	3	2	2	$(18.1,1.6,1.0)$
$(18,2,0)$	1	1	3	3	2	2	$(18.1,2.6,1.0)$
$(18,3,0)$	1	1	3	3	2	2	$(18.1,3.6,1.0)$
$(18,4,0)$	1	1	3	3	2	2	$(18.1,4.6,1.0)$
$(18,5,0)$	1	1	3	3	2	2	$(18.1,5.6,1.0)$
$(18,6,0)$	1	1	3	3	2	2	$(18.1,6.6,1.0)$
$(18,7,0)$	1	1	3	3	2	2	$(18.1,7.6,1.0)$
$(18,8,0)$	1	1	3	3	2	2	$(18.1,8.6,1.0)$
$(18,9,0)$	1	1	3	3	2	2	$(18.1,9.6,1.0)$
$(18,10,0)$	1	1	3	3	2	2	$(18.1,10.6,1.0)$
$(18,11,0)$	1	1	3	3	2	2	$(18.1,11.6,1.0)$
$(18,12,0)$	1	1	3	3	2	2	$(18.1,12.6,1.0)$
$(18,13,0)$	1	1	3	3	2	2	$(18.1,13.6,1.0)$
$(18,14,0)$	1	1	3	3	2	2	$(18.1,14.6,1.0)$
$(18,15,0)$	1	1	3	3	2	2	$(18.1,15.6,1.0)$
$(18,16,0)$	1	1	3	3	2	2	$(18.1,16.6,1.0)$
$(18,17,0)$	1	1	3	3	2	2	$(18.1,17.6,1.0)$
$(18,18,0)$	1	3	3	1	2	2	$(18.35,18.35,1.0)$
$(19,0,0)$	1	1	2	3	3	2	$(19.1,0.8,0.8)$
$(19,1,0)$	1	1	3	3	2	2	$(19.1,1.6,1.0)$
$(19,2,0)$	1	1	3	3	2	2	$(19.1,2.6,1.0)$
$(19,3,0)$	1	1	3	3	2	2	$(19.1,3.6,1.0)$
$(19,4,0)$	1	1	3	3	2	2	$(19.1,4.6,1.0)$
$(19,5,0)$	1	1	3	3	2	2	$(19.1,5.6,1.0)$
$(19,6,0)$	1	1	3	3	2	2	$(19.1,6.6,1.0)$
$(19,7,0)$	1	1	3	3	2	2	$(19.1,7.6,1.0)$
$(19,8,0)$	1	1	3	3	2	2	$(19.1,8.6,1.0)$
$(19,9,0)$	1	1	3	3	2	2	$(19.1,9.6,1.0)$
$(19,10,0)$	1	1	3	3	2	2	$(19.1,10.6,1.0)$
$(19,11,0)$	1	1	3	3	2	2	$(19.1,11.6,1.0)$
$(19,12,0)$	1	1	3	3	2	2	$(19.1,12.6,1.0)$
$(19,13,0)$	1	1	3	3	2	2	$(19.1,13.6,1.0)$
$(19,14,0)$	1	1	3	3	2	2	$(19.1,14.6,1.0)$
$(19,15,0)$	1	1	3	3	2	2	$(19.1,15.6,1.0)$
$(19,16,0)$	1	1	3	3	2	2	$(19.1,16.6,1.0)$
$(19,17,0)$	1	1	3	3	2	2	$(19.1,17.6,1.0)$
$(19,18,0)$	1	1	3	3	2	2	$(19.1,18.6,1.0)$
$(19,19,0)$	1	1	3	3	2	2	$(19.1,19.6,1.0)$

Table E.13: The DP results for the third schedule of a small instance when the states are ordered lexicographically and the arrival rates per task can differ per schedule.

State	$p=1$		$p=2$		$p=3$		Expected new state
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$	
$(0,0,0)$	1	2	2	1	3	2	$(5.05,5.05,0.6)$
$(1,0,0)$	3	3	1	2	2	1	$(1.6,5.05,5.05)$
$(1,1,0)$	1	2	3	3	2	1	$(6.05,1.6,5.05)$
$(2,0,0)$	3	3	1	2	2	1	$(2.6,5.05,5.05)$
$(2,1,0)$	3	3	1	2	2	1	$(2.6,6.05,5.05)$
$(2,2,0)$	3	3	1	2	2	1	$(2.6,7.05,5.05)$
$(3,0,0)$	3	3	1	2	2	1	$(3.6,5.05,5.05)$
$(3,1,0)$	3	3	1	2	2	1	$(3.6,6.05,5.05)$
$(3,2,0)$	3	3	1	2	2	1	$(3.6,7.05,5.05)$
$(3,3,0)$	3	3	1	2	2	1	$(3.6,8.05,5.05)$
$(4,0,0)$	3	3	1	2	2	1	$(4.6,5.05,5.05)$
$(4,1,0)$	3	3	1	2	2	1	$(4.6,6.05,5.05)$

$(4,2,0)$	3	3	1	2	2	1	$(4.6, 7.05, 5.05)$
$(4,3,0)$	3	3	1	2	2	1	$(4.6, 8.05, 5.05)$
$(4,4,0)$	1	2	3	3	2	1	$(9.05, 4.6, 5.05)$
$(5,0,0)$	1	1	2	3	3	2	$(5.1, 5.3, 5.3)$
$(5,1,0)$	1	3	2	1	3	2	$(5.35, 6.05, 5.3)$
$(5,2,0)$	1	3	2	1	3	2	$(5.35, 7.05, 5.3)$
$(5,3,0)$	1	3	2	1	3	2	$(5.35, 8.05, 5.3)$
$(5,4,0)$	1	3	2	1	3	2	$(5.35, 9.05, 5.3)$
$(5,5,0)$	1	3	3	1	2	2	$(5.35, 5.35, 10.0)$
$(6,0,0)$	1	1	2	3	3	2	$(6.1, 5.3, 5.3)$
$(6,1,0)$	1	3	2	1	3	2	$(6.35, 6.05, 5.3)$
$(6,2,0)$	1	3	2	1	3	2	$(6.35, 7.05, 5.3)$
$(6,3,0)$	1	3	2	1	3	2	$(6.35, 8.05, 5.3)$
$(6,4,0)$	1	3	2	1	3	2	$(6.35, 9.05, 5.3)$
$(6,5,0)$	1	1	3	3	2	2	$(6.1, 5.6, 10.0)$
$(6,6,0)$	1	3	3	1	2	2	$(6.35, 6.35, 10.0)$

Table E.14: The DP results for the second schedule of a small instance when the states are ordered lexicographically and the arrival rates per task can differ per schedule.

State	$p=1$		$p=2$		$p=3$		Expected new state
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$	
$(0,0,0)$	1	1	2	2	3	3	$(0.1, 1.0, 0.6)$
$(1,0,0)$	2	2	1	1	3	3	$(2.0, 0.1, 0.6)$
$(1,1,0)$	2	2	3	3	1	1	$(2.0, 1.6, 0.1)$
$(2,0,0)$	2	2	1	3	3	1	$(3.0, 0.35, 0.35)$
$(2,1,0)$	2	2	1	1	3	3	$(3.0, 1.1, 0.6)$
$(2,2,0)$	2	2	3	3	1	1	$(3.0, 2.6, 0.1)$
$(3,0,0)$	2	2	1	3	3	1	$(4.0, 0.35, 0.35)$
$(3,1,0)$	2	2	1	1	3	3	$(4.0, 1.1, 0.6)$
$(3,2,0)$	2	2	1	1	3	3	$(4.0, 2.1, 0.6)$
$(3,3,0)$	3	3	2	2	1	1	$(3.1, 4.0, 0.6)$
$(4,0,0)$	2	2	1	3	3	1	$(5.0, 0.35, 0.35)$
$(4,1,0)$	2	2	1	1	3	3	$(5.0, 1.1, 0.6)$
$(4,2,0)$	2	2	1	1	3	3	$(5.0, 2.1, 0.6)$
$(4,3,0)$	2	2	1	1	3	3	$(5.0, 3.1, 0.6)$
$(4,4,0)$	1	1	2	2	3	3	$(4.1, 5.0, 0.6)$

Table E.15: The DP results for the first schedule of a small instance when the states are ordered lexicographically and the arrival rates per task can differ per schedule.

State	$p=1$		$p=2$		$p=3$		Expected new state
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$	
$(0,0,0)$	1	1	3	3	2	2	$(0.1, 0.6, 1.0)$

E.2.4 Different arrival rates last schedule

The results from the DP with three persons who can be scheduled over three tasks with two time blocks and when five schedules are made, are shown in Tables E.16 (fourth schedule) – E.19 (first schedule). These tables show an optimal action given the state and decision epoch. A 1 corresponds to task 1 with λ_1, a 2 corresponds to task 2 with $\lambda_{2,d}$ and a 3 corresponds to task 3 with λ_3. The parameters used are shown below.

$$\begin{aligned}
\lambda_1 &= 0.05 \\
\lambda_{2,d} &= \begin{cases} 0.5 & \text{if day } d \text{ is during schedule 1, 2, 3 or 4} \\ 5.0 & \text{if day } d \text{ is during schedule 5} \end{cases} \\
\lambda_3 &= 0.3 \\
q &= 0.01 \\
N &= 6 \\
J &= 3 \\
M &= 3 \\
B &= 2
\end{aligned}$$

Table E.16: The DP results for the fourth schedule of a small instance when the states are ordered lexicographically and the arrival rates per task can differ in the last schedule.

State	$p=1$		$p=2$		$p=3$		Expected new state	Variance of the schedule
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$		
$(0,0,0)$	3	3	2	2	1	1	$(0.6,1.0,0.1)$	$(0.6,1.0,0.1)$
$(1,0,0)$	2	2	1	1	3	3	$(2.0,0.1,0.6)$	$(1.0,0.1,0.6)$
$(1,1,0)$	2	2	3	3	1	1	$(2.0,1.6,0.1)$	$(1.0,0.6,0.1)$
$(2,0,0)$	2	2	1	3	3	1	$(3.0,0.35,0.35)$	$(1.0,0.35,0.35)$
$(2,1,0)$	2	2	1	1	3	3	$(3.0,1.1,0.6)$	$(1.0,0.1,0.6)$
$(2,2,0)$	2	2	3	3	1	1	$(3.0,2.6,0.1)$	$(1.0,0.6,0.1)$
$(3,0,0)$	2	2	1	3	3	1	$(4.0,0.35,0.35)$	$(1.0,0.35,0.35)$
$(3,1,0)$	2	2	1	1	3	3	$(4.0,1.1,0.6)$	$(1.0,0.1,0.6)$
$(3,2,0)$	2	2	1	1	3	3	$(4.0,2.1,0.6)$	$(1.0,0.1,0.6)$
$(3,3,0)$	1	1	2	2	3	3	$(3.1,4.0,0.6)$	$(0.1,1.0,0.6)$
$(4,0,0)$	2	2	1	3	3	1	$(5.0,0.35,0.35)$	$(1.0,0.35,0.35)$
$(4,1,0)$	2	2	1	1	3	3	$(5.0,1.1,0.6)$	$(1.0,0.1,0.6)$
$(4,2,0)$	2	2	1	1	3	3	$(5.0,2.1,0.6)$	$(1.0,0.1,0.6)$
$(4,3,0)$	2	2	1	1	3	3	$(5.0,3.1,0.6)$	$(1.0,0.1,0.6)$
$(4,4,0)$	1	1	2	2	3	3	$(4.1,5.0,0.6)$	$(0.1,1.0,0.6)$
$(5,0,0)$	1	2	2	1	3	3	$(5.55,0.55,0.6)$	$(0.55,0.55,0.6)$
$(5,1,0)$	2	3	1	1	3	2	$(5.8,1.1,0.8)$	$(0.8,0.1,0.8)$
$(5,2,0)$	3	3	1	1	2	2	$(5.6,2.1,1.0)$	$(0.6,0.1,1.0)$
$(5,3,0)$	3	3	1	1	2	2	$(5.6,3.1,1.0)$	$(0.6,0.1,1.0)$
$(5,4,0)$	3	3	1	1	2	2	$(5.6,4.1,1.0)$	$(0.6,0.1,1.0)$
$(5,5,0)$	2	3	3	2	1	1	$(5.8,5.8,0.1)$	$(0.8,0.8,0.1)$
$(6,0,0)$	1	1	2	3	3	2	$(6.1,0.8,0.8)$	$(0.1,0.8,0.8)$
$(6,1,0)$	1	3	3	1	2	2	$(6.35,1.35,1.0)$	$(0.35,0.35,1.0)$
$(6,2,0)$	3	3	1	1	2	2	$(6.6,2.1,1.0)$	$(0.6,0.1,1.0)$
$(6,3,0)$	3	3	1	1	2	2	$(6.6,3.1,1.0)$	$(0.6,0.1,1.0)$
$(6,4,0)$	3	3	1	1	2	2	$(6.6,4.1,1.0)$	$(0.6,0.1,1.0)$
$(6,5,0)$	3	3	2	2	1	1	$(6.6,6.0,0.1)$	$(0.6,1.0,0.1)$
$(6,6,0)$	2	3	3	2	1	1	$(6.8,6.8,0.1)$	$(0.8,0.8,0.1)$
$(7,0,0)$	1	1	2	3	3	2	$(7.1,0.8,0.8)$	$(0.1,0.8,0.8)$
$(7,1,0)$	1	1	3	3	2	2	$(7.1,1.6,1.0)$	$(0.1,0.6,1.0)$
$(7,2,0)$	1	3	3	1	2	2	$(7.35,2.35,1.0)$	$(0.35,0.35,1.0)$
$(7,3,0)$	3	3	1	1	2	2	$(7.6,3.1,1.0)$	$(0.6,0.1,1.0)$
$(7,4,0)$	3	3	1	1	2	2	$(7.6,4.1,1.0)$	$(0.6,0.1,1.0)$
$(7,5,0)$	3	3	2	2	1	1	$(7.6,6.0,0.1)$	$(0.6,1.0,0.1)$
$(7,6,0)$	3	3	2	2	1	1	$(7.6,7.0,0.1)$	$(0.6,1.0,0.1)$
$(7,7,0)$	2	3	3	2	1	1	$(7.8,7.8,0.1)$	$(0.8,0.8,0.1)$

Table E.17: The DP results for the third schedule of a small instance when the states are ordered lexicographically and the arrival rates per task can differ in the last schedule.

State	$p=1$		$p=2$		$p=3$		Expected new state	Variance of the schedule
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$		
$(0,0,0)$	3	3	2	2	1	1	$(0.6,1.0,0.1)$	$(0.6,1.0,0.1)$
$(1,0,0)$	2	2	3	2	1	1	$(2.0,0.6,0.1)$	$(1.0,0.6,0.1)$
$(1,1,0)$	2	2	3	3	1	1	$(2.0,1.6,0.1)$	$(1.0,0.6,0.1)$
$(2,0,0)$	2	2	1	3	3	1	$(3.0,0.35,0.35)$	$(1.0,0.35,0.35)$
$(2,1,0)$	2	2	1	1	3	3	$(3.0,1.1,0.6)$	$(1.0,0.1,0.6)$
$(2,2,0)$	1	1	2	2	3	3	$(2.1,3.0,0.6)$	$(0.1,1.0,0.6)$
$(3,0,0)$	2	2	1	3	3	1	$(4.0,0.35,0.35)$	$(1.0,0.35,0.35)$
$(3,1,0)$	2	2	1	1	3	3	$(4.0,1.1,0.6)$	$(1.0,0.1,0.6)$
$(3,2,0)$	2	2	1	1	3	3	$(4.0,2.1,0.6)$	$(1.0,0.6,0.1)$
$(3,3,0)$	1	1	2	2	3	3	$(3.1,4.0,0.6)$	$(0.1,1.0,0.6)$
$(4,0,0)$	2	2	1	3	3	1	$(5.0,0.35,0.35)$	$(1.0,0.35,0.35)$
$(4,1,0)$	2	2	1	1	3	3	$(5.0,1.1,0.6)$	$(1.0,0.1,0.6)$
$(4,2,0)$	2	2	1	1	3	3	$(5.0,2.1,0.6)$	$(1.0,0.1,0.6)$
$(4,3,0)$	2	2	1	1	3	3	$(5.0,3.1,0.6)$	$(1.0,0.1,0.6)$
$(4,4,0)$	1	1	2	2	3	3	$(4.1,5.0,0.6)$	$(0.1,1.0,0.6)$
$(5,0,0)$	3	3	1	2	2	1	$(5.6,0.55,0.55)$	$(0.6,0.55,0.55)$
$(5,1,0)$	2	2	1	1	3	3	$(6.0,1.1,0.6)$	$(1.0,0.1,0.6)$
$(5,2,0)$	2	3	1	1	3	2	$(5.8,2.1,0.8)$	$(0.8,0.1,0.8)$
$(5,3,0)$	2	3	1	1	3	2	$(5.8,3.1,0.8)$	$(0.8,0.1,0.8)$
$(5,4,0)$	2	3	1	1	3	2	$(5.8,4.1,0.8)$	$(0.8,0.1,0.8)$
$(5,5,0)$	2	3	3	2	1	1	$(5.8,5.8,0.1)$	$(0.8,0.8,0.1)$
$(6,0,0)$	1	1	2	3	3	2	$(6.1,0.8,0.8)$	$(0.1,0.8,0.8)$
$(6,1,0)$	3	3	1	1	2	2	$(6.6,1.1,1.0)$	$(0.6,0.1,1.0)$
$(6,2,0)$	3	3	1	1	2	2	$(6.6,2.1,1.0)$	$(0.6,0.1,1.0)$
$(6,3,0)$	3	3	1	1	2	2	$(6.6,3.1,1.0)$	$(0.6,0.1,1.0)$
$(6,4,0)$	3	3	1	1	2	2	$(6.6,4.1,1.0)$	$(0.6,0.1,1.0)$
$(6,5,0)$	3	3	2	2	1	1	$(6.6,6.0,0.1)$	$(0.6,1.0,0.1)$
$(6,6,0)$	2	3	3	2	1	1	$(6.8,6.8,0.1)$	$(0.8,0.8,0.1)$

Table E.18: The DP results for the second schedule of a small instance when the states are ordered lexicographically and the arrival rates per task can differ in the last schedule.

State	$p=1$		$p=2$		$p=3$		Expected new state	Variance of the schedule
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$		
$(0,0,0)$	2	2	3	3	1	1	$(1.0,0.6,0.1)$	$(1.0,0.6,0.1)$
$(1,0,0)$	2	2	1	1	3	3	$(2.0,0.1,0.6)$	$(1.0,0.1,0.6)$
$(1,1,0)$	2	2	3	3	1	1	$(2.0,1.6,0.1)$	$(1.0,0.6,0.1)$
$(2,0,0)$	2	2	1	3	3	1	$(3.0,0.35,0.35)$	$(1.0,0.35,0.35)$
$(2,1,0)$	2	2	1	1	3	3	$(3.0,1.1,0.6)$	$(1.0,0.1,0.6)$
$(2,2,0)$	1	1	2	2	3	3	$(2.1,3.0,0.6)$	$(0.1,1.0,0.6)$
$(3,0,0)$	2	2	1	3	3	1	$(4.0,0.35,0.35)$	$(1.0,0.35,0.35)$
$(3,1,0)$	2	2	1	1	3	3	$(4.0,1.1,0.6)$	$(1.0,0.1,0.6)$
$(3,2,0)$	2	2	1	1	3	3	$(4.0,2.1,0.6)$	$(1.0,0.1,0.6)$
$(3,3,0)$	1	1	2	2	3	3	$(3.1,4.0,0.6)$	$(0.1,1.0,0.6)$
$(4,0,0)$	2	2	1	3	3	1	$(5.0,0.35,0.35)$	$(1.0,0.35,0.35)$
$(4,1,0)$	2	2	1	1	3	3	$(5.0,1.1,0.6)$	$(1.0,0.1,0.6)$
$(4,2,0)$	2	2	1	1	3	3	$(5.0,2.1,0.6)$	$(1.0,0.1,0.6)$
$(4,3,0)$	2	2	1	1	3	3	$(5.0,3.1,0.6)$	$(1.0,0.1,0.6)$
$(4,4,0)$	1	1	2	2	3	3	$(4.1,5.0,0.6)$	$(0.1,1.0,0.6)$

Table E.19: The DP results for the first schedule of a small instance when the states are ordered lexicographically and the arrival rates per task can differ in the last schedule.

State	$p=1$		$p=2$		$p=3$		Expected new state	Variance of the schedule
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$		
$(0,0,0)$	3	3	2	2	1	1	$(0.6,1.0,0.1)$	$(0.6,1.0,0.1)$

E.2.5 Bigger differences arrival rates

The results from the DP with three persons who can be scheduled over three tasks with two time blocks and when three schedules are made, are shown in Tables E.20 (third schedule) – E.22 (first schedule). These tables show an optimal action given the state and decision epoch. A 1 corresponds to task one with λ_1, a 2 corresponds to task two with $\lambda_{2,d}$ and a 3 corresponds to task three with λ_3.

$$\begin{aligned} \lambda_1 &= 0.1 \\ \lambda_{2,d} &= 3 \\ \lambda_3 &= 1 \\ q &= 0.01 \\ N &= 4 \\ P &= 3 \\ B &= 2 \end{aligned}$$

Table E.20: The DP results for the third schedule of a small instance with bigger differences arrival rates.

State	$p=1$		$p=2$		$p=3$		Expected new state	Variance of the schedule
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$		
$(0,0,0)$	1	2	2	1	3	3	$(3.1,3.1,2.0)$	$(3.1,3.1,2.0)$
$(1,0,0)$	3	3	1	2	2	1	$(3.0,3.1,3.1)$	$(2.0,3.1,3.1)$
$(1,1,0)$	1	2	3	3	2	1	$(4.1,3.0,3.1)$	$(3.1,2.0,3.1)$
$(2,0,0)$	1	3	2	1	3	2	$(3.1,3.1,4.0)$	$(1.1,3.1,4.0)$
$(2,1,0)$	1	3	2	1	3	2	$(3.1,4.1,4.0)$	$(1.1,3.1,4.0)$
$(2,2,0)$	1	2	3	1	2	3	$(5.1,3.1,4.0)$	$(3.1,1.1,4.0)$
$(3,0,0)$	1	1	2	3	3	2	$(3.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(3,1,0)$	1	3	2	1	3	2	$(4.1,4.1,4.0)$	$(1.1,3.1,4.0)$
$(3,2,0)$	1	3	2	1	3	2	$(4.1,5.1,4.0)$	$(1.1,3.1,4.0)$
$(3,3,0)$	1	3	3	1	2	2	$(4.1,4.1,6.0)$	$(1.1,1.1,6.0)$
$(4,0,0)$	1	1	2	3	3	2	$(4.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(4,1,0)$	1	1	2	3	3	2	$(4.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(4,2,0)$	1	3	2	1	3	2	$(5.1,5.1,4.0)$	$(1.1,3.1,4.0)$
$(4,3,0)$	1	1	3	3	2	2	$(4.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(4,4,0)$	1	3	3	1	2	2	$(5.1,5.1,6.0)$	$(1.1,1.1,6.0)$
$(5,0,0)$	1	1	2	3	3	2	$(5.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(5,1,0)$	1	1	2	3	3	2	$(5.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(5,2,0)$	1	1	3	3	2	2	$(5.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(5,3,0)$	1	1	3	3	2	2	$(5.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(5,4,0)$	1	1	3	3	2	2	$(5.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(5,5,0)$	1	3	3	1	2	2	$(6.1,6.1,6.0)$	$(1.1,1.1,6.0)$
$(6,0,0)$	1	1	2	3	3	2	$(6.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(6,1,0)$	1	1	2	3	3	2	$(6.2,5.0,4.0)$	$(0.1,4.0,4.0)$
$(6,2,0)$	1	1	3	3	2	2	$(6.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(6,3,0)$	1	1	3	3	2	2	$(6.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(6,4,0)$	1	1	3	3	2	2	$(6.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(6,5,0)$	1	1	3	3	2	2	$(6.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(6,6,0)$	1	3	3	1	2	2	$(7.1,7.1,6.0)$	$(1.1,1.1,6.0)$
$(7,0,0)$	1	1	2	3	3	2	$(7.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(7,1,0)$	1	1	2	3	3	2	$(7.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(7,2,0)$	1	1	3	3	2	2	$(7.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(7,3,0)$	1	1	3	3	2	2	$(7.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(7,4,0)$	1	1	3	3	2	2	$(7.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(7,5,0)$	1	1	3	3	2	2	$(7.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(7,6,0)$	1	1	3	3	2	2	$(7.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(7,7,0)$	1	3	3	1	2	2	$(8.1,8.1,6.0)$	$(1.1,1.1,6.0)$
$(8,0,0)$	1	1	2	3	3	2	$(8.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(8,1,0)$	1	1	2	3	3	2	$(8.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(8,2,0)$	1	1	3	3	2	2	$(8.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(8,3,0)$	1	1	3	3	2	2	$(8.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(8,4,0)$	1	1	3	3	2	2	$(8.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(8,5,0)$	1	1	3	3	2	2	$(8.2,7.0,6.0)$	$(0.2,2.0,6.0)$

$(8,6,0)$	1	1	3	3	2	2	$(8.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(8,7,0)$	1	1	3	3	2	2	$(8.2,9.0,6.0)$	$(0.2,2.0,6.0)$
$(8,8,0)$	1	3	3	1	2	2	$(9.1,9.1,6.0)$	$(1.1,1.1,6.0)$
$(9,0,0)$	1	1	2	3	3	2	$(9.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(9,1,0)$	1	1	2	3	3	2	$(9.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(9,2,0)$	1	1	3	3	2	2	$(9.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(9,3,0)$	1	1	3	3	2	2	$(9.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(9,4,0)$	1	1	3	3	2	2	$(9.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(9,5,0)$	1	1	3	3	2	2	$(9.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(9,6,0)$	1	1	3	3	2	2	$(9.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(9,7,0)$	1	1	3	3	2	2	$(9.2,9.0,6.0)$	$(0.2,2.0,6.0)$
$(9,8,0)$	1	1	3	3	2	2	$(9.2,10.0,6.0)$	$(0.2,2.0,6.0)$
$(9,9,0)$	1	3	3	1	2	2	$(10.1,10.1,6.0)$	$(1.1,1.1,6.0)$
$(10,0,0)$	1	1	2	3	3	2	$(10.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(10,1,0)$	1	1	2	3	3	2	$(10.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(10,2,0)$	1	1	3	3	2	2	$(10.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(10,3,0)$	1	1	3	3	2	2	$(10.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(10,4,0)$	1	1	3	3	2	2	$(10.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(10,5,0)$	1	1	3	3	2	2	$(10.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(10,6,0)$	1	1	3	3	2	2	$(10.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(10,7,0)$	1	1	3	3	2	2	$(10.2,9.0,6.0)$	$(0.2,2.0,6.0)$
$(10,8,0)$	1	1	3	3	2	2	$(10.2,10.0,6.0)$	$(0.2,2.0,6.0)$
$(10,9,0)$	1	1	3	3	2	2	$(10.2,11.0,6.0)$	$(0.2,2.0,6.0)$
$(10,10,0)$	1	3	3	1	2	2	$(11.1,11.1,6.0)$	$(1.1,1.1,6.0)$
$(11,0,0)$	1	1	2	3	3	2	$(11.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(11,1,0)$	1	1	2	3	3	2	$(11.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(11,2,0)$	1	1	3	3	2	2	$(11.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(11,3,0)$	1	1	3	3	2	2	$(11.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(11,4,0)$	1	1	3	3	2	2	$(11.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(11,5,0)$	1	1	3	3	2	2	$(11.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(11,6,0)$	1	1	3	3	2	2	$(11.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(11,7,0)$	1	1	3	3	2	2	$(11.2,9.0,6.0)$	$(0.2,2.0,6.0)$
$(11,8,0)$	1	1	3	3	2	2	$(11.2,10.0,6.0)$	$(0.2,2.0,6.0)$
$(11,9,0)$	1	1	3	3	2	2	$(11.2,11.0,6.0)$	$(0.2,2.0,6.0)$
$(11,10,0)$	1	1	3	3	2	2	$(11.2,12.0,6.0)$	$(0.2,2.0,6.0)$
$(11,11,0)$	1	3	3	1	2	2	$(12.1,12.1,6.0)$	$(1.1,1.1,6.0)$
$(12,0,0)$	1	1	2	3	3	2	$(12.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(12,1,0)$	1	1	2	3	3	2	$(12.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(12,2,0)$	1	1	3	3	2	2	$(12.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(12,3,0)$	1	1	3	3	2	2	$(12.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(12,4,0)$	1	1	3	3	2	2	$(12.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(12,5,0)$	1	1	3	3	2	2	$(12.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(12,6,0)$	1	1	3	3	2	2	$(12.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(12,7,0)$	1	1	3	3	2	2	$(12.2,9.0,6.0)$	$(0.2,2.0,6.0)$
$(12,8,0)$	1	1	3	3	2	2	$(12.2,10.0,6.0)$	$(0.2,2.0,6.0)$
$(12,9,0)$	1	1	3	3	2	2	$(12.2,11.0,6.0)$	$(0.2,2.0,6.0)$
$(12,10,0)$	1	1	3	3	2	2	$(12.2,12.0,6.0)$	$(0.2,2.0,6.0)$
$(12,11,0)$	1	1	3	3	2	2	$(12.2,13.0,6.0)$	$(0.2,2.0,6.0)$
$(12,12,0)$	1	3	3	1	2	2	$(13.1,13.1,6.0)$	$(1.1,1.1,6.0)$
$(13,0,0)$	1	1	2	3	3	2	$(13.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(13,1,0)$	1	1	2	3	3	2	$(13.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(13,2,0)$	1	1	3	3	2	2	$(13.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(13,3,0)$	1	1	3	3	2	2	$(13.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(13,4,0)$	1	1	3	3	2	2	$(13.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(13,5,0)$	1	1	3	3	2	2	$(13.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(13,6,0)$	1	1	3	3	2	2	$(13.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(13,7,0)$	1	1	3	3	2	2	$(13.2,9.0,6.0)$	$(0.2,2.0,6.0)$
$(13,8,0)$	1	1	3	3	2	2	$(13.2,10.0,6.0)$	$(0.2,2.0,6.0)$
$(13,9,0)$	1	1	3	3	2	2	$(13.2,11.0,6.0)$	$(0.2,2.0,6.0)$
$(13,10,0)$	1	1	3	3	2	2	$(13.2,12.0,6.0)$	$(0.2,2.0,6.0)$
$(13,11,0)$	1	1	3	3	2	2	$(13.2,13.0,6.0)$	$(0.2,2.0,6.0)$
$(13,12,0)$	1	1	3	3	2	2	$(13.2,14.0,6.0)$	$(0.2,2.0,6.0)$
$(13,13,0)$	1	3	3	1	2	2	$(14.1,14.1,6.0)$	$(1.1,1.1,6.0)$
$(14,0,0)$	1	1	2	3	3	2	$(14.2,4.0,4.0)$	$(0.2,4.0,4.0)$

$(14,1,0)$	1	1	2	3	3	2	$(14.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(14,2,0)$	1	1	3	3	2	2	$(14.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(14,3,0)$	1	1	3	3	2	2	$(14.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(14,4,0)$	1	1	3	3	2	2	$(14.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(14,5,0)$	1	1	3	3	2	2	$(14.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(14,6,0)$	1	1	3	3	2	2	$(14.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(14,7,0)$	1	1	3	3	2	2	$(14.2,9.0,6.0)$	$(0.2,2.0,6.0)$
$(14,8,0)$	1	1	3	3	2	2	$(14.2,10.0,6.0)$	$(0.2,2.0,6.0)$
$(14,9,0)$	1	1	3	3	2	2	$(14.2,11.0,6.0)$	$(0.2,2.0,6.0)$
$(14,10,0)$	1	1	3	3	2	2	$(14.2,12.0,6.0)$	$(0.2,2.0,6.0)$
$(14,11,0)$	1	1	3	3	2	2	$(14.2,13.0,6.0)$	$(0.2,2.0,6.0)$
$(14,12,0)$	1	1	3	3	2	2	$(14.2,14.0,6.0)$	$(0.2,2.0,6.0)$
$(14,13,0)$	1	1	3	3	2	2	$(14.2,15.0,6.0)$	$(0.2,2.0,6.0)$
$(14,14,0)$	1	3	3	1	2	2	$(15.1,15.1,6.0)$	$(1.1,1.1,6.0)$
$(15,0,0)$	1	1	2	3	3	2	$(15.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(15,1,0)$	1	1	2	3	3	2	$(15.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(15,2,0)$	1	1	3	3	2	2	$(15.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(15,3,0)$	1	1	3	3	2	2	$(15.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(15,4,0)$	1	1	3	3	2	2	$(15.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(15,5,0)$	1	1	3	3	2	2	$(15.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(15,6,0)$	1	1	3	3	2	2	$(15.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(15,7,0)$	1	1	3	3	2	2	$(15.2,9.0,6.0)$	$(0.2,2.0,6.0)$
$(15,8,0)$	1	1	3	3	2	2	$(15.2,10.0,6.0)$	$(0.2,2.0,6.0)$
$(15,9,0)$	1	1	3	3	2	2	$(15.2,11.0,6.0)$	$(0.2,2.0,6.0)$
$(15,10,0)$	1	1	3	3	2	2	$(15.2,12.0,6.0)$	$(0.2,2.0,6.0)$
$(15,11,0)$	1	1	3	3	2	2	$(15.2,13.0,6.0)$	$(0.2,2.0,6.0)$
$(15,12,0)$	1	1	3	3	2	2	$(15.2,14.0,6.0)$	$(0.2,2.0,6.0)$
$(15,13,0)$	1	1	3	3	2	2	$(15.2,15.0,6.0)$	$(0.2,2.0,6.0)$
$(15,14,0)$	1	1	3	3	2	2	$(15.2,16.0,6.0)$	$(0.2,2.0,6.0)$
$(15,15,0)$	1	3	3	1	2	2	$(16.1,16.1,6.0)$	$(1.1,1.1,6.0)$
$(16,0,0)$	1	1	2	3	3	2	$(16.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(16,1,0)$	1	1	2	3	3	2	$(16.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(16,2,0)$	1	1	3	3	2	2	$(16.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(16,3,0)$	1	1	3	3	2	2	$(16.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(16,4,0)$	1	1	3	3	2	2	$(16.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(16,5,0)$	1	1	3	3	2	2	$(16.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(16,6,0)$	1	1	3	3	2	2	$(16.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(16,7,0)$	1	1	3	3	2	2	$(16.2,9.0,6.0)$	$(0.2,2.0,6.0)$
$(16,8,0)$	1	1	3	3	2	2	$(16.2,10.0,6.0)$	$(0.2,2.0,6.0)$
$(16,9,0)$	1	1	3	3	2	2	$(16.2,11.0,6.0)$	$(0.2,2.0,6.0)$
$(16,10,0)$	1	1	3	3	2	2	$(16.2,12.0,6.0)$	$(0.2,2.0,6.0)$
$(16,11,0)$	1	1	3	3	2	2	$(16.2,13.0,6.0)$	$(0.2,2.0,6.0)$
$(16,12,0)$	1	1	3	3	2	2	$(16.2,14.0,6.0)$	$(0.2,2.0,6.0)$
$(16,13,0)$	1	1	3	3	2	2	$(16.2,15.0,6.0)$	$(0.2,2.0,6.0)$
$(16,14,0)$	1	1	3	3	2	2	$(16.2,16.0,6.0)$	$(0.2,2.0,6.0)$
$(16,15,0)$	1	1	3	3	2	2	$(16.2,17.0,6.0)$	$(0.2,2.0,6.0)$
$(16,16,0)$	1	3	3	1	2	2	$(17.1,17.1,6.0)$	$(1.1,1.1,6.0)$
$(17,0,0)$	1	1	2	3	3	2	$(17.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(17,1,0)$	1	1	2	3	3	2	$(17.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(17,2,0)$	1	1	3	3	2	2	$(17.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(17,3,0)$	1	1	3	3	2	2	$(17.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(17,4,0)$	1	1	3	3	2	2	$(17.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(17,5,0)$	1	1	3	3	2	2	$(17.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(17,6,0)$	1	1	3	3	2	2	$(17.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(17,7,0)$	1	1	3	3	2	2	$(17.2,9.0,6.0)$	$(0.2,2.0,6.0)$
$(17,8,0)$	1	1	3	3	2	2	$(17.2,10.0,6.0)$	$(0.2,2.0,6.0)$
$(17,9,0)$	1	1	3	3	2	2	$(17.2,11.0,6.0)$	$(0.2,2.0,6.0)$
$(17,10,0)$	1	1	3	3	2	2	$(17.2,12.0,6.0)$	$(0.2,2.0,6.0)$
$(17,11,0)$	1	1	3	3	2	2	$(17.2,13.0,6.0)$	$(0.2,2.0,6.0)$
$(17,12,0)$	1	1	3	3	2	2	$(17.2,14.0,6.0)$	$(0.2,2.0,6.0)$
$(17,13,0)$	1	1	3	3	2	2	$(17.2,15.0,6.0)$	$(0.2,2.0,6.0)$
$(17,14,0)$	1	1	3	3	2	2	$(17.2,16.0,6.0)$	$(0.2,2.0,6.0)$
$(17,15,0)$	1	1	3	3	2	2	$(17.2,17.0,6.0)$	$(0.2,2.0,6.0)$
$(17,16,0)$	1	1	3	3	2	2	$(17.2,18.0,6.0)$	$(0.2,2.0,6.0)$

$(17, 17, 0)$	1	3	3	1	2	2	$(18.1, 18.1, 6.0)$	$(1.1, 1.1, 6.0)$
$(18, 0, 0)$	1	1	2	3	3	2	$(18.2, 4.0, 4.0)$	$(0.2, 4.0, 4.0)$
$(18, 1, 0)$	1	1	2	3	3	2	$(18.2, 5.0, 4.0)$	$(0.2, 4.0, 4.0)$
$(18, 2, 0)$	1	1	3	3	2	2	$(18.2, 4.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 3, 0)$	1	1	3	3	2	2	$(18.2, 5.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 4, 0)$	1	1	3	3	2	2	$(18.2, 6.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 5, 0)$	1	1	3	3	2	2	$(18.2, 7.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 6, 0)$	1	1	3	3	2	2	$(18.2, 8.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 7, 0)$	1	1	3	3	2	2	$(18.2, 9.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 8, 0)$	1	1	3	3	2	2	$(18.2, 10.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 9, 0)$	1	1	3	3	2	2	$(18.2, 11.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 10, 0)$	1	1	3	3	2	2	$(18.2, 12.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 11, 0)$	1	1	3	3	2	2	$(18.2, 13.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 12, 0)$	1	1	3	3	2	2	$(18.2, 14.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 13, 0)$	1	1	3	3	2	2	$(18.2, 15.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 14, 0)$	1	1	3	3	2	2	$(18.2, 16.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 15, 0)$	1	1	3	3	2	2	$(18.2, 17.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 16, 0)$	1	1	3	3	2	2	$(18.2, 18.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 17, 0)$	1	1	3	3	2	2	$(18.2, 19.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(18, 18, 0)$	1	3	3	1	2	2	$(19.1, 19.1, 6.0)$	$(1.1, 1.1, 6.0)$
$(19, 0, 0)$	1	1	2	3	3	2	$(19.2, 4.0, 4.0)$	$(0.2, 4.0, 4.0)$
$(19, 1, 0)$	1	1	2	3	3	2	$(19.2, 5.0, 4.0)$	$(0.2, 4.0, 4.0)$
$(19, 2, 0)$	1	1	3	3	2	2	$(19.2, 4.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 3, 0)$	1	1	3	3	2	2	$(19.2, 5.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 4, 0)$	1	1	3	3	2	2	$(19.2, 6.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 5, 0)$	1	1	3	3	2	2	$(19.2, 7.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 6, 0)$	1	1	3	3	2	2	$(19.2, 8.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 7, 0)$	1	1	3	3	2	2	$(19.2, 9.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 8, 0)$	1	1	3	3	2	2	$(19.2, 10.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 9, 0)$	1	1	3	3	2	2	$(19.2, 11.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 10, 0)$	1	1	3	3	2	2	$(19.2, 12.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 11, 0)$	1	1	3	3	2	2	$(19.2, 13.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 12, 0)$	1	1	3	3	2	2	$(19.2, 14.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 13, 0)$	1	1	3	3	2	2	$(19.2, 15.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 14, 0)$	1	1	3	3	2	2	$(19.2, 16.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 15, 0)$	1	1	3	3	2	2	$(19.2, 17.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 16, 0)$	1	1	3	3	2	2	$(19.2, 18.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 17, 0)$	1	1	3	3	2	2	$(19.2, 19.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 18, 0)$	1	1	3	3	2	2	$(19.2, 20.0, 6.0)$	$(0.2, 2.0, 6.0)$
$(19, 19, 0)$	1	3	3	1	2	2	$(20.1, 20.1, 6.0)$	$(1.1, 1.1, 6.0)$

Table E.21: The DP results for the second schedule of a small instance with bigger differences arrival rates.

State	$p = 1$		$p = 2$		$p = 3$		Expected new state	Variance of the schedule
	$d = 1$	$d = 2$	$d = 1$	$d = 2$	$d = 1$	$d = 2$		
$(0, 0, 0)$	1	2	2	1	3	3	$(3.1, 3.1, 2.0)$	$(3.1, 3.1, 2.0)$
$(1, 0, 0)$	3	3	1	2	2	1	$(3.0, 3.1, 3.1)$	$(2.0, 3.1, 3.1)$
$(1, 1, 0)$	1	2	3	3	2	1	$(4.1, 3.0, 3.1)$	$(3.1, 2.0, 3.1)$
$(2, 0, 0)$	3	3	1	2	2	1	$(4.0, 3.1, 3.1)$	$(2.0, 3.1, 3.1)$
$(2, 1, 0)$	3	3	1	2	2	1	$(4.0, 4.1, 3.1)$	$(2.0, 3.1, 3.1)$
$(2, 2, 0)$	1	2	3	3	2	1	$(5.1, 4.0, 3.1)$	$(3.1, 2.0, 3.1)$
$(3, 0, 0)$	1	3	3	2	2	1	$(4.1, 4.0, 3.1)$	$(2.1, 4.0, 3.1)$
$(3, 1, 0)$	1	3	2	1	3	2	$(4.1, 4.1, 4.0)$	$(2.1, 3.1, 4.0)$
$(3, 2, 0)$	1	3	2	1	3	2	$(4.1, 5.1, 4.0)$	$(2.1, 3.1, 4.0)$
$(3, 3, 0)$	1	3	2	1	3	2	$(4.1, 6.1, 4.0)$	$(2.1, 3.1, 4.0)$
$(4, 0, 0)$	1	1	2	3	3	2	$(4.2, 4.0, 4.0)$	$(0.2, 4.0, 4.0)$
$(4, 1, 0)$	1	3	2	1	3	2	$(5.1, 4.1, 4.0)$	$(2.1, 3.1, 4.0)$
$(4, 2, 0)$	1	3	2	1	3	2	$(5.1, 5.1, 4.0)$	$(1.1, 3.1, 4.0)$
$(4, 3, 0)$	1	3	2	1	3	2	$(5.1, 6.1, 4.0)$	$(2.1, 3.1, 4.0)$
$(4, 4, 0)$	1	3	3	1	2	2	$(5.1, 5.1, 6.0)$	$(1.1, 1.1, 6.0)$
$(5, 0, 0)$	1	1	2	3	3	2	$(5.2, 4.0, 4.0)$	$(0.2, 4.0, 4.0)$

$(5,1,0)$	1	1	2	3	3	2	$(5.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(5,2,0)$	1	3	2	1	3	2	$(6.1,5.1,4.0)$	$(2.1,3.1,4.0)$
$(5,3,0)$	1	3	2	1	3	2	$(6.1,6.1,4.0)$	$(2.1,3.1,4.0)$
$(5,4,0)$	1	1	3	3	2	2	$(5.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(5,5,0)$	1	3	3	1	2	2	$(6.1,6.1,6.0)$	$(1.1,1.1,6.0)$
$(6,0,0)$	1	1	2	3	3	2	$(6.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(6,1,0)$	1	1	2	3	3	2	$(6.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(6,2,0)$	1	1	2	3	3	2	$(6.2,6.0,4.0)$	$(0.2,4.0,4.0)$
$(6,3,0)$	1	1	3	3	2	2	$(6.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(6,4,0)$	1	1	3	3	2	2	$(6.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(6,5,0)$	1	3	3	1	2	2	$(7.1,6.1,6.0)$	$(1.1,1.1,6.0)$
$(6,6,0)$	1	3	3	1	2	2	$(7.1,7.1,6.0)$	$(1.1,1.1,6.0)$
$(7,0,0)$	1	1	2	3	3	2	$(7.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(7,1,0)$	1	1	2	3	3	2	$(7.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(7,2,0)$	1	1	2	3	3	2	$(7.2,6.0,4.0)$	$(0.2,4.0,4.0)$
$(7,3,0)$	1	1	3	3	2	2	$(7.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(7,4,0)$	1	1	3	3	2	2	$(7.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(7,5,0)$	1	1	3	3	2	2	$(7.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(7,6,0)$	1	3	3	1	2	2	$(8.1,7.1,6.0)$	$(1.1,1.1,6.0)$
$(7,7,0)$	1	3	3	1	2	2	$(8.1,8.1,6.0)$	$(1.1,1.1,6.0)$
$(8,0,0)$	1	1	2	3	3	2	$(8.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(8,1,0)$	1	1	2	3	3	2	$(8.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(8,2,0)$	1	1	3	3	2	2	$(8.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(8,3,0)$	1	1	3	3	2	2	$(8.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(8,4,0)$	1	1	3	3	2	2	$(8.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(8,5,0)$	1	1	3	3	2	2	$(8.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(8,6,0)$	1	1	3	3	2	2	$(8.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(8,7,0)$	1	3	3	1	2	2	$(9.1,8.1,6.0)$	$(1.1,1.1,6.0)$
$(8,8,0)$	1	3	3	1	2	2	$(9.1,9.1,6.0)$	$(1.1,1.1,6.0)$
$(9,0,0)$	1	1	2	3	3	2	$(9.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(9,1,0)$	1	1	2	3	3	2	$(9.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(9,2,0)$	1	1	3	3	2	2	$(9.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(9,3,0)$	1	1	3	3	2	2	$(9.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(9,4,0)$	1	1	3	3	2	2	$(9.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(9,5,0)$	1	1	3	3	2	2	$(9.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(9,6,0)$	1	1	3	3	2	2	$(9.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(9,7,0)$	1	1	3	3	2	2	$(9.2,9.0,6.0)$	$(0.2,2.0,6.0)$
$(9,8,0)$	1	3	3	1	2	2	$(10.1,9.1,6.0)$	$(1.1,1.1,6.0)$
$(9,9,0)$	1	3	3	1	2	2	$(10.1,10.1,6.0)$	$(1.1,1.1,6.0)$
$(10,0,0)$	1	1	2	3	3	2	$(10.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(10,1,0)$	1	1	2	3	3	2	$(10.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(10,2,0)$	1	1	3	3	2	2	$(10.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(10,3,0)$	1	1	3	3	2	2	$(10.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(10,4,0)$	1	1	3	3	2	2	$(10.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(10,5,0)$	1	1	3	3	2	2	$(10.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(10,6,0)$	1	1	3	3	2	2	$(10.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(10,7,0)$	1	1	3	3	2	2	$(10.2,9.0,6.0)$	$(0.2,2.0,6.0)$
$(10,8,0)$	1	1	3	3	2	2	$(10.2,10.0,6.0)$	$(0.2,2.0,6.0)$
$(10,9,0)$	1	3	3	1	2	2	$(11.1,10.1,6.0)$	$(1.1,1.1,6.0)$
$(10,10,0)$	1	3	3	1	2	2	$(11.1,11.1,6.0)$	$(1.1,1.1,6.0)$
$(11,0,0)$	1	1	2	3	3	2	$(11.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(11,1,0)$	1	1	2	3	3	2	$(11.2,5.0,4.0)$	$(0.2,4.0,4.0)$
$(11,2,0)$	1	1	3	3	2	2	$(11.2,4.0,6.0)$	$(0.2,2.0,6.0)$
$(11,3,0)$	1	1	3	3	2	2	$(11.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(11,4,0)$	1	1	3	3	2	2	$(11.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(11,5,0)$	1	1	3	3	2	2	$(11.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(11,6,0)$	1	1	3	3	2	2	$(11.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(11,7,0)$	1	1	3	3	2	2	$(11.2,9.0,6.0)$	$(0.2,2.0,6.0)$
$(11,8,0)$	1	1	3	3	2	2	$(11.2,10.0,6.0)$	$(0.2,2.0,6.0)$
$(11,9,0)$	1	1	3	3	2	2	$(11.2,11.0,6.0)$	$(0.2,2.0,6.0)$
$(11,10,0)$	1	3	3	1	2	2	$(12.1,11.1,6.0)$	$(1.1,1.1,6.0)$
$(11,11,0)$	1	3	3	1	2	2	$(12.1,12.1,6.0)$	$(1.1,1.1,6.0)$
$(12,0,0)$	1	1	2	3	3	2	$(12.2,4.0,4.0)$	$(0.2,4.0,4.0)$
$(12,1,0)$	1	1	2	3	3	2	$(12.2,5.0,4.0)$	$(0.2,4.0,4.0)$

$(12,2,0)$	1	1	2	3	3	2	$(12.2,6.0,4.0)$	$(0.2,4.0,4.0)$
$(12,3,0)$	1	1	3	3	2	2	$(12.2,5.0,6.0)$	$(0.2,2.0,6.0)$
$(12,4,0)$	1	1	3	3	2	2	$(12.2,6.0,6.0)$	$(0.2,2.0,6.0)$
$(12,5,0)$	1	1	3	3	2	2	$(12.2,7.0,6.0)$	$(0.2,2.0,6.0)$
$(12,6,0)$	1	1	3	3	2	2	$(12.2,8.0,6.0)$	$(0.2,2.0,6.0)$
$(12,7,0)$	1	1	3	3	2	2	$(12.2,9.0,6.0)$	$(0.2,2.0,6.0)$
$(12,8,0)$	1	1	3	3	2	2	$(12.2,10.0,6.0)$	$(0.2,2.0,6.0)$
$(12,9,0)$	1	1	3	3	2	2	$(12.2,11.0,6.0)$	$(0.2,2.0,6.0)$
$(12,10,0)$	1	1	3	3	2	2	$(12.2,12.0,6.0)$	$(0.2,2.0,6.0)$
$(12,11,0)$	1	3	3	1	2	2	$(13.1,12.1,6.0)$	$(1.1,1.1,6.0)$
$(12,12,0)$	1	3	3	1	2	2	$(13.1,13.1,6.0)$	$(1.1,1.1,6.0)$

Table E.22: The DP results for the first schedule of a small instance with bigger differences arrival rates.

State	$p=1$		$p=2$		$p=3$		Expected new state	Variance of the schedule
	$d=1$	$d=2$	$d=1$	$d=2$	$d=1$	$d=2$		
$(0,0,0)$	1	2	2	1	3	3	$(3.1,3.1,2.0)$	$(3.1,3.1,2.0)$