An ADP approach for the allocation of orthopaedic patients to the operating rooms at the Sint Maartenskliniek

Nico de Jongh

December 24, 2020

MSc Thesis
Stochastic Operations Research
Applied Mathematics
University of Twente

Daily Supervisors:
prof. dr. R.J.Boucherie (UT)
dr. ir. M. de Graaf (UT)
ir. R.F.M. Vromans (Rhythm B.V.)
ir. J.W.M. Otten (UT)

Graduation Committee:
prof. dr. R.J.Boucherie (UT)
dr. ir. M. de Graaf (UT)
ir. R.F.M. Vromans (Rhythm B.V.)
ir. J.W.M. Otten (UT)
dr. M. Walter (UT)

UNIVERSITY OF TWENTE.

rhythm

Sint Maartenskliniek

Preface

In front of you lies the thesis called "An ADP approach for the allocation of orthopaedic patients to the operating rooms at the Sint Maartenskliniek". With this research we aimed to find a smart patient-booking strategy for the operating rooms of the Sint Maartenskliniek (SMK), that could improve the operating room performance. I am grateful to the SMK and to Rhythm B.V. for giving me the opportunity to work on a topic that perfectly fits within my interests.

I started the project in March 2020. It all started with a guided tour through the SMK, where I was introduced to the most advanced technological developments in the field of orthopedics, and where I got a real valuable present afterwards: a mug of the SMK. Two weeks later, I was heading back home from a working day at the SMK when Mark Rutte gave a press conference to announce the measures regarding the COVID-19 virus, whereafter I could not be on location that much anymore. I was asked to join a project team to contribute to the problems that the SMK faced with due to this pandemic. Helping with this side-project felt really valuable and I am grateful for being asked to participate in this project. After this I continued working from home on the final project, with having only a few outings to the SMK, that were necessary to obtain some data. I really enjoyed being at the SMK, having my personal service card and working with the friendly colleagues from Rhythm and the SMK.

This report would not be here without the help of my supervisors. I would like to thank Rob, Richard, Maarten, and Maurits for their support and guidance through out the project. I have always enjoyed the meetings with you to discuss the progress and come up with new ideas. Especially I would like to thank Rob for the extra time he took in the end to run the programs on an external computer.

I would like to appreciate the graduation committee for taking the time and effort to read my work.

Finally, I end this preface by thanking the people in my personal surroundings. First mentioning my gratefulness to my family, who have always supported me. Thanking my friends, with whom I drank coffee to relax or took a forest walk in between studying, with whom I had discussions about important political issues that had to be addressed in a friendly manner, and with whom I did lots of other stuff that made my student life much more fascinating. Lastly, thanking my current- and old roommates for offering me a place where I have always felt at home.

Nico de Jongh,

Enschede, 20 November 2020.

Abstract

Operating rooms (ORs) are expensive areas of the hospital. Efficient use of the ORs is desired to be cost-effective, and to offer patients a good service level. This study focuses on improving the offline operational booking of orthopaedic patients to operating room (OR) sessions at the Sint Maartenskliniek hospital (SMK) to improve the OR performance. OR-patient booking is a challenging problem, due the existence of multi-resource constraints, multi-priority levels, and the multiple objectives, that seek to find a balance in the conflicting interests of the three stakeholders: patients, hospital, and personnel. Moreover, the patient-booking deals with an uncertain incoming demand, including emergencies, the variance of surgical procedures, and other diverse complications. We propose an method for patient booking, taking into account the access times of patients in an equitable manner, the productivity of the operating rooms to be cost-effective, and the expected earliness and tardiness (ET) costs of the end times of the sessions. We introduce two approaches to affect the schedule in terms of the earliness and tardiness costs of the end time of the session, referred to as the risk-pooling and the risk-spreading method. We formulate the problem as a Markov decision process (MDP) that takes into account the current patient schedule, the current session plan, and the future arrivals, cancellations, and newly opened sessions. We use a linear programming approach as a basis to solve the MDP. To deal with an intractable number of variables and constraints, we develop an ADP approach for this problem, using an affine value function approximation. We solve the approximate linear program (ALP) by the column generation algorithm, to obtain an approximate optimal policy (AOP) for OR-patient booking. We analyse the approximate optimal policy and evaluate the performance of the AOP against the FIFO and the myopic policy through simulation. The size of the problem and the non-linearity in the objective caused that the analysis could take place on a small instance to obtain results within time. First simulations tend to show that the AOP outperforms the FIFO strategy in terms of productivity and access times for non-acute patients, but performs worse in terms of cancellations and access times of acute patients. The model leaves some freedom to the decision maker to decide on the importance of each objective. A validation of the simulation with more runs and a detailed analysis of the performance on the AOP on multiple larger instances is needed to verify and generalize the results. The expected earliness and tardiness costs of the end times of session can be affected by the method of risk-pooling and the method of risk-spreading. Incorporating these objectives increase complexity of the model, therefore a trade-off should be made in computation time and quality of the solution.

Contents

Glossary

Access time	The time a patient has to wait for his or her surgery once the patient enters the waiting list.
Advanced scheduling	Determines the day of surgery of the patient, given the future capacity and the future demand.
Allocation scheduling	Determines the operating room and the starting time or the sequence of the procedures on the planned day of surgery.
Anaesthesiologist	The anaesthesiologist is responsible for the patient's condition during surgery.
Block scheduling	Slots or blocks (i.e., a combination of an OR an a day) are typically allocated to a discipline or to a surgeon (group). In the next step surgeons book cases into the blocks assigned to them.
Booking accuracy	Defines the percentage of surgeries that end within fifteen minutes from the planned surgery time.
Booking horizon	The period for how far in the future surgeries can be scheduled.
Capacity reservation	Keeping capacity (time of a session) free for specific patients, mostly used for emergencies.
Change-over time	The time between two successive surgeries.
Elective patient	Patients for whom the surgery can be planned in advance.
Inpatients	Hospitalized patients who have to stay overnight.
Invitation time	The time between the announcement of the surgery date and the day of surgery. Maximal internal access time is the maximum time between the moment that a patient is added to the waiting list of the OR until the moment that the patient goes to the OR for surgery.
Non-elective patient	Patients for whom a surgery is unexpected and hence needs to be fitted into the schedule on short notice.
Offline planning	Involves collecting treatment requests until all demand for a certain period is known and then booking all requests.
Online planning	Patients get a direct response to their treatment request in the form of a starting week.
Orthopedic surgeon	The orthopedic surgeon performs the surgery on the patient.
Outpatients	Patients who enter and leave the hospital on the same day.
Productivity	The average number of surgeries per session.
Surgery time	The duration of a surgery, existing of the preparation time, the cutting time, and the discharge time.

Abbreviations

ADP	Approximate Dynamic Programming.
ALP	Approximate Linear Programming.
HNR	A dutch abbreviation: Herhaal na Radiologie consult. Stands for: repeat after radiology-consult.
ICU	Intensive Care Unit.
MDP	Markov Decision Process.
MP	Master Problem.
MSS	Master Surgery Scheduling.
k-NN	The k nearest-neighbours method.
OR	Operating Room.
PACU	Post-anesthesia care unit.
PP	Pricing Problem.
RMP	Restricted Master Problem.
SMK	Sint Maartenskliniek.
SVF	Shortest Variance First.

1 Introduction

Efficient-scheduling of the operating rooms (ORs) has significant advantages for healthcare [1]. It not only offers the possibility to reduce long waiting lists and give patients timely access to care, it also reduces costs. Operating rooms are a significant source of the hospital's income: they are cost expensive in the sense of labor and capital [1]. Healthcare costs are known to be rising in the Netherlands, due to technological developments and the ageing population [2], and efficient use of the ORs becomes even more important. With the current COVID-19 pandemic the world is facing, regular care is downsized leading to longer patient waiting lists. To decrease the size of the waiting list in the future, improvement of the OR-efficiency is supportive.

OR scheduling is a complex problem. The ORs are (highly) influenced by variance, due to diversity of surgical procedures, complications, emergencies, and characteristics of patients. Complexity also comes with the conflicting interests of the stakeholders: the hospital, the personnel, and the patient. Where the hospital would prefer maximal filled sessions to be cost-effective, the personnel would experience more overwork, and where patients would prefer to know their surgery date as soon as possible, the probability of a cancellation due to the arrival of a more urgent patient increases. The operating rooms are only a small part of the orthopedic chain, restrictions come from both up- and downstream resources, next to the capacity-restrictions of the ORs itself. Given the importance of the operating rooms, the research in this field has already obtained a lot of attention in the past [1, 3].

The research performed in this thesis is done at the Sint Maartenskliniek (SMK). The Sint Maartenskliniek is a specialised dutch hospital that is leading in the field of posture and movement. The hospital offers its treatments in Nijmegen, Woerden, Boxmeer, and Tiel. The establishment in Nijmegen is the largest, which has the departments orthopedics, rheumatism, and a rehabilitation centre. This study will focus on the orthopedic department. In the year of 2019 the SMK performed over 8000 orthopedical procedures, which resulted in a revenue of more than 70 million euros. The SMK expects to see an increase in demand in the future, due to the ageing population in the Netherlands [4]. Providing good service to the patient is one goal, but the complexity increases when wishes of the personnel and hospital need also to be taken into account.

The topic of this research came up due to the currently experienced dissatisfaction on multiple performances of the OR at the SMK. The main complaints are the currently experienced number of offline cancellations, the below-norm experienced access times of patients, and on the variable working days of personnel, that means experiencing working times past the working hours. We will elaborate on the indicators in the Section 1.1. To improve these performances, this study focuses on the improvement of the booking procedure of patients to the operating rooms. The goal is to find a smart OR-patient booking strategy that leads to an increase in satisfaction level for the hospital by having a higher productivity, for the personnel by having less variable working times, and for the patient to experience fewer cancellations and shorter access times to the OR.

Section 1.1 motivates the research in this topic. Section 1.2 points out the scope of the research and demonstrates the relevance of the research. Section 1.3 gives the problem description and states the research goal, moreover research questions are formulated. Section 1.3 can be used as a reading guide for the remainder of this thesis.

1.1 Motivation

This section motivates the research on this topic. The motivation can roughly be divided into two main reasons. First, we will motivate the need for attention by discussing the current experiences of the stakeholders. Second, we will motivate the added value of having a supportive automated planning system.

The current performance of the OR-complex leads to dissatisfaction within the hospital, among the OR-personnel, and among the patients. On the hospital's side of view a below-norm productivity is experienced. The hospital built a new OR-complex in November 2019 [5]. This new complex was built to have an extra OR, but mostly to increase OR-efficiency. The increase of productivity, that is an increase of the average number of surgeries performed per session, has not been realised yet. From the OR-personnel's perspective of view an unacceptable variability in working hours is experienced. A normal working day for OR-personnel starts around 7:30am and ends around 16:30pm. In the current situation the personnel experiences days that end to soon, or end very late. Currently, around one-fourth of the sessions end in a 15 minute range from the planned end times. This is non-desirable for the OR-personnel. From the patient's perspective of view cancellations are experienced once their surgery date was given. Currently one out of the [redacted] patients experiences re-scheduling by the SMK. More than half of the cancellations done by the SMK occur due to the arrivals of patient with higher priority, while full-capacity was already assigned. This means that the more urgent patient receives the spot of the cancelled patient. Next to this, patients experience below-norm access times to the OR. This means that once the patient is put on the waiting list, it takes too long before the patient can really go into surgery. This increases the potential impact of delays on patients' health, which is non-desirable. Overall, there is a strong desire for improvement on the above mentioned issues.

An additional motivation for research on this topic is supporting the orthopaedic planning department. The patient scheduling for the operating rooms is currently done manually by the planner. This is seen as a complex job, due to the many restrictions the planner has to take into account, and due to the dynamics of the process. A system that provides information on the optimal allocation of patients to the OR could be of support to the planning department. Moreover, the employees of the planning department use their experience of the past years to make, according to their knowledge, the best choices. In case the planning department is in need for an extra planner, experience cannot be transferred completely, in which case an automated supporting tool for the patient booking could be a solution.

The current experiences show the need for research in this field, coping with the wishes of the different stakeholders, and increasing their satisfaction level. The orthopedic department believes there is a high improvement potential on the OR-performance.

1.2 Scope and relevance

This study is done for the orthopedics department of the Sint Maartenskliniek. The research focuses on the offline operational patient booking part of the operating rooms in Nijmegen. Scheduling of personnel, beds and operating materials is outside the scope of this research. Offline operational booking is done before the day of surgery, online operational booking deals with the dynamic planning of events on the day itself. Next to the operational decisions, strategic and tactical decisions are made [6]. Strategic decisions involve the size of the capacity, such as the budget for the operating rooms. The tactical decisions involve the timing of capacity. More information on the different decisions level, specified for the SMK, will be given in Section 2.1.3. We use the outcome of the strategic and tactical decisions as input for the operational decisions; the mapping of capacity to patients. The SMK states to already have a sufficient model for the strategic and tactical decisions, whereas the operational decisions do not have sufficient support at this moment. Therefore we focus on the offline operational decisions.

In this research, literature is consulted for methodologies that could improve the OR-patient booking. Literature tells us that patient scheduling can be roughly divided into two streams: Allocation scheduling and advanced scheduling. Advanced scheduling deals with the allocation given the future capacity and the future demand, whereas allocation scheduling refers to assigning specific appointment times and resources to patients, once all patients for a given service have been identified [3]. Based on these findings, and the fact that we focus on the complete offline operational booking part, the research will be done separately on these two parts of scheduling. The most effort in this research will be given on the advanced scheduling part, since once this is done, standard methods could be used to for the allocation scheduling.

A scheduling problem can be approached as a deterministic process, or as a stochastic process. We model the OR-patient booking as a stochastic process, in order to incorporate multiple uncertainties that affect the OR-schedule. One of these uncertainties is the (future) patient arrival process. In literature patients can be divided into elective and non-elective patients [1]. Elective patients are patients for whom the surgery can be planned in advance. Non-elective patients are patients for whom a surgery is unexpected and hence needs to be fitted into the schedule on short notice. Since we are focusing on the orthopedic chain, the most urgent non-elective patients are not in danger to life, but have to be treated within two days after arrival. Other non-elective patients are grouped into a maximum internal access time of 14 days or 30 days, where elective patients are grouped to be treated within 60 days, or within 180 days after arrival. This research deals with the uncertain arrival process of the different priority levels of patients. Moreover, we deal with having an uncertain supply, that is, it is not exactly known how much capacity of the operating rooms will be assigned to a surgeon in the future. Next to this, we introduce the possibility of having an already booked surgery, that is cancelled by the patient or by the SMK. The uncertainty of cancellations is included since current experiences show that over one third of the bookings is currently rescheduled, due to a cancellation by the hospital or by the patient.

This research focuses on improving the OR-performance on several indicators. In this research we set the most important objectives to be booking equitably while minimizing access times, and maximizing productivity, where we seek to find a balance between these objectives. To overcome the problem of the insufficient prediction of the end times of sessions, we introduce the minimization of the expected deviation of the planned end times as a second objective. For this, we make use of two separate approaches; risk-pooling and a slightly modified version of earliness and tardiness (ET) scheduling referred to with risk-spreading. With the aid of a model we try to achieve an optimal strategy regarding the previously mentioned objectives, taking into account the previously mentioned uncertainties. We compare the obtained policy to a policy similar to the current policy, and a myopic policy, through simulation.

The relevance of this research is that eventually the approach results in a program that can automatically assign patients to sessions, where the goal is to book patients in a better way than the current strategy does. Moreover, it could be of support to the planning department, and it takes away a small piece of the importance of having experience at in operating room planning. Next to this, it could result in a better functioning of the orthopedic chain, which is in everyone's interest. It will result in a higher satisfaction within the hospital, employees, and patients. The research is applicable for other health care institutions dealing with a similar problem.

1.3 Problem Description and Research Goal

We deal with a multi-resource, multi-priority, and multi-objective problem. Patients of multiple surgery specialties, with different urgency classes should be booked onto sessions of their performing surgeon, taking into account a combination of equitable patient booking with minimal access times, minimal deviation of the planned end times, and productivity. This research attempts to build a patient booking policy that seeks to take these characteristics and objectives into account, incorporating the stochastic nature of the surgery times and typical OR-events.

The main goal of this research, extending already developed research, is to improve the allocation of different type of patients, including the emergency-scale of patients, to specific sessions in order to minimize earliness and tardiness of the sessions. This would yield satisfaction within the employees (less over- and undertime), within the patients (fewer cancellations), and within the SMK, due to an increased usage of the sessions leading to more surgeries.

We aim to achieve this goal by answering the following research questions. These questions are used as a guideline through out this thesis.

1. *What is the current patient planning process at the SMK?*
   Section 2.1 of Chapter 2 describes and analyses the planning process for the operating rooms, and focuses on the patient planning process.
2. *What is the current OR performance in the SMK and which performances need attention?*
   Section 2.2 of Chapter 2 analyses the OR performance with the aid of Key Performance Indicators (KPIs) and process indicators. Eventually the indicators that are in scope are determined.
3. *What literature is available on patient planning methods that can improve OR performance?*
   Chapter 3 discusses the relevant literature on this topic and concludes on the contribution of this research.
4. *Which methods can be used to minimize the earliness and tardiness of session end times?*
   Chapter 4 discusses three methods that came up from the literature search. These methods are briefly explained, applied to SMK data, and compared to each other. The chapter concludes on relating the methodologies to a combined model that takes into account the other goals of this research.
5. *What type of model can be used to find the optimal policy that improve OR performance on the prescribed indicators?* Chapter 5 summarizes the objectives and constraints of such a model, formulates a Markov Decision Model that fits to the objectives and constraints, and explains the solution approach.
6. *What is the optimal policy for OR patient-booking and does the policy outperform the current policy and other heuristics?* Chapter 6 gives insights into policy and compares the performance of the approximate optimal policy against the FIFO policy that is a reasonable approximation of the current booking procedure, and against a myopic policy.

Chapter 7 concludes on the results obtained in this research and discusses the pros and cons of the proposed method, as well as recommendations for future research.

2 Current situation

This chapter describes the current planning process in Section 2.1. The section informs on the characteristics of the operating room scheduling at the SMK and explains to which decision level this research is applied. Section 2.2 discusses and analyses the current OR-performance on several key performance indicators (KPIs) and process indicators used by the SMK. The section shows that not all performances are currently at the desired level, and that there is need for a policy that improves the performance. The model that is proposed in this study is evaluated by the performance indicators explained in Section 2.2.

2.1 Current planning process

This section provides information on the department of orthopedics at the SMK. We start by explaining the orthopedic chain with its components in Section 2.1.1. Section 2.1.2 provides information on the current OR-complex, an important component of the orthopedic chain, on which this research will focus. Section 2.1.3 provides an overview of the current planning process of the operating rooms. Section 2.1.4 explains the process on the surgery day itself. Section 2.1.5) explains the characteristics of the demand side for the operating rooms, that is the different types of patients that visit the SMK for treatment.

2.1.1 The orthopedic chain

Orthopedics focuses on the condition of the musculoskeletal system, this includes bones, muscles, ligaments, tendons, and joints of the human body. The orthopedic chain exists of several components. Figure (2.1) shows the most common patient flows between these components. The orthopedic chain starts at the outpatient department where patients arrive for an initial consultation with an orthopedic surgeon. The outpatient department is the department where patients enter and leave the hospital on the same day. After the initial consultation, the patient could come back for further consultation, could be sent to the radiology department or the treatment could be terminated. Patients can receive an OR-ticket which adds the patient to the queue for a screening appointment. The screening appointment is often planned once the surgery date is known, since screening has to take place a few weeks before surgery. The OR-ticket is given by the orthopedic surgeon after a consultation by phone, after a follow-up consultation or after an HNR-consultation (Herhaal na Radiologie consult). Once the screening is done, the patient can go into surgery. The performing surgeon is often the same as the one for consultation. These surgeries take place in the OR-complex of the SMK. After surgery, patients are sent to the recovery, thereafter patients go to the inpatient clinic or the wards. When the stay at the ward is finished, patients have a final dismissal consultation to finish or continue treatment. Acute patients should be treated within 2 days after arrival, these patients pass through the orthopedic chain much faster than elective patients.

This research, in context of the orthopedic chain, focuses on the operating rooms of the OR-complex, an important subdivision of the orthopedic chain. This research comes in place once an OR ticket is given to a patient, and the patient enters the waiting list for the operating rooms.

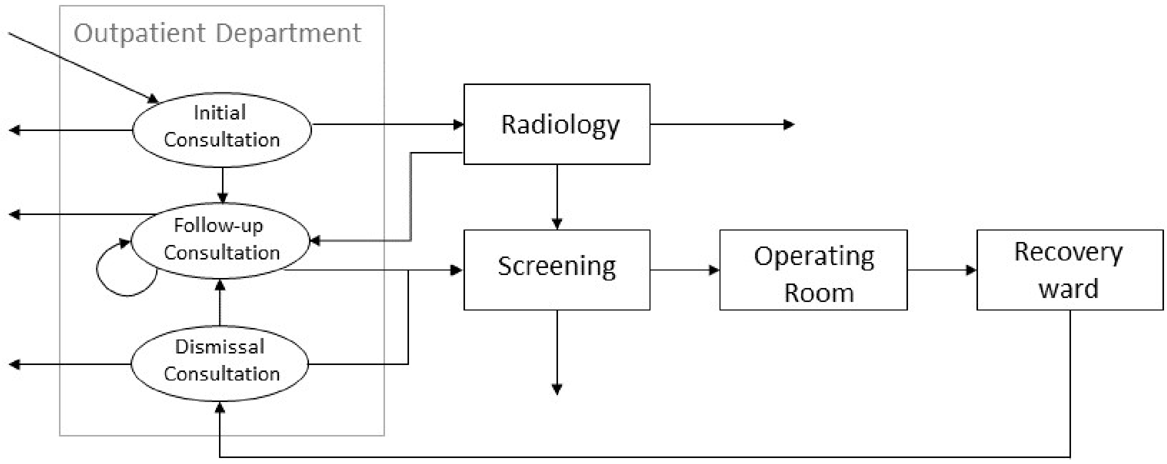

Figure 2.1.: Most common patient transitions between several components of the orthopedic chain.

2.1.2 The operating rooms

Surgeries are performed at the OR complex. The SMK renewed the OR-complex in November 2019 [5]. The new OR-complex exists of seven operating rooms, where the previous OR-complex had six operating rooms. Unlike the old OR-complex, the new ORs are foreseen of the newest technological developments that improve the operating environment for both the patient and the OR-team. Moreover, the new ORs are furnished in a similar fashion, such that each surgery can be executed at every OR. Next to this, small areas next to each OR are added for the set-up of instruments for the next patient to decrease change-over time between consecutive surgeries. Unlike the old OR-complex, the new ORs are on the same level, to increase personnel efficiency. In the old ORs mobile tables where used to move patients, in the new ORs patients have to be lifted up to a table, this can increase preparation time. On the hospital's point of view the hypothesis is that the new OR-complex should eventually lead to an overall increase of OR-efficiency.

A top view of the OR complex is given by Figure (2.2). This figure shows that each operating room has its own set-up room, and it shows that the operating rooms are at the same level. In the centre the recovery and the post-anesthesia care unit (PACU) are located. An operating room takes 55 squared metres.

The operating room complex is used by the orthopedics department and by the anaesthesia department. Once every two weeks one OR is used for surgeries by the anaesthesia department. The patient flow for anaesthesiologic surgeries is independent of the patent flow for orthopedic surgeries, and has other characteristics in terms of urgency levels and requirements for surgeries. The OR-patient booking of orthopedic surgeries can therefore be considered as a separate problem of the planning of the anesthesia department.

The orthopedics department made the choice to make capacity reservations at operating room 7 for emergency arrivals. In literature this is called the use of a dedicated OR to non-elective patients [7]. Capacity is saved from 13:00pm at OR 7. Due to the stochastic nature of the arrival process, it occurs that sometimes the reserved capacity is not fully booked. This situation is reflected in the fill-rate of OR7, which is discussed in Section 2.2.5.

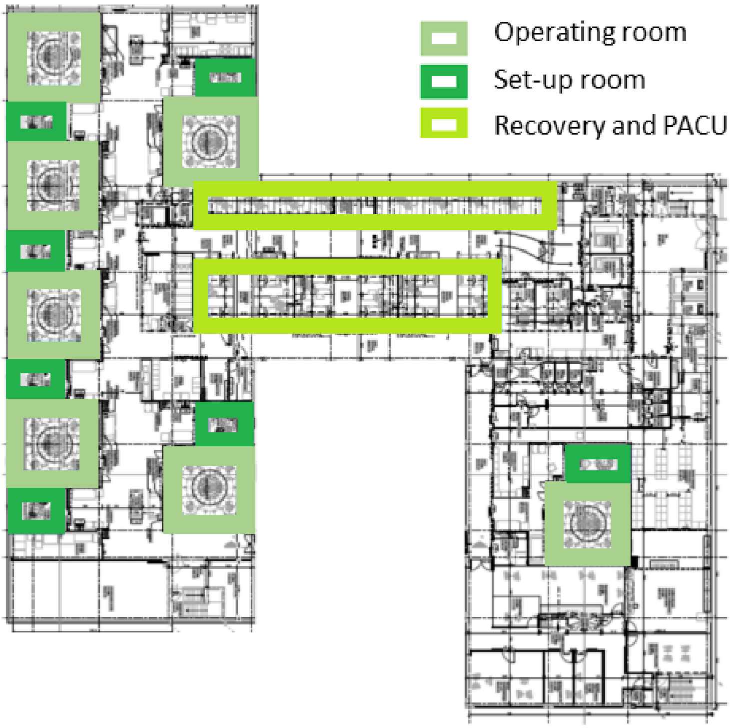

Figure 2.2.: A top view of the OR-complex.

2.1.3 The levels of planning

In this section we will describe the current patient booking process done by the SMK. First we will describe the level system the SMK uses to assign capacity. This results into a step wise planning procedure which we will describe.

The SMK distinguishes between several levels of planning of the OR. In the literature this level system is described with a strategic level (size of capacity), tactical level (timing of capacity), and an operational level where the capacity is mapped to patients before the day itself (offline) and during the day (online) [6]. Decisions made at each level correspond to decisions made at higher levels. Figure 2.3 shows the levels and the decisions that belong to each level.

The strategic level focuses on the planning on the long term. At the SMK this level exists of the yearly budget, based on the production goals. The SMK made the strategic decision to built a new OR complex with a seventh OR. At the tactical level, the timing of the capacity is done, for the SMK this includes deciding on the number of sessions per week, which is called the session plan. Next to this tactical decisions include deciding on which sessions are performed on which OR's and by which surgeon. Moreover, the tactical decision is made on the amount of capacity used for emergency arrivals. The last level, the operational level, focuses on creating the daily OR-schedule. First, offline operational decisions are made, these are decisions prior to the day of surgery. This includes booking the patients of the surgeon's waiting list to these sessions. This means that the patient gets his or her date of surgery, and a few days before surgery the time of surgery. Online operational decisions are made on the day itself, dealing with the uncertain events during the day and adjusting the schedule.

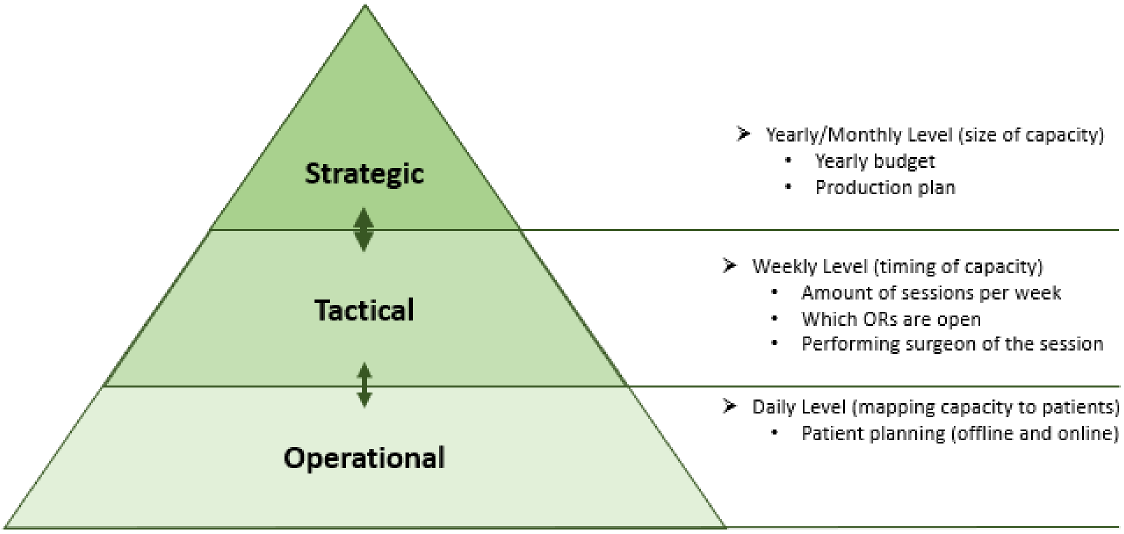

Figure 2.3.: Different levels of capacity planning at the OR.

To place this research in context of the planning, we will explain the successive decisions in detail, with the aid of Figure (2.4).

First, the OR-budget is set. This is done in November of each year, and thus 3 to 15 months in advance. At the same moment an first session plan for the whole year is created, which is updated quarterly. The session plan includes information on the opened OR-sessions and number of emergency slots. Shortly after the session plan has been made final, the session roster is created. In this roster the sessions are assigned to the operating rooms, and surgeons are assigned to the sessions. In the mean time, patients are visiting the hospital and can obtain an OR-ticket. This ticket includes medical information, such as the type of surgery, the expected surgery time and the performing surgeon. The expected surgery time is calculated by HiX Chipsoft, that determines the surgery time based on the twenty most recent similar surgeries performed by the surgeon. This proposed surgery time is personally checked by the performing surgeon, whom can adjust the time to his or her experience. The medical decision has been made, and now the information of the medical side is combined with the information on the supply side (i.e. the session roster) to book the patients to a session day. This research focuses on improving the decision of booking the patient onto a session. Approximately three weeks in advance the planned surgeries are made final and the patients are informed by telephone. Rescheduling can still take place if necessary, but the goal is to have no rescheduling. One week before surgery there is contact with patients to communicate whether or not the treatment is still possible under the current circumstances. Two days before treatment the order of surgeries is made final and one day before treatment the start time of the surgery is communicated to the patient.

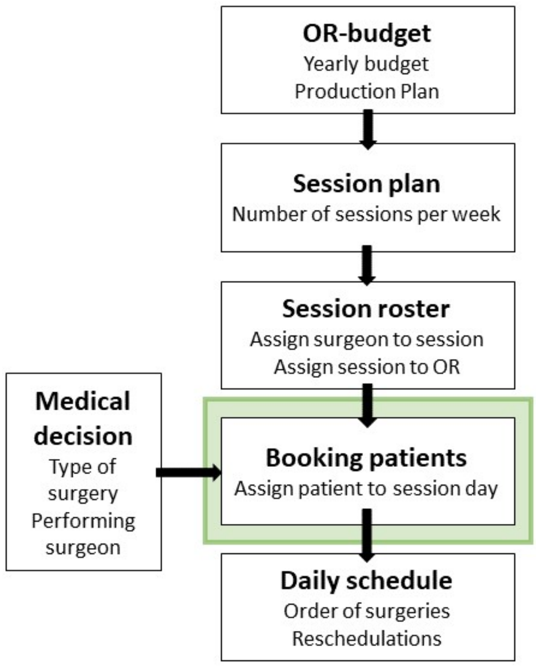

Figure 2.4.: The current planning process of the OR.

We refer to Appendix A.1 for a more detailed representation of the capacity-assignment of sessions.

2.1.4 The operating day

On the day itself the online planning part is executed. This means that due to new information (i.e. arrival of emergencies, illness of the staff or patient, no available equipment etc.) the schedule can be adjusted. Cancellations could lead to patients that enter the waiting list again, and therefore should be booked again in the offline planning part. Since the goal of this research is to decrease the realised deviation of the planned end times of the sessions, we need information on processes that cause variability of the end times.

At the operating day the patient is brought in on the OR complex before the surgery starts. In the data of 2019 of one surgeon we noticed that patients seem to always arrive on time; the latest arrivals was 12 minutes before treatment. The patient is often operated by the surgeon who diagnosed the patient. An anaesthesiologist is responsible for the patient's condition during the surgery. One anaesthesiologist can be responsible for two ORs at the same time, however the surgeries should start at different times, since the anaesthesiologist is needed at the start of surgery. We leave the need of an anaesthesiologist outside the scope of this research. Surgical assistants are needed for setting up the instruments and for help during surgery. Some surgeries are performed by or are used to school doctors in training, which increases surgery time. Once the surgery is finished the patient is brought to the recovery area or PACU.

The activities at one OR during the day can be divided into different components: the preparation-time, cut-time, discharge-time, change-overtime, idle time between ORs, and vacancy after idle time after the last surgery. The surgery time is the sum of the preparation-time, the cut-time, and the discharge-time. In this research the surgery time is stochastic, whereas we assume change-overtime is deterministic and does not depend on the type of surgery, the operating room, nor the personnel. The change-overtime is set to fifteen minutes, based on the goal of the SMK. The idle time between ORs, and the vacancy after idle time after the last surgery is preferred to be zero, to maximize OR-utilization.

2.1.5 Type of patients consulting the SMK

Different type of surgeries are performed at the operating rooms. The SMK distinguishes between 6 surgery units: spine, hip, knee, upper extremities (arms/shoulders/hands) (BE), foot, and orthopedics for children (for example scoliosis). The orthopedics for children are mostly performed at the location in Boxmeer, but since this research focuses on the operating rooms in Nijmegen, as indicated in Section 1.2, we leave them out of the scope of this research. Within these surgery units a surgery can be divided in a regular/first surgery, or in a special/revision surgery. Special surgeries are often complex and are the most recent development treatments offered by the SMK. Table 2.1 shows the monthly average number of patients treated by the SMK per surgery unit in the year of 2019. On average over 500 surgeries are performed per month, of which most surgeries concern the knee.

At the SMK five different urgency classifications are used for orthopedic patients based on their maximum internal access time to the OR. The maximum internal access time is the maximum time between the moment that a patient receives an OR-ticket until the moment that the patient goes to the OR for surgery. Patients are divided into urgency categories; elective, 2 months, 1 month, emergency, and acute, as shown by Table 2.2. The percentage of the treatments per urgency category per surgery unit are given as well in Table 2.2. Most of the patients that visit the SMK for treatment are elective patients. On average [redacted] acute patient arrive monthly. Note that acute orthopedic surgeries are not a matter of life and death, and therefore acute patients do not have to be treated directly, but are preferred to be treated within two days.

Table 2.1.: The monthly average of patient treatments per surgery unit in 2019.

Surgery unit	Average total (average specials)	Percentage
Back	[redacted]	[redacted]
Hip	[redacted]	[redacted]
Knee	[redacted]	[redacted]
Upper extremities (BE)	[redacted]	[redacted]
Foot	[redacted]	[redacted]
Unknown	[redacted]	[redacted]
Total	507 [redacted]	100%

Table 2.2.: Percentage of the treatments per urgency category per surgery unit in 2019.

Urgency Category	Elective	2 Months	1 Month	Emergency	Acute
Maximal Internal access (in days)	180	60	30	14	2
Back	70.0%	22.3%	3.7%	4.7%	2.3%
Hip	68.9%	12.8%	5.8%	6.9%	5.6%
Knee	73.2%	13.7%	5.3%	4.8%	3.0%
Shoulder (BE)	68.5%	18.0%	4.3%	5.9%	3.3%
Foot	77.9%	12.8%	3.1%	3.3%	2.8%
All surgery units	71.5%	15.1%	4.7%	5.2%	3.5%

2.2 OR performance

The SMK evaluates the performance of the OR-complex with the aid of Key Performance Indicators (KPIs), influenced by their underlying process indicators. First we give a list in Table 2.3 of the KPIs and a list of the underlying process indicators in Table 2.4. Then we elaborate on the KPIs in Sections 2.2.1-2.2.4 by analysing their current performance and discussing their intended behaviour, and we briefly explain the performance of the process indicators in Section 2.2.5.

Table 2.3.: List of performance indicators used by the SMK

Key Performance Indicators
Productivity (Section 2.2.1)
Correct planned end-times of sessions (Section 2.2.2)
Access times and invitation times (Section 2.2.3)
Cancellations (Section 2.2.4)

Table 2.4.: List of process indicators used by the SMK

Process indicators (Section 2.2.5)
Booking accuracy
Fill-rate
Net-usage
Gross-usage

The KPIs from Table 2.3 can roughly be linked to the three stakeholders. The hospital required a certain productivity to be cost-effective. The personnel aims for minimal deviation of their work times, which means the planned end-times should be realised as often as possible. The patients are best helped with a short access time, and patients are helped with a long invitation time, such that they know their day of surgery as soon as possible. Lastly, cancellations leads to dissatisfaction among the patients. Note that there is already a conflicting interest between the number of cancellations and the invitation time. The sooner the day of surgery is announced, the higher the probability that a cancellation will take place. There are more negative relations between these KPIs, for which trade-offs have to be made by the management team.

There are other performance indicators that are not in the scope of our research, such as workload of surgeons, and the balance in outflow to the recovery or PACU. The last one is considered very important in similar research [8], however, we do not consider this performance since this is already taken into account in the session roster and the SMK currently experiences no capacity issues at the PACU or recovery.

The values of the performance and process indicators that are analysed in Sections 2.2.1-2.2.5 are taken over the period of October 2019 up to and including February 2020, unless stated else. This period is chosen to start at October 2019 due to use of the new OR-complex, and this period is chosen to end at February 2020 due to the start of the COVID-19 Pandemic in March. The data from the orthopedic sessions and surgeries of the OR-complex in Nijmegen is used.

2.2.1 Productivity

The productivity is a business indicator, to incorporate the wishes of the hospital. The productivity of the operating rooms is defined by the SMK as the average number of surgeries per session. The performance differs per surgery unit, and also per performing surgeon. Table 2.5 shows the average number of surgeries per session per surgery unit, and the average number of surgeries per session over all surgery units.

Table 2.5.: The productivity per surgery unit in Oct19-Feb20.

Surgery unit	Average
Back	[redacted]
Hip	[redacted]
Knee	[redacted]
Shoulder (BE)	[redacted]
Foot	[redacted]
Total	[redacted]

The average productivity differs per surgery unit: for back surgeries on average [redacted] surgeries can be performed, whereas for knee surgeries on average [redacted] surgeries can be performed. The SMK aims to perform on average [redacted] surgeries per session. This goal was set after the renovation of the OR-complex. From Table 2.5 we can conclude that this goal has not been realised yet, since the average productivity equals [redacted]. Before the renovation, the goal and the realization were in line with [redacted] surgeries.

2.2.2 Correct planned end-times of sessions

The correct planned end-times of sessions is an employee welfare indicator, to incorporate the wishes of the personnel. The SMK defined this KPI to be the percentage of the sessions with realised end times in a 15 minutes range from the planned end times. The norm is absolute, whereas a relative norm (the deviation in relation to the session duration) arguably also could have been used. Because the session durations are almost always the same, an absolute norm has been chosen by the SMK. In this research we sometimes refer to the KPI of correct planned end-times of session by using the term of booking accuracy of the session.

At this moment the norm for having the correct end time of the sessions comparing to the planned end times is set to 40%. This means that 40% of the realised end times of the sessions should be in the range of 15 minutes from the planned end times. In January and February of 2020 percentages of 27.9% and 20.3% were realised, respectively, shown in Figure 2.5.

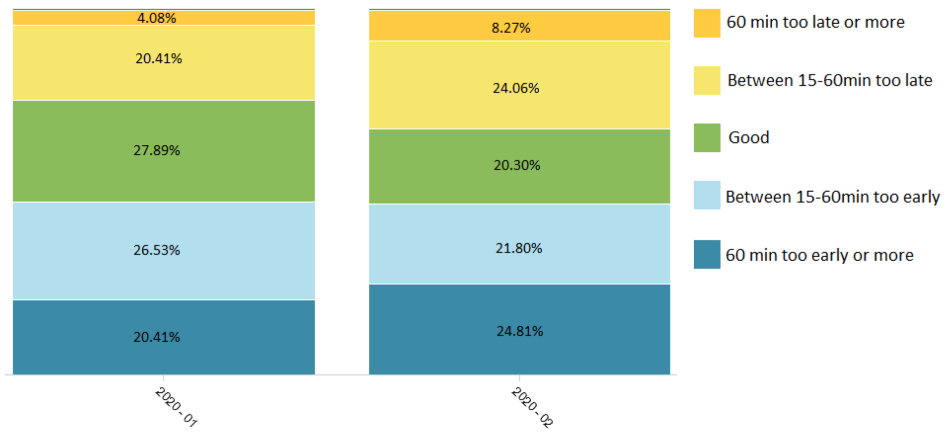

Figure 2.5.: Percentage of the sessions in January and February 2020 that ended in a specific range from the planned time.

The data shows that often the starting time of the OK differs from the norm of 08:06h, on average start times of sessions are 8 minutes earlier than planned. The deviation is caused by the fact the one anaesthesiologist is responsible for the start of two surgeries. Starting earlier is not a problem, however sessions ending too early due to this is not desirable. In Figure 2.5 it can be seen that it occurs more often that sessions end too early than too late, this is consistent with the utilization.

The main cause of earliness and/or tardiness of the sessions is that the realized surgery duration differs from the scheduled duration. The SMK uses the booking accuracy as a process indicator for the quality of forecasted surgery time. In Section 2.2.5 the booking accuracy will be explained further.

2.2.3 Access times and invitation times

The access times and invitation times are patient welfare indicators. The access time represents the time that a patient has to wait for his or her surgery once the patient enters the waiting list. The invitation time is the time between the moment that the patient receives the surgery date and the day of surgery itself. The definitions are visualised in Figure 2.6.

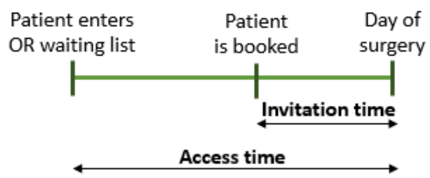

Figure 2.6.: Graphical interpretation of the access- and invitation time.

The KPI access time is introduced to make sure that patients are treated within their maximal internal access time. The maximum internal access time is the maximum time from the moment that a patient receives an OR-ticket until the patient goes to the OR for surgery. The goal set by the SMK is to treat patients within their maximum internal access times. The KPI invitation time is introduced to prevent surgeries from being scheduled last minute. The goal set by the SMK is to make sure that [redacted]% of the patients know their surgery date six weeks in advance.

In Table 2.6 the average invitation times and access times per urgency category are shown for the period of 2018 and 2019. Note that these are the average values, taken over all product groups, whereas performances differ per type of product group, and that the average can be influenced by registration errors, that occur most often for acute patients. Table 2.6 also shows the percentage of patients that was treated within the maximal internal access period per urgency category. Notice that the average access times for acute patients is higher than the norm, whereas most of the acute patients have been treated within their maximum internal access times. It is important to treat patients within their maximal internal access time, to provide the best service. It is difficult to save exactly enough capacity for uncertain future arrivals, therefore the SMK has set the norm to have [redacted]% of the patient for each urgency category to be treated within the maximal internal access times. This norm was realised for the acute patients in the period of Oct-19 until Feb-20.

Table 2.6.: Average access times and invitation times per urgency category (2018 and 2019) and the percentage of patients that were treated within their maximal internal access period per urgency category for the period of Oct-19 until Feb-20.

Urgency category	**Average access time** (days)	**Average invitation time** (days)	**Percentage treated within their maximal internal access times**
Elective (180days)	[redacted]	[redacted]	[redacted]%
2 Months (60days)	[redacted]	[redacted]	[redacted]%
1 Month (30days)	[redacted]	[redacted]	[redacted]%
Emergency (14days)	[redacted]	[redacted]	[redacted]%
Acute (2days)	[redacted]	[redacted]	[redacted]%

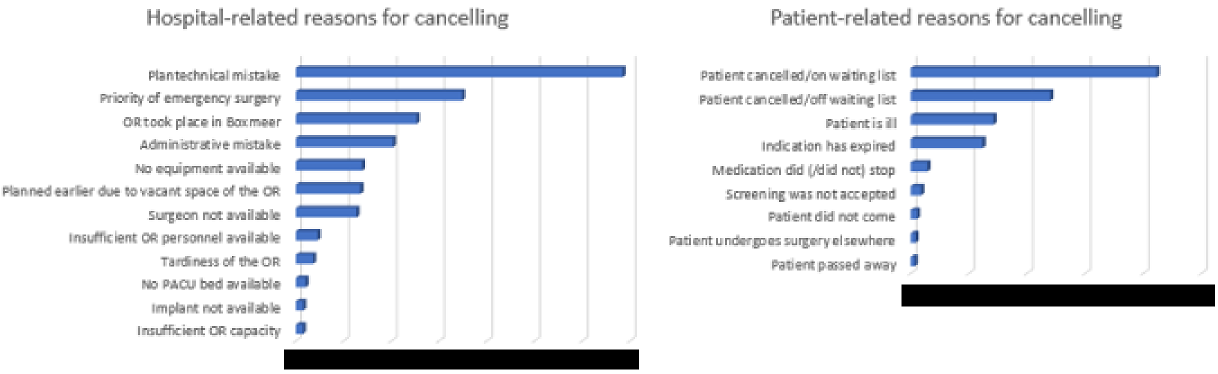

Figure 2.7.: Hospital-related (left) and patient-related (right) reasons for cancelling in percentage of the total hospital or patient cancellations, respectively, for the period Oct19-Feb20.

2.2.4 Cancellations

The KPI cancellations represents the total number of cancelled appointments in a certain period. The number of cancellations is a patient welfare indicator. This KPI is introduced to reduce the number of cancellations, since cancellations could lead to dissatisfaction by patients, and could lead to unused OR capacity.

Cancellations can be made on the surgery day itself (online), or before the day of surgery (offline). The number of offline cancellations is higher than the number of online cancellations: In the period of October 2019 until February 2020, there were on average [redacted] offline cancellations per month and [redacted] online cancellations (Table 2.7). These surgeries can be cancelled by the patient, or by the hospital. Table 2.7 also shows the percentages of the online and offline cancellations that were cancelled by the patient, or by the SMK. On average more cancellations are made by the SMK than by the patient.

Table 2.7.: Monthly average number of online and offline cancellations in the period of Oct19-Feb20.

	Cancellations	**Patient related reason**	**Hospital related reason**
Online	[redacted]	[redacted]%	[redacted]%
Offline	[redacted]	[redacted]%	[redacted]%
Total	[redacted]	[redacted]%	[redacted]%

Several reasons for cancelling exist, Figure 2.7 shows on the left the hospital-related cancellations reasons in percentage of the total hospital-related cancellations. Plan technical mistakes are the most common reason for cancellations done by the hospital. Plan technical mistakes are situations whereby the schedule is already too full too early in the planning process to meet the agreed access times of other surgeries. Second highest reason are higher priority emergency that need the time of lower priority surgery, which results into a cancellation of the lower priority surgery. This happens for both the online and offline planning.

Figure 2.7 shows on the right the patient-related cancellations in percentage of the total patient-related cancellations. The most common reason is a change in surgery date proposed by the patient. Note that the second reason, a patient cancels and wants off the waiting list, results into vacant time at the ORs, and a smaller waiting list. Patient cancellations are currently difficult to predict, because the underlying causes are unknown,

2.2.5 Process indicators

The KPIs are affected by underlying process-indicators. In this section we will discuss and analyse these indicators, and we will comment on their influence on the KPIs. The process-indicators are dealt with in the following order: Booking accuracy, Fill-rate, Net-usage, and Gross-usage. A summary of the performances of the process indicators is given in Table 2.9, at the end of this section.

First, the booking accuracy. The booking accuracy represents the percentage of the total number of surgeries with realised end times in the range of 15 minutes from the planned end times. This means the norm is absolute, whereas a relative norm arguably also could have been used. The booking accuracy for January and February 2020 for orthopedic surgeries of the OR-complex was respectively 36.2% and 34.3%, whereas the norm was set to 50%. Around 34% of the surgeries is underestimated with a minimum of 15 minutes and 30% is overestimated with a minimum of 15 minutes. There are some extreme points where the realised times deviate more than an hour from the planned times. Different factors could be appointed to explain the wrong estimation, such as type of surgery, operating surgeon, certain patient characteristics, etc. Improving the booking accuracy will reduce the probability of sessions ending late or early and with that reduce the number of online cancellations. Eventually this could lead to a higher productivity and lower access times.

Second, the fill-rate. The fill-rate represents how much of the available capacity of a session is filled with procedures at the start of the operating day. Most of the sessions have a capacity of 510 minutes. The norm for the fill-rate for each OR is set to [redacted]%. The fill-rate can be larger than 100%, however this is not preferred. The fill-rate differs per OR, since the seventh OR is available for emergency patients, where the other ORs are mostly filled by patients with lower priority. The average fill-rate for October 2019 up to and until February 2020 for OR 1-6 and for OR 7, the emergency OR, are given by Table 2.8. OR 7 has a significant lower fill-rate. A fill-rate less than 100% means that not all capacity is fully booked. This is overall the case, due to the fact that the planning department always try to book less than 100% of the capacity, to incorporate uncertainties during the day. A higher fill-rate leads to a higher productivity, which in turn leads to lower access times. However, a higher fill-rate could also lead to more cancellations, since there is less capacity available for uncertain arrivals of high priority patients.

Table 2.8.: Average fill rates per OR for the months October-19 till February-20.

	Oct-19	Nov-19	Dec-19	Jan-20	Feb-20
OR1-6	[redacted]%	[redacted]%	[redacted]%	[redacted]%	[redacted]%
OR7	[redacted]%	[redacted]%	[redacted]%	[redacted]%	[redacted]%

Third, Net-usage and Gross-usage. The net-usage represents the total time the means are used within or without the operating time by a patient, excluding the change-over time, compared to the operating time. The net-usage of the OR should get closer to the norm set by the SMK. Due to the change-over time this KPI cannot get close to 100%, and therefore the SMK chose to set the norm to [redacted]%. This research will not focus on improving change-over times, but better forecast of start- and end times could improve change-over times. The gross-usage represents the total time the means are used within or without the operating time by a patient compared to the operating time. The gross usage of the OR should get closer to the norm of [redacted]% set by the SMK. This process indicator should be as close as possible to 100% gain maximum productivity and correct end times. The gross-usage for January and February 2020 was [redacted]% and [redacted]%, respectively, whereas the norm was set to [redacted]%. The gross-usage in operating time, meaning all surgeries that were done in the planned operating time (8:00-16:30) plus change-overs divided by the planned operating time, was in both January and February 2020 [redacted]%, with norm [redacted]%. This means that more than one fourth of the total operating time is unused, this results in unused OR-capacity. The net-usage in January and February 2020 was [redacted]% and [redacted]%, respectively, both close to the norm of [redacted]%.

It should be noted that even though the net-usage is at norm, the amount of surgeries per session is at present not as budgeted. Where in the old OR complex on average [redacted] surgeries took place, the plan was to increase this to [redacted] surgeries per session due to a decrease of change-over time. This increase of the amount of surgeries per session has not been realised yet and so this increase is not reflected in the norm of the net usage at this moment.

To include a graphical representation of time of the sessions used for the different activities that take place during the sessions, we show Figure 2.8. The figure shows the utilization of the OR over 2019 and over the first two months of 2020, divided into these activities that take place during a session, where 100% equals the total session time. One can see that the change-over time has decreased, possibly due to the new OR-complex. The preparation time did however increase, this can be caused by the fact that patients have to be lifted up to a table in the new operating rooms, whereas mobile tables were used first.

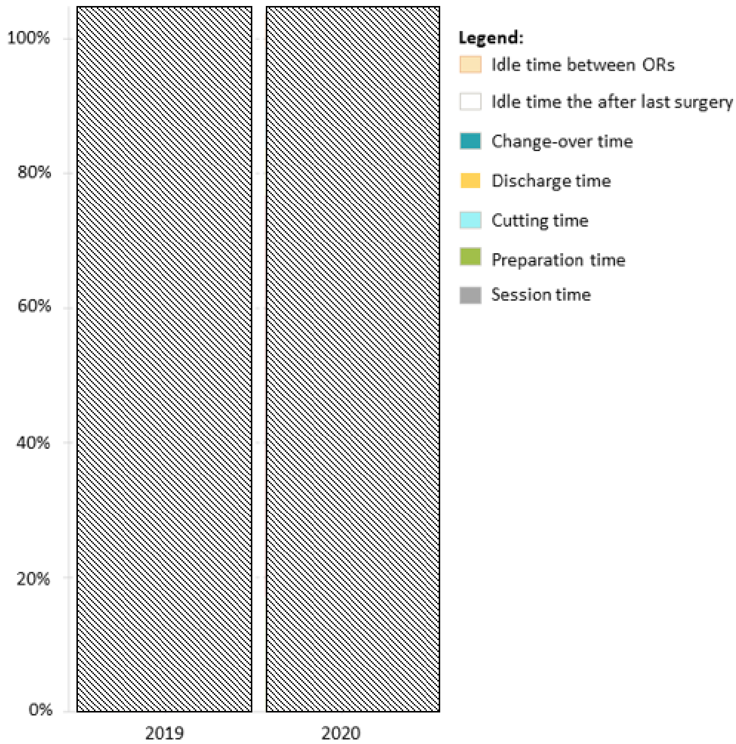

Figure 2.8.: OR-utilization in 2019 and the first two months of 2020.

In this research the OR-patient booking policy will be analysed on the key performance indicators through simulation, to conclude whether or not a smart booking policy can improve on OR-performance. The performance of the process-indicators for the proposed booking policy will not be analysed in this research.

Table 2.9.: The realisation of the process-indicators of the OR.

Indicator	January 2020	February 2020	Norm
Booking accuracy	36.2%	34.4%	50.0%
Fill-rate	[redacted]%	[redacted]%	[redacted]%
Gross usage	[redacted]%	[redacted]%	[redacted]%
Gross usage in operating time	[redacted]%	[redacted]%	[redacted]%
Net usage	[redacted]%	[redacted]%	[redacted]%

3 Literature

This chapter discusses the related literature on two topics. First the operations research literature relevant to our problem (Section 3.1). With Section 3.1 we first discuss different steps and different approaches in OR scheduling in Section 3.1.1, then we consult literature on scheduling methods that can decrease the variability of the schedule in Section 3.1.2. In Section 3.1.3 we explain that our type of problem can be modelled as an MDP and we discuss the application of ADP for obtaining approximate solutions. Second, we discuss the literature on the forecasting of procedure times in Section 3.2. We use this literature, since one of the goals of less realised deviation of the end times of sessions, could partly be achieved by increasing the prediction accuracy of the surgery time. We will conclude the literature and place our contribution in context of previous work in Section 3.3.

3.1 Operations Research

3.1.1 Operating room scheduling

The complexity of the surgical scheduling problem has often led researchers to concentrate on one part of the overall process at a time. The surgical scheduling problem can be segmented into four stages [8]. First, the total operating room capacity is determined and allocated to various surgical specialties. Second, the a cyclic master or block schedule is created that allocates specific blocks of OR time to each surgical specialty. Third, the scheduling of patients is done, to determine the number of surgeries of a session. At last, the appointment scheduling problem is involved, to assign specific start times, and thus determine the order, of the surgeries in each session. Part of this step is done before the day of surgery (offline planning), and during the day of surgery (online planning). Our research focuses on offline planning of the last two stages of the surgical scheduling problem. The third stage is often mentioned as the advanced scheduling problem, and the fourth stage as the allocation scheduling problem [3].

Since the second stage passes information onto the third stage, we incorporate literature on the second stage. An example of an approach for the second stage of surgery scheduling is Master Surgery Scheduling (MSS). The goal is to maximize utilization and level capacity usage of resources, also taking into account the supporting facilities as the PACU and the ICU. The constraints exist of capacity constraints (probabilistic) and all surgery types must be planned. Eventually MSS tries to determine the length of a surgery scheduling cycle and a list of surgery types for each OR-day. The generation of OR-day schedules with as goal the capacity utilization, can be solved as an ILP with column generation and rounding [9]. The capacity of the operating rooms at the SMK are not assigned to surgery specialties in a cyclic manner. In our research we know the number of assigned sessions to surgery specialties, however, we do not know the specific day of these sessions to take place.

The importance of selecting surgery groups, which are clustered surgery procedure types that share comparable characteristics (e.g. expected time, specialty, expertise of surgeons), is shown by [10], where an holistic approach is given for surgery scheduling making use of MSS. A single step model where bed usage variation is minimized and the OR utilization is maximized is proposed. They concluded that scheduling surgery groups, instead of groups only based on surgical specialty, reduces probability of overtime, and variation in bed usage. This shows that OR blocks should not be allocated to surgery specialty, but to so called surgery groups.

The third stage, the advanced scheduling problem, becomes complicated due to uncertainties during the process. For the most part, the OR schedule is affected by the uncertain arrival proces of non-elective patients. Elective patients are known patients that can be scheduled, however, non-elective patients are patient arrivals which in most cases are impossible to predict in advance and will take a random amount of time which should be taken into account in the schedule. To deal with non-elective patients, different methods have been researched. Van Essen et al. [11] incorporate a break-in moment, which is a time point when an elective surgery is finished, presenting the opportunity to serve a waiting non-elective patient in the freed-up OR. Others use a dedicated OR to non-elective patients. A combination of dedicated and flexible ORs outperforms the use of a dedicated OR or only flexible ORs, in terms of patient waiting time and OR overtime [12]. Wullink et al. [7] claim that closing emergency operating rooms improves efficiency, to incorporate emergencies they advice to save some capacity divided over all ORs. Currently the SMK uses OR-7 as a dedicated OR in the afternoon, there is no capacity reserved at other ORs for non-elective patients. The SMK has less need for capacity for non-elective patients, since the percentage of non-elective patients for orthopedics is much less than other surgery specialties.

Deviations from the schedule on the day itself are due to uncertainty of emergency arrivals, a variable workload in downstream units, staff unavailability, equipment failure, late arrival of patients or staff, patient no shows, and deviation of surgery times [1]. The forecasting of the surgery times, such that less deviation will take place on surgery end times, will be discussed in Section 3.2.

Astaraky et al. [8] created a model that, given a master schedule that provides a cyclic breakdown of total OR availability into specific daily allocations to each surgical specialty, provides a scheduling policy for all surgeries that minimizes a combination of the lead time between patient request and surgery date, overtime in the operating room, and congestion in the wards. They formulated the problem as a Markov Decision Process (MDP), for which a direct solution for a realistic instance became intractable, and therefore the MDP was solved through the least squares approximate policy iteration algorithm. We can relate our problem to [8], however we do not have a cyclic master schedule and the objectives differ. We will dive more into the literature on MDP in Section 3.1.3.

Roland et al. [13] propose a two stage planning for operating room scheduling, combining stage three and four, where due to the separate models, a sub-optimal solution can be achieved. They start the planning stage during which an operating day is fixed for each surgery. This planning stage is next followed by a scheduling stage that determines the starting time of each operation occurring on a given day. They wish an approach focussed as much on human resources as on economic factors. The approach incorporates meta-heuristics for realistic instances, since their exact approach with the Mixed-Integer Programm (MIP) becomes unusable in practice. Their approach was done in a deterministic fashion, excluding emergencies, whereas we include emergencies in our research.

The allocation problem has been researched by Denton et al. [14]. They define the order of the day on three measures: waiting time, idling time, and tardiness of the session in an offline fashion. They conclude, scheduling in shortest variance first (SVF) creates a high potential, regarding the previously mentioned measures on current used policies.

To our knowledge, considering the objective of minimal deviation from the planned end times, has not been incorporated in surgical scheduling before. Methods for including this, are mentioned in Section 3.1.2. We proceed in adding this objective, next to meeting patient lead-times and productivity, in the advanced scheduling problem, where an MDP formulation will be used (see Section 3.1.3).

3.1.2 Earliness and Tardiness

The main cause of early and tardy sessions is that the realized surgery duration differs from the scheduled duration. In part this can be solved by developing methods and models to predict surgery durations more accurately, which is discussed in Section 3.2. Due to the stochastic nature of processes of during an OR day, there will always be some variability that has to be taken into account [15]. This section focuses on scheduling methods for dealing with the variability of the schedule.

The Earliness/Tardiness (ET) problem aims to schedule jobs on machines in such a way that the total weighted deviation from their due dates is minimized. In our problem, jobs can be seen as surgeries and machines can be seen as operating rooms. An approach to minimize earliness and tardiness (ET) of a schedule with multiple machines is done by Otten et al. [16]. They aim to improve the robustness of the schedule by considering the expected deviation from the intended schedule in terms of the expected earliness and tardiness of surgeries. They showed that the Shortest Variance First (SVF) approach improves the schedule regarding the ET of sessions for multiple ORs, if the jobs are normally distributed. Log normal distribution seems to model the uncertainty of surgical procedure times the best [17]. It was not proven that for log normal distributions SVF is optimal, however a simulation suggested this could hold. They show that a secondary objective could be introduced where Otten et al. used the make span or the minimization of overtime. For this, they group surgeries on standard deviation, and assume that interchanging surgeries within the same standard deviation group between operating rooms, does not influence the total ET costs that much. With this, a finite group of surgeries can be scheduled with minimal ET costs and minimal deviation from the capacity, by using the make span. Due to the dynamics of the patient planning process, the use of ET-scheduling should be incorporated in a dynamic model.

Van Houdenhoven et al. [18] investigated applying the bin packing method and portfolio techniques, these techniques incorporate minimal total slack of all operating rooms together, to surgical case scheduling, and showed that smart scheduling of procedures with specific variances can improve on current methods, assuming that surgery times are independent and normally-distributed. In order to incorporate variations in surgery times, an adjustable parameter is used, that allows a certain amount of idle time for each operating room.

Hans et al. [19] consider the robust surgery loading problem (“stochastic knapsack” problem), which concerns assigning surgeries and sufficient planned slack to operating room days with an objective of maximizing capacity utilization and minimizing the risk of overtime. They propose several heuristics (i.e. Base solution determination using First Fit, Longest processing time dispatching, certain sampling procedures) to exploit the portfolio effect, thereby minimizing the required slack, and eventually show that the operating room utilization could be improved. The best constructive approach found to be was regret-based random sampling. They observed that as a result of the portfolio effect, surgeries with similar duration variability are often clustered on the same OR-day. This is called risk-pooling, which means clustering high risk surgeries and clustering low risk surgeries. This would lead to some sessions having a high variability, whereas most of the sessions would have lower variability. In total, the risk-pooling approach would lead to less variability in the complete schedule.

Both risk-pooling and ET-scheduling could be of use to improve the schedule on achieving less variability. For this, Section 4.2 goes into more detail of the risk-pooling approach and we will analyse this method for an SMK testcase. Section 4.3, will go into more detail of the ET scheduling approach and analyse this method as well for an SMK testcase. Eventually both methods will be compared on performance in Section 4.4.

3.1.3 Markov Decision Theory and the solution approach

The offline booking of patients exists of decisions that are taken in compliance with the uncertain future. Taking into account future demand and uncertain events over time, involves an optimization over time, which can be represented by a Markov Decision Problem. The basic concepts of Markov chains can be found in Ross [20]. Markov chains are useful due to the property that no information prior to the current time step is needed to make predictions for the states in the future, this is called the Markov property.

Well-known direct solutions methods for MDP are value iteration, policy iteration, and the linear programming formulation [21]. Large state- and action spaces, known as the curses of dimensionality, often make a direct solution to the MDP impossible. The field of approximate dynamic programming has been introduced to overcome the curses of dimensionality. An overview and explanation of several ADP techniques is given by [22]. ADP approaches on patient planning have been done before [[8],[23],[24],[25],[26]].

The ADP approach is an extension of the direct solution methods. In 2003 De Farias and Van Roy [27] introduced ADP in linear programming. The direct linear programming approach yields two major problems as a consequence of the curses of dimensionality. First, the large state space results into too many variables, this issue is solved by the ADP approach through introducing an approximate value function. Several basis functions can be used to approximate the value function, such as affine functions, splines, and polynomials. When choosing we should consider the approximation accuracy of the linear combinations of the basis functions and we should consider the effort involved in computing with such approximations [28]. Second, the large action space, together with the state space, results in too many state-actions pairs, which results in a huge number of constraints. The ADP approach to linear programming can overcome this issue by using column generation methods [[29],[30]]. Sauré et al. [23] used the ADP approach to linear programming to solve a dynamic multi priority patient scheduling problem, and made use of an approximate value function and column generation to overcome the curses of dimensionality.

Our allocation problem will face the curses of dimensionality. In this paper we will deal with them through the ADP approach to linear programming. To keep the problem linear we will introduce an approximation to the value function. We will solve the Approximate Linear Program (ALP) through column generation.

The ADP approach to linear programming can be used for MDP, since each MDP induces a linear program [21]. Making use of the column generation algorithm creates a new optimization problem, called the pricing problem. The type of program of the pricing problem, depends on the type of cost function that is defined for the Markov decision process. Linear cost functions are preferred, since efficient linear solvers are available and non-linear solvers are computationally expensive. Non-linearity increases the complexity of the problem. Some non-linear objectives can be made linear by using LP-modelling tricks. For separable functions, piece-wise linear approximations can be used to solve the problem of non-linearity [31]. In this research we will deal with a pricing problem that is non-linear. Information on Mixed Integer Non-Linear Programming (MINLP) can be found in Section 5.3.5.

3.2 Forecasting procedure times

It is found that variability in the surgery time with patient profile needs to be considered for a realistic scheduling of the operation rooms [32]. Surgery times are difficult to predict, because for some surgeries the magnitude of the procedure only becomes apparent once the surgery is already in progress. Next to this, surgery times depend on different complex factors, e.g. the characteristics of the patient, the surgeon, and the surgical team [1]. Combes et al. [32] made clear that classifying patients is difficult: one patient may suffer from the same condition as another, but will not necessarily receive the same treatment.

Devi et al. [33] make use of several predictions methods for surgery times. They consider the method of multiple linear regression analysis (MLRA), and the machine learning methods of artificial neural networks (ANN), and adaptive neuro fuzzy inference system (ANFIS). The forecasting is done by taking into account the surgical environment, including variables such as surgeon's experience, staff's experience, but also using patients characteristics as patient's precondition and patient's age. At last the duration of the surgery is included. The ANFIS model is found to out-perform the other two models. Machine learning has been used for different purposes in health care. Shown by [34] machine learning can be a good addition on current prediction methods, their particular interest was identifying surgeries with high risk of cancellation. Fairley et al. [35] used machine learning for recovery length prediction. The features ranked from importance beginning with the most important were; type of procedure, weight and age of the patient, scheduled post-operation destination, service (the surgical specialism performing the procedure), the scheduled procedure length, the patient class (inpatient/outpatient), and gender. A research on analysis of variance of surgery time found that team composition, experience, and time of the day to be the most significant factors, ignoring factors relating to patients [36]. Neural networks can find an unknown underlying structure in the correspondence of the different variables and thus a better prediction. The disadvantage is the flexibility of neural networks, it is susceptible for overfitting [37].

Research done by the ZGT [38] comparing well known regression methods to machine learning for predicting surgical case durations concluded that neural networks don't have to outscore simple regression methods. It was noted that possibly some important features were missing . Master et al. [39] predicted pediatric surgical durations, a case including extreme variations within the group. They showed that machine learning methods combined with surgeon's estimations of surgery times could provide better estimates than just averaging or surgeons' estimation alone. However, accurate predictions without surgeons' estimations would be preferable, since their time is expensive which leads to increased costs.

Prediction methods based on data are useful to achieve better estimations and to reduce the work involved. For adult surgeries it was shown that these methods lead to modest improvements in accuracy over human experts [40]. Evaluating model predictions mathematically can be done in different ways such as k-fold cross validation, Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE) [36].

We realize that a lot of research has been done on forecasting methods for surgery times. We refer to [37] for a review of logistic regression and artificial intelligence models used for medical purposes. Due to the limited time of this research, and the fact that the forecasting project is seen as a less important goal than achieving a scheduling method that decreases variability of the schedule, we choose to only analyse the k-nearest neighbour (kNN) approach. This approach fits to the current method of forecasting procedure times by the SMK, since surgeons change proposed surgery times by HiX based on their experience of similar surgeries performed in the past. Therefore, the kNN method is intuitively explainable.

The advantage that k-nearest neighbors has over other algorithms is the fact that the neighbors can provide an explanation for the classification result, whereas black-box models do not provide such an explanation. The disadvantage of k-nearest neighbors lies in the calculation of the case neighborhood. Important is to define a metric that measures the distance between data items equally. In most applications these metrics are defined by trial and error, such that the relative importance of data components is reflected in the metric [37].

The book of Friedman et al. [41] provides information on statistical learning. In case a lot of explanatory variables are available, the method of principal component analysis (PCA) could be used to reduce the number of variables, by combining the variables that have a similar influence on the outcome. PCA cannot be used for categorial variables, in such a case multiple correspondence analysis (MCA) could be used. The choice of k in the kNN method is often determined by cross validation.

Improving the forecast of the procedure times, results into more accurate predictions and therefore less variability in surgery times during the day. This has no influence on the performances of the model, and therefore we will treat the forecasting problem as a separate problem from the model. In Section 4.1 the use of the kNN method on SMK data is discussed.

3.3 Positioning of the research and its contribution

Our contribution is the development of a method for offline patient scheduling over multiple weeks, taking into account multiple uncertainties in the future and optimizing over a combination of equitably patient booking with minimal access times, minimal deviation from the planned end-times and maximal utilization. Especially the objective of minimizing the ET costs of the planned end times has not been taken into account yet by previous research.

Literature shows that patient scheduling can be divided into roughly two streams: Advanced scheduling and allocation scheduling. Allocation scheduling follows after advanced scheduling; the patient group has been chosen by advanced scheduling and now patients can be matched with the resources and surgery times can be given. We propose the same order of decision making in patient booking for the SMK. The first decision is to assign patient groups to an OR session (advanced patient scheduling). The second decision exists of assigning the specific patient from the patient group to the OR session. See Figure (3.1 for a visual representation of the two consecutive decision for patient booking. This splitting has the extra advantage of decreasing the size of the problem. Information can now be stored per patient group, instead of saving information per patient.

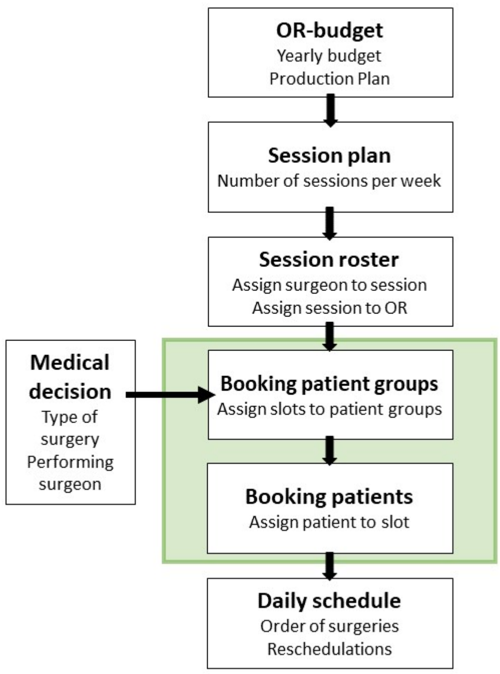

Figure 3.1.: Proposed planning process

We formulate a Markov decision problem for the advanced patient scheduling and use ADP with an application to linear programming, which we solve through column generation, to find approximate policies. For the allocation scheduling a simple policy-rule is suggested. The forecasting of procedure times is seen as a side-project, the main focus lies on creating a robust schedule.

4 Methodology

One of the extensions of this research to previous research, is to introduce the minimization of the deviation of the expected end times of sessions into the dynamic OR scheduling process. In literature we found several methods to decrease the variability of the end times of sessions. The main cause of early and tardy sessions, is that the realised surgery times of the surgeries taking place in the sessions, differ from the planned duration. To improve this, we did a literature search on forecasting procedure times in Section 3.2. Section 4.1 uses the outcomes of that literature search and tests the method of k-Nearest Neighbours (kNN) on the SMK data. Procedure-times will always be of stochastic nature [15], and therefore we also seek to find methods to decrease the variability of the schedule by smart scheduling. The first method that came up from literature, called risk-pooling is discussed and applied to SMK data in Section 4.2. The second method, called ET-scheduling, is discussed and applied to SMK data in Section 4.3. Section 4.4 compares both methods. Section 4.5 concludes on the methodologies and relates the methodologies to the model that is proposed in Chapter 5.

4.1 Improving the booking accuracy

As we discussed in Section 3.2, several methods exist for forecasting procedure times, based on historic data (i.e. neural networks, linear regression, etc.). Improving the current method of forecasting of the SMK could lead to more accurate predictions and thus less deviation from the intended schedule. We will first state the current surgery prediction method use by the SMK, then we will analyse its accuracy, and eventually we will come up with a list of variables that we will use for the $k-$NN method to improve the prediction accuracy. The data we use in this section exists of the surgeries performed by one specific surgeon in the year of 2019, including 382 surgeries in total.

First we analyse the accuracy of the current prediction methods. The current prediction method exist of three steps, shown by Figure 4.1, where HiX Chipsoft is a supportive company for ICT, that base their prediction on the average surgery times of the twenty most recent performed similar surgeries. Taking the median would seem more suitable, due to the assymetric distribution of surgery times. Not including the second step, the prediction by the surgeon, would save useful time of the surgeon. Therefore it would be useful if the prediction by the $k-$NN method would be as good, or even better than the prediction by the surgeon.

To analyse the current predictions, we created plots that show the difference between the proposed surgery time by HiX and the realised surgery time, and the proposed surgery tim by the surgeon and the realised surgery time. Figure 4.2 shows a point cloud for the four most common treatment codes. Figure 4.3 shows the differences for the eleven most common treatment codes. In the figures we observe that in most cases the surgery time is underestimated. We perform an one-tailed hypothesis test for this observation, under the assumption of normal distributed independent surgery duration. We want to reject the null hypothesis that the difference of the means of the realised surgery time and the estimated surgery time is less than or equal to zero. Making use of the t-test, with a significance level of 5%, we cannot reject the null hypothesis. Mathematically we cannot state that

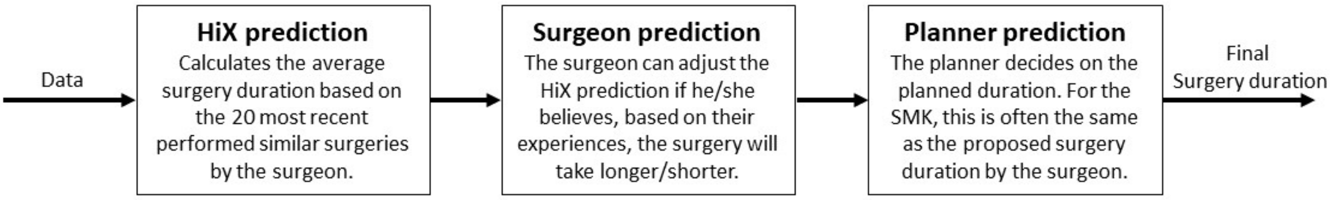

Figure 4.1.: Current roadmap at the SMK to determine the surgery time.

the surgery time is underestimated, whereas Figure 4.3 (right) suggests this is the case. Next to this we notice that the prediction proposed by the surgeon is more accurate than the prediction by HiX. The Mean Absolute Error (MAE) indicates this as well, since the MSA for the HiX prediction equals 21.6, whereas the MSA for the surgeon's prediction equals 17.0, see Table 4.2.

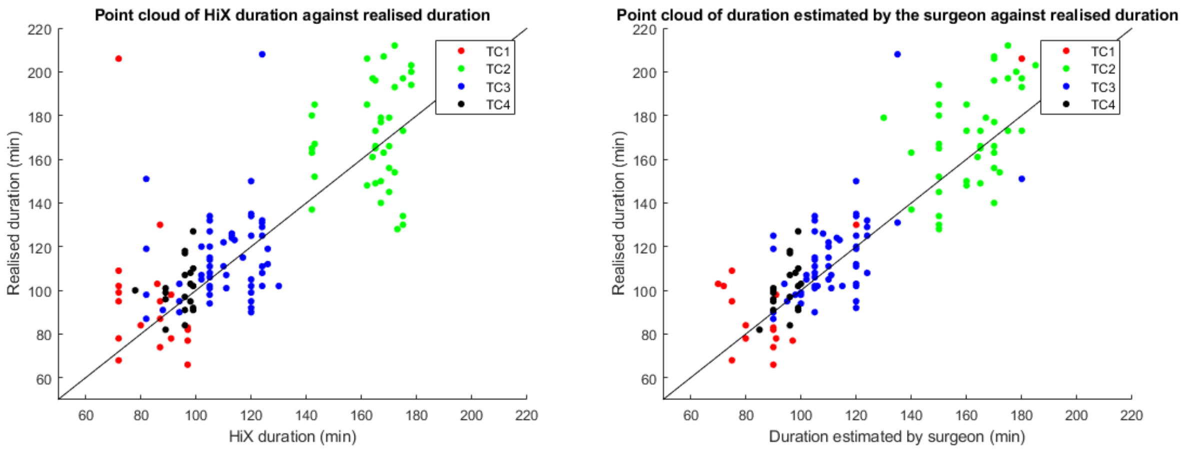

Figure 4.2.: Point clouds of the realised surgery time against the HiX prediction (left) and the surgeon prediction (right).

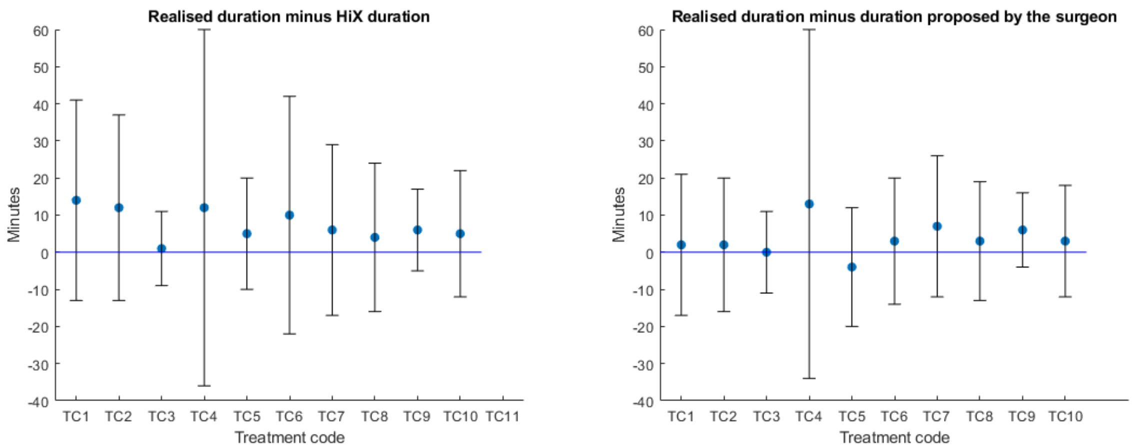

Figure 4.3.: Mean deviation and standard deviation of the mean for the HiX prediction (left) and the surgeon prediction (right) against the realised surgery time.

A list of variables that could be of influence on the surgery time, is given in Appendix A.2. This list could be heavily extended. Measurements of a few variables can be used for this study. Our prediction problem includes the predicting (/explanatory) variables of Table 4.1, to predict the surgery duration.

Table 4.1.: List of variables available to predict the surgery time.

Predicting variables	**Range**
Treatment code	TC1-TC64
Product group/Surgery unit	Knee/Knee.Sp/Hip/Hip.Sp
Surgeon	S1
Priority (/urgency class)	$1-5$
Proposed surgery time by HiX	$20-228$
Expected surgery time by surgeon	$20-270$
Expected surgery time by planner	$20-290$
Age of the patient	$18-90$
Gender of the patient	M/F
Criterion variable	**Range**
Realised surgery time	$17-306$

In case of too many columns of data, Principle Component Analysis (PCA) for numerical data, or Multiple Correspondence Analysis (MCA) for categorical data, could be used to reduce the set of variables. With PCA we search for a linear combination of the original variables that describe as much as possible of the original variance. This means that components can be found that represent the variance of the original set, where these components are independent of each other. Notice that this approach leads to loss of information. We need PCA or MCA if we have too many variables in our dataset to perform the method of k-Nearest Neighbors, currently this is not the case.

The nearest neighbor approach is a supervised learning approach. The method is used to predict a new observation based on the values of the nearest neighbor observations. A training set is used to eventually predict the nearest neighbors of an observation. An advantage of the approach Nearest neighbor is that the method is intuitively explainable, that is, the trained object with characteristics most equal to the observed object is most likely to represent the observed object. A disadvantage of the kNN approach is that variables should be scaled equal, since the nearest neighbor is based on the distance. Therefore data has to be standardized, for example with the use of z-scores, or by using the range (i.e. calculating the ratio of the value relative to the maximum en minimum the values can take). Categorical data is difficult to scale, to solve this, we introduce dummy variables for each category existing in the categorical data. If a row belongs to the category, the dummy variable is equal to one, else the dummy variable equals zero. It seems intuitively wrong to compare surgeries of different treatment codes with each other, for this the weighted nearest neighbours method can be used to increase the importance of the treatment code, and keep the treatment code as one numerical variable with high weight.

The choice of k is important: Choosing k too small will increase the influence of noise and the results are going to be less generalizable, choosing k too large leads to a neighborhood that can exist of other classes. We use cross validation to find the best value of k for the $k-$NN approach. For this, we divide the data into five subsets, each having 20% of the rows of the data as test data. Next, we set k equal to some integer, and we perform the $k-$NN prediction five times, each time differing the train data and test data on the five subsets. Then, we compute the cross validated error, which is equal to the average MAE of the five $k-$NN predictions. We compute the cross validated error for multiple k, and eventually choose k equal to case with the lowest cross validated error. Note that the data set for cross validation to determine k should be independent of the data set for the final prediction, in order to get an generalised prediction. This would however split the data set even further, for which not enough data is available. Applying the method with independent sets for cross validation and prediction on the largest available data set, shows us that k remains the same, therefore we assume no independent sets are needed, even though this affects the general conclusion of the experiment.

We inspect the following cases for prediction:

(1) We handle the categorical variables, that is the treatment code, the product group, and the gender of the patient, through the use of dummy variables.

  (1-1) We use all predicting variables from Table 4.1.

  (1-2) We use all predicting variables from Table 4.1, except the treatment code, to see the influence of this variable on the prediction.

  (1-3) We use all predicting variables from Table 4.1, except the product group of the patient, to see the influence of this variable on the prediction.

  (1-4) We use all predicting variables from Table 4.1, except the priority of the patient, to see the influence of this variable on the prediction.

  (1-5) We use all predicting variables from Table 4.1, except the age of the patient, to see the influence of this variable on the prediction.

  (1-6) We use all predicting variables from Table 4.1, except the gender of the patient, to see the influence of this variable on the prediction.

(2) We use all predicting variables from Table 4.1, except the expected surgery time by the surgeon. If the prediction can become as accurate as the surgeon's prediction or even better, surgeon's work time can be saved. We handle the categorical variables, that is the treatment code, the product group, and the gender of the patient, through the use of dummy variables.

(3) We use all predicting variables from Table 4.1, however we specify the treatment code to the most frequent used treatment code, that is the one that occurred 59 times. We handle the categorical variables, that is the gender of the patient, through the use of dummy variables. The product group belonging to this treatment code exist of one type.

(4) We use all predicting variables from Table 4.1, however we specify the treatment code to the most frequent used treatment code, that is the one that occurred 59 times, and next to this we do not use the prediction variable of the proposed surgeon's surgery time, to see if we can outperform the surgeon's estimation. We handle the categorical variables, and the gender of the patient, through the use of dummy variables. The product group belonging to this treatment code exist of one type.

The results for the cases (1) and (2) are shown in Table 4.2, results for cases (3) and (4) are shown in Table 4.3, conclusions will be given afterwards. For cases (2),(3),and (4) some features can be left out as well, as is done for case (1). The conclusions of that will also be given afterwards. For each case k is chosen based on the cross validation analysis, where the train set each time existed of 80% of the data, and the test set of 20% of the data.

From Table 4.2 we can conclude with the use of the method of dummy variables for the treatment code, and the used set of variables, we cannot outperform the accuracy of the current planned surgery time. We notice that excluding one variable in the prediction, does not result into a significant change of the MAE. The worst estimation is done when the product group of the patient is excluded (case (1-3)), and the best estimation is done for case (1-4), when the priority of the patient is excluded. Next to this, we cannot outperform the prediction of the surgeon, since the MAE of (2) is higher than the MAE of the current surgeon estimated surgery time. Not taking into account certain predicting variables for case (2) will not results in a high improvement, only leaving out the priority of the patient would result in a better MSA, that is 17.70.

Table 4.2.: Results of the kNN method for the cases (1) and (2)

Case	Num. of Obs.	k	MSPE	MAE
$(1-1)$	382	8	622.03	17.88
$(1-2)$	382	13	624.15	17.67
$(1-3)$	382	11	639.19	18.45
$(1-4)$	382	10	595.36	17.40
$(1-5)$	382	8	642.24	18.00
$(1-6)$	382	11	617.80	17.94
(2)	382	9	612.17	17.81
Current planned surgery time			573.18	16.62
Current surgeon estimated surgery time			582.54	16.95
Current HiX estimated surgery time			966.38	21.57

Table 4.3.: Results of the kNN method for the cases (3) and (4).

Case	Num. of Obs.	k	MSPE	MAE
(3)	59	3	251.25	11.31
(4)	59	3	236.37	10.59
Current planned surgery time			268.12	11.31
Current surgeon estimated surgery time			279.81	11.41
Current HiX estimated surgery time			420.24	14.37

From Table 4.3 we can conclude that for the most common treatment type, we can improve the prediction compared to the current planned surgery time, the current surgeon estimated surgery time, and the current HiX estimated surgery time. Case (3) performs as well as the current planned surgery time. Case (4) even outperforms the current planned surgery time, where the only difference between Case (3) and (4) is, that Case (4) excludes the surgeon's estimated surgery time. Unfortunately, using $k-$NN per treatment code, is not applicable to each treatment code, since not enough data on the treatment code is available. Only 10 out of the 64 treatment codes were performed more than 10 times by the specific surgeon during 2019. Therefore, this method only works for treatment codes that are performed multiple times in the past. Based on the observations it is recommended to save data on more surgeries, and eventually improve the prediction of the surgery time per treatment code with the aid of the used prediction variables.

4.2 Risk pooling

This section explains the method of risk-pooling. In Section 4.2.1 we will show the improvements of the portfolio-effect to the current SMK planning. The goal of risk pooling is to reduce the total uncertainty of a schedule and thus minimize the expected deviation from the end times of the session..

We illustrate risk-pooling with the example given by Figure (4.4) and Table 4.4. Suppose we have two operating rooms, both having two surgeries scheduled. The first surgery on both days has an expected surgery time μ of 100 minutes, with a standard deviation σ of 10 minutes. For the second surgery this is 100 minutes and 50 minutes, respectively. We assume that the surgery times are independent and normally-distributed. Applying risk-pooling, means the we cluster the surgeries with similar variance. The hospital can incorporate some time, the planned slack, to reduce the risk over overtime. Let β denote the risk factor to deal with overtime. The risk factor is multiplied with the expected standard deviation of the session duration to calculate the planned slack time. The total planned slack before risk-pooling equals 102.0β, whereas the total planned slack after risk-pooling equals 84.8β. With this method, we have increased the available capacity of the sessions, since the planned slack time is reduced.

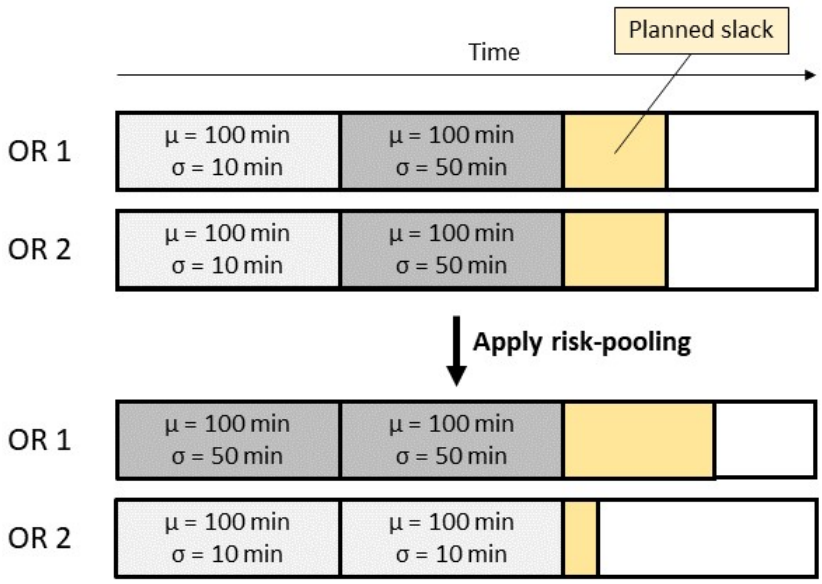

Figure 4.4.: Illustration of the risk-pooling method.

Table 4.4.: Values belonging to Figure (4.4), indicating that the total planned slack is less after applying risk-pooling.

	Standard deviation of the total duration	Planned slack	Total planned slack
OR 1	51.0 min	51.0β	102.0β
OR 2	51.0 min	51.0β	
- Apply risk-pooling -			
OR 1	70.7 min	70.7β	84.8β
OR 2	14.1 min	14.1β	

Mathematically speaking risk-pooling comes down to minimizing the sum over all the sessions of the standard deviation of the total duration for each session. This does not mean that the expected deviation of the end time for each session is minimal, but that the expected deviation of all the sessions together is minimal. The method of risk-pooling can thus decrease the total expected deviation of the end times of the sessions. What is noteworthy, is that the risk-pooling method causes some days to have a really high variance, and some days with really low variance, however, for the whole schedule, the risk of having overtime is decreased. Applying risk-pooling in practice would ask flexibility from the OR-personnel, and it can be communicated to the OR-personnel which days would have a high deviation in working hours, and which days not.

Within our approach we do not plan slack time, and thus do not make use of the risk factor β. We want to use full capacity of the session, for which the realised end time of the session has as little as possible expected deviation from the planned end time. With the use of risk-pooling, the total expected deviation over all sessions from the end times can be decreased.

With the aid of the Mixed Integer Non-Linear Program (MINLP) given in (4.1), the risk-pooling method can be implemented on a static deterministic version (i.e. solving at one moment, not taking into account uncertainties of future arrivals nor cancellations) of the scheduling problem. Sets, variables, and parameters for (4.1) are given by Table 4.5. We assume that surgery times are independent and normally-distributed, for the ease of calculating the total session duration, even though literature shows that surgery time is often log normally distributed [17]. Given the surgeries to be performed and the sessions, this program gives a solution that books each surgery into a session, that doesn't exceed the capacity of the sessions, and minimizes the total expected deviation from the planned end times of the session by risk-pooling. We make the assumption that standard deviation only depends on characteristics of the patient, and not on surgeon nor session. Note that the problem becomes infeasible if no combination of surgeries exists such that all surgeries can be booked into the session, without exceeding capacity. The capacity constraint is added since otherwise all surgeries would be booked on one session. We can solve (4.1) with a convex solver for non-linear programs, since the negative square root is convex, for this we have to adjust the program to obtain an objective with a negative square root.

Table 4.5.: Sets, variables, and parameters for (4.1)

Sets	**Description**
$\mathcal{S} = \{0, 1, \ldots, S\}$	Set of sessions, indexed by s, with S the number of sessions.
$\mathcal{P} = \{0, 1, \ldots, P\}$	Set of patients, indexed by p or t, with P the number of patients.
Variables	**Description**
$x_{s,p}$	Binary variable, indicating whether patient p has been booked on session s or not.
Parameters	**Description**
μ_p	Represents the expected surgery time of patient p.
σ_p	Represents the expected standard deviation of the surgery time of patient p.
C_s	Represents the capacity in minutes of session s.
CO	Represents the changeover time in minutes.

$$
\begin{aligned}
\min_{x} \quad & \sum_{s} \sqrt{\sum_{p} x_{s,p}\sigma_p^2} \\
\text{s.t.} \quad & \sum_{s} x_{s,p} = 1 \qquad \forall p \in \mathcal{P}, \\
& \sum_{p} \mu_p x_{s,p} \leq C_s \quad \forall s \in \mathcal{S}.
\end{aligned} \tag{4.1}
$$

4.2.1 Risk pooling applied to SMK data

To show the effect of risk-pooling in terms of the KPI 'Correct planned end-times of sessions', we apply risk-pooling to an arbitrary instance of the SMK. We take the schedule for one week of one surgeon. The specific surgeon had two sessions during this week. On the first session five surgeries were planned, and on the second session six surgeries were planned. We denote the stochastic variable indicating the surgery time of surgery i with Z_i, where $i = 1, \dots 11$, and we denote the stochastic variable for the total duration of a session as S_j, where $j = 1, 2$. We assume the surgery times are independent and normally- distributed, and we assume there are no further restrictions on interchanging surgeries within these sessions. We do not include the change-over times.

Information about the scheduled surgeries per session following the SMK schedule is given in Table 4.6. The distribution of S_1 and S_2 is the sum of the surgery time distributions of the surgeries of that respective session. For the normal distribution it holds that the sum of normally independent distributed stochastic variables is again normally distributed. Table 4.7 shows the schedule after applying risk-pooling to the SMK schedule.

Session 1	**Session 2**
$Z_1 \sim \mathcal{N}(41, 14^2)$	$Z_6 \sim \mathcal{N}(93, 17^2)$
$Z_2 \sim \mathcal{N}(48, 15^2)$	$Z_7 \sim \mathcal{N}(41, 14^2)$
$Z_3 \sim \mathcal{N}(50, 6^2)$	$Z_8 \sim \mathcal{N}(93, 17^2)$
$Z_4 \sim \mathcal{N}(38, 17^2)$	$Z_9 \sim \mathcal{N}(104, 19^2)$
$Z_5 \sim \mathcal{N}(127, 16^2)$	$Z_{10} \sim \mathcal{N}(41, 14^2)$
	$Z_{11} \sim \mathcal{N}(73, 11^2)$
$S_1 \sim \mathcal{N}(377, 32^2)$	$S_2 \sim \mathcal{N}(445, 38^2)$

Table 4.6.: Surgeries per session following the SMK schedule.

Session 1	**Session 2**
$Z_3 \sim \mathcal{N}(50, 6^2)$	$Z_1 \sim \mathcal{N}(41, 14^2)$
$Z_5 \sim \mathcal{N}(127, 16^2)$	$Z_2 \sim \mathcal{N}(48, 15^2)$
$Z_9 \sim \mathcal{N}(104, 19^2)$	$Z_4 \sim \mathcal{N}(38, 17^2)$
$Z_{11} \sim \mathcal{N}(73, 11^2)$	$Z_6 \sim \mathcal{N}(93, 17^2)$
	$Z_7 \sim \mathcal{N}(41, 14^2)$
	$Z_8 \sim \mathcal{N}(93, 17^2)$
	$Z_{10} \sim \mathcal{N}(41, 14^2)$
$S_1 \sim \mathcal{N}(354, 28^2)$	$S_2 \sim \mathcal{N}(395, 41^2)$

Table 4.7.: Surgeries per session after applying risk-pooling to the SMK schedule.

The KPI 'correct end times of sessions' reflect the percentage of sessions that ended in between a 15 minute range of the planned end times. We denote the planned duration of a session with D_j, $j = 1, 2$. To calculate the value of this KPI for session j, we calculate the probability that the total session duration S_j lies within 15 minutes of the planned duration D_j, as follows,

$$P(D_j - 15 < S_j < D_j + 15) = F_{S_j}(D_j + 15) - F_{S_j}(D_j - 15), \tag{4.2}$$

where F_{S_j} is the cumulative distribution function of the real valued random variable S_j, which is assumed to be normally distributed. We state the planned end time to be equal to the sum of the expected surgery time, since we assume that other processes that happen during the session, such as change-overs, are not stochastic. The method of risk-pooling improves the performance of the KPI 'correct end times of sessions' slightly, theoretically speaking, shown by Table 4.8.

Table 4.8.: The improvement of the KPI for correct end times in case risk-pooling is applied.

	D_j	Probability that the realised end time is within 15 minutes of the planned	Value of the KPI for correct end times
Session 1 ($j = 1$)	377 min	0.36	33.5%
Session 2 ($j = 2$)	445 min	0.31	
		- Apply risk-pooling -	
Session 1 ($j = 1$)	354 min	0.41	35.0%
Session 2 ($j = 2$)	395 min	0.29	

To conclude, the risk-pooling approach can be used to improve the schedule regarding the performance indicator of the correct end times of sessions. This would yield clustered surgeries with high variability and clustered surgeries with low variability, which causes some sessions to have a high expected deviation of the end times, and some with low. The SMK can decide to reserve slack time to deal with the uncertainty based on the expected variance of the session time. We only tested the risk-pooling method on a small instance in this Section. We refer to Section 4.4 where for a larger instance the method of risk-pooling is tested against the schedule of the SMK, for which ILP 4.1 is used, and the results are tested against the method of ET-scheduling, which is explained in the next Section 4.3.

4.3 Minimal Earliness and Tardiness

From literature we found that one method to decrease the total variability of a schedule is Earliness/Tardiness (ET) scheduling. In this section we will explain ET scheduling, and we will discuss the use of ET scheduling for our model.

The stochastic ET scheduling problem minimizes the total ET costs of a schedule. The ET costs are defined as the sum, over all surgeries in the schedule, of the expected deviation of the surgery completion times from the planned completion times. The goal of our model is to only minimize the expected deviation of the session completion time from the planned completion time of the session. To see how we can use ET scheduling to achieve this goal, we will dive deeper into the method of ET scheduling.

The single machine Earliness/Tardiness problem is a classical problem in machine scheduling. The deterministic version has been proven to be NP-Complete. For the single machine problem it has been proven that for general distributed processing times in order to achieve minimal earliness and tardiness costs scheduling jobs in increasing order of variance is optimal [42]. Several variants of the E/T problem are extended to multiple machine models. Otten et al. [16] consider the stochastic E/T scheduling problem on multiple machines, in which, in addition to the schedule, the due dates also need to be selected. We will continue with the approach of Otten et al.

Given a finite number of surgeries n with known distribution of their completion time and a finite number of operating rooms m the multiple machine E/T-problem is to produce a schedule with minimal total earliness and tardiness, i.e. a minimal expected deviation from the intended schedule. Let C_j^i denote the completion time, that is defined as the sum of the processing time of the surgery itself and the processing times of the successors of the surgery in the produced schedule, and let d_i^j denote the due date of surgery j on operating room i. Let $E_j^i(d_j^i) = (d_j^i - C_j^i)^+$ be the earliness of surgery j on operating room i, and $T_j^i(d_j^i) = (C_j^i) - d_j^i)^+$ be the tardiness of surgery j on operating room i. The multiple machine ET problem (MET) is given by,

$$\min_{\Pi,\, d^{\Pi}} \quad \sum_{i=1}^{m} \sum_{j=1}^{n} \mathbb{E}\left[\alpha_j^i E_j^i(d_j^i) + \beta_j^i T_j^i(d_j^i)\right], \tag{4.3}$$

where $\Pi = [\pi_1 \pi_2 \dots \pi_m]$ denotes the schedule (that is the sequence of surgeries for each operating room), where d^{Π} denotes the due dates of the jobs under schedule Π, and where α_j^i and β_j^i denote the unit earliness and tardiness cost parameters of surgery j on operating room i. The ET cost function can therefore be defined as,

$$F_j^i(d_j^i) = \mathbb{E}\left[\alpha_j^i E_j^i(d_j^i) + \beta_j^i T_j^i(d_j^i)\right]. \tag{4.4}$$

The symmetric multiple machine ET problem (SMET) is the problem where the cost parameters are equal for every surgery and operating room, that is $\alpha_j^i = \alpha$ and $\beta_j^i = \beta$, for $j = 1, \dots, n$.

Otten et al. use the E/T concavity property to generate an optimal schedule for the symmetric multiple machine variant of the E/T problem. A probability distribution is E/T-concave if the E/T cost function is concave in the standard deviation of the completion time. For a formal definition we refer [16]. The E/T-Concavity property holds for normal and exponential distributed surgery time. For log normal distribution simulations intend to show that the property holds, however this has not been proven yet.

We continue with the assumption that surgery times are independent and normally-distributed. Since the E/T-concave property holds, Otten et al. proved that the following method of scheduling is optimal in obtaining minimal earliness and tardiness costs. The optimal schedule with minimal E/T-costs is the schedule where there are no surgeries p_a and q_b, such that the number of successors of surgery p_a on OR a is higher than that of job q_b on OR b, and the variance of the processing time of job p_a is higher than the variance of the processing time of job q_b. This method is illustrated with Figure 4.5. If the amount of surgeries cannot be equally divided among the ORs, some ORs will have more surgeries than others.

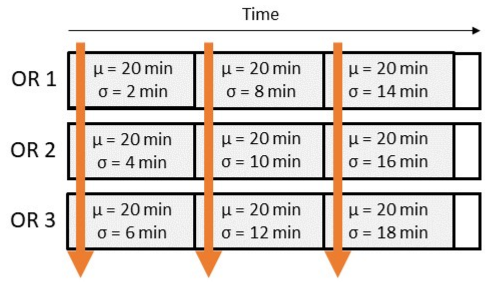

Figure 4.5.: Schedule in increasing variance to obtain an optimal ET schedule.

The total ET costs for operating room i with independent normal distributed surgery times can, in case symmetry in ET costs is assumed, be calculated by the following formula:

$$ET_i = \sum_{j=1}^{n} F_j^i(d_j^i) = \frac{1}{\sqrt{2\pi}} \sum_{j}^{n} \sigma_j^C, \tag{4.5}$$

where σ_j^C is the standard deviation of the completion time of surgery j. Since the sum of independent normal distributed variables is again normal distributed, the standard deviation of the completion time of surgery j can be calculated by taking the square root of the sum of variances of its successors and the variance of surgery j itself. The complete derivation for formula (4.5) can be found in Appendix A.3.

Minimizing the expected ET costs of a schedule is not the only goal of a hospital. The proposed method could result into having some ORs with a long total duration, and some ORs with a short total duration, whereas it could be preferred to have all the ORs ending around the same time. For the SMK this is the case, since sessions normally start at 8:00am, and end at 16:30pm. For this, secondary optimization is proposed, since changing the ET schedule slightly, results only in a minor change in ET costs. Otten et al. state that surgeries within the same ranking group can be interchanged. An example for this is given by Figure (4.6) and Figure (4.7). In Figure (4.6) surgeries are scheduled following the optimal ET schedule. The surgeries can be divided in to four ranking groups, where group 1 exists of surgery a and b, group 2 of surgery c and d, etc. Using the makespan as a secondary objective, yields into interchanging surgery e and f, such that both OR sessions are planned within a short distance from the session duration. The new schedule is illustrated in Figure (4.7).

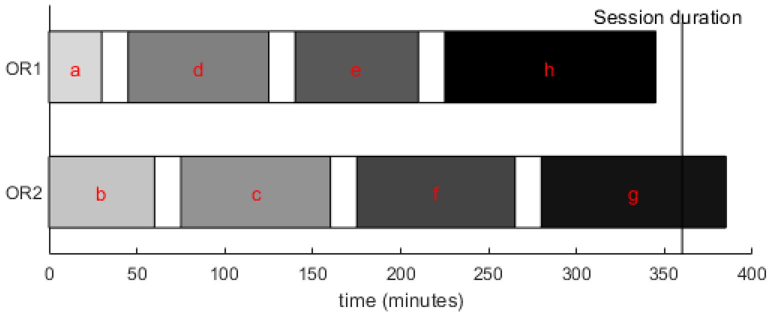

Figure 4.6.: Optimal ET-Schedule of eight surgeries over two operating rooms. Length corresponds to expected surgery time and the color with the variance (darker means higher variance).

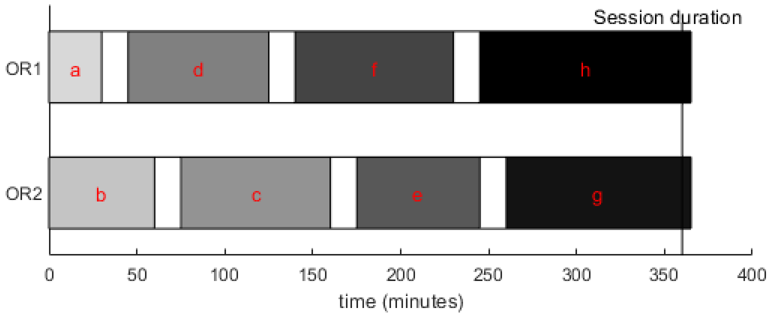

Figure 4.7.: ET-Schedule with secondary makespan optimization. Length corresponds to expected surgery time and the color with the variance (darker means higher variance).

The integer linear program for obtaining the schedule of Figure (4.7), that is a combination of ET scheduling and makespan optimization, is given by the ILP (4.6). In this formulation we make use of the sets P_l, with $l = 1, 2, \dots L$, where L is the average number of patients per session rounded above. The set P_l then exists of the surgeries with standard deviations that fall in the same ranking group l, where dummy patients are added to ranking groups that have less than h patients, where h is the round to above quotient of number patients and the number of sessions. The complete list of sets, variables, and parameters used for (4.7) is given by Table 4.9.

Table 4.9.: Sets, variables, and parameters for (4.6).

Sets	**Description**
$\mathcal{S} = \{0, 1, \dots, S\}$	Set of sessions, indexed by s, with S the number of sessions.
$\mathcal{P} = \{0, 1, \dots, P\}$	Set of patients, indexed by p, with P the number of patients.
$\mathcal{L} = \{1, \dots, L\}$	Set of ranking groups, indexed by l , with L the number of ranking groups.
$\mathcal{P}_l = \{1, \dots, h\}$	Subset of patients, existing of patients within ranking group l, with $h = \left\lceil \frac{P}{S} \right\rceil$.
Variables	**Description**
$x_{s,p}$	Binary variable, indicating whether patient p has been booked on session s or not.
C_{max}	Represents the expected total duration of the longest session.
Parameters	**Description**
μ_p	Represents the expected surgery time of patient p.

$$\begin{aligned}
\min_{x, C_{max}} \quad & C_{max}, \\
\text{s.t.} \quad & \sum_{p \in \mathcal{P}} x_{s,p}\mu_p \leq C_{max}, \quad && \forall s \in \mathcal{S}, \\
& \quad \sum_{p \in I_l} x_{s,p} = 1, && \forall s \in \mathcal{P}, l \in \mathcal{L}., \\
& \quad \sum_{s \in \mathcal{S}} x_{s,p} = 1, && \forall p \in \mathcal{P}.
\end{aligned} \tag{4.6}$$

In (4.6) the first constraint determines the session duration, and the second constraint states that all ORs should have one patient from each ranking group, and the third constraint states that each patient can be booked once.

Including the objective of minimizing ET costs directly in an integer program, where we do not have to pre-compute the set of ranking groups. is difficult. For this, the completion time of each surgery should be calculated and saved, which depends on the order of surgeries. Define $y_{s,p}$ to be the completion time of the surgery of patient p on session s. Let t also indicate a patient. Defining $y_{s,p}$ would result into a constraint, stating the following:

$$y_{s,p} = \frac{1}{\sqrt{2\pi}} \sqrt{\sum_{t: \sigma_p \geq \sigma_t} \sigma_t x_{s,t}^2}, \quad \forall s, p,$$

where σ_p is the expected standard deviation of the surgery time of patient p. We would then minimize over the sum of all sessions and patients over $y_{s,p}$ to obtain a schedule with minimal earliness and tardiness costs. This would result in a non-desired complex MINLP.

ET-Scheduling includes setting the order of the surgeries to achieve minimal total ET costs. Changing the order of surgeries, asks flexibility from the OR, and the order should be feasible; that is acceptation by the personnel, and available resources at the specified times. Experiences of the OR-planning department are that there is only little space in setting the order of surgeries. The focus of this research is not on the expected completion times of all the surgeries, but only on the expected completion time of the last surgery of the day. The expected earliness and tardiness of the end time of the last surgery does not depend on the order of surgeries on the day itself. It can however be influenced by interchanging surgeries between multiple sessions, such that each session has the least possible expected deviation of the end time. This idea comes down to a risk-spreading method.

Risk-spreading can be achieved by the following MINLP:

$$\begin{aligned}
\min_{x} \quad & \sum_{s} \left(\sum_{p} \sigma_p^2 x_{s,p}\right)^2, \\
\text{s.t.} \quad & \quad \sum_{s} x_{s,p} = 1, \quad && \forall p \in \mathcal{P}, \\
& \sum_{p} \mu_p x_{s,p}, \leq C_s && \forall s \in \mathcal{S},
\end{aligned} \tag{4.7}$$

where $x_{s,p}$ is the binary variable stating if a patient p is booked on session s or not, where μ_p is the expected surgery time, where σ_p is the expected standard deviation of the surgery time of patient p, and where C_s is the capacity of session s. Including the power of two in the objective, makes sure that it is preferred to book a surgery onto a session with low variance, instead of booking the session onto a session with high variance. The first constraint makes sure that each patient is planned. In order to fit within the capacity of the sessions, the last constraint is added. The combination of constraint 1 and 2 can cause in-feasibility, in case no combination of surgeries exists that fits within the available capacity. The relaxation of (4.7) is a convex problem, therefore the program can be solved with a convex MINLP solver. Note that (4.7) only assigns a surgery date to a patient, since the order of surgeries still has to be determined. According to the ET-scheduling, this booking should be done by the shortest variance first rule. The risk spreading approach spreads the earliness and tardiness costs of the end time of the sessions out over the multiple sessions, this does not mean that an optimal ET schedule is achieved after applying SVF to each separate sessions of the risk-spreading schedule.

The program of (4.7) can be used for a given set of surgeries and a given set of sessions. The schedule it creates is not built in compliance with the uncertain future, existing of new arrivals and cancellations, nor does it include patient lead-time restrictions. The problem that we are dealing with, has a finite set of surgeries and finite set of sessions given a moment in time, however, one time-step further these sets could have changed by for example by a new acute arrival. Assuming that the waiting list is always big enough to completely fill the session capacity, the problem arises that sessions were already fully booked, while this new acute arrival also needs treatment. For this, we have to use a decision making method that takes into account the future events. Section 4.5 explains how the method risk-spreading will be applied in this research.

4.3.1 Application of the ET-algorithm to SMK data

We first apply the single machine ET problem to a specific session of one surgeon in the year of 2020, and we compare the ET costs on a surgery level between the SMK schedule and an ET schedule. Then we will apply ET-scheduling to multiple sessions and show the improvement against the current schedule regarding the total ET costs. At last we apply the risk-spreading methods, described in the previous session, to the SMK data, to see the effect of the risk-spreading method in terms of the KPI 'correct end times of sessions'.

First we will compare the planned schedule of the SMK for a certain session with an optimal ET schedule. We will do this for a session in the beginning of 2020. If we took a session from the beginning of 2019, there would be a difference in the schedule, since the change-overtime has been set from 25 minutes to 15 minutes, after the opening of the new OR complex in November 2019. The session starts at 8:00am till 16:30pm and contains 6 different patients that were booked onto the session. The ET algorithm affects the order in which the patients are scheduled during the day. The results are shown in Figure 4.8. The figures shows the ET schedule and the planned schedule by the SMK. The length of the surgeries represent the expected surgery time and the color of the surgery represents the variance of the surgery time, the darker the higher the variance. In red patient numbers are shown, to see how the schedule has changed when SVF is applied. The time between surgeries (blank space) is the change-overtime. The expected earliness and tardiness costs per start of a surgery for the schedules of Figure 4.8 are given in the accompanying Table 4.10.

For this specific session (of Figure 4.8) the ET costs could have been decreased by applying the shortest variance first rule. The ET costs for the optimal ET schedule were in total 38.5 minutes, whereas the cost of the SMK schedule 53.4 minutes. On a surgery level, we see that during the session, there is less variance is expected of the start time of each surgery. Changing the order of surgeries assumes there are no restrictions on the order of surgeries during the day. Note that the expected ET costs of the end time of the session do not decrease in case SVF is applied to one session.

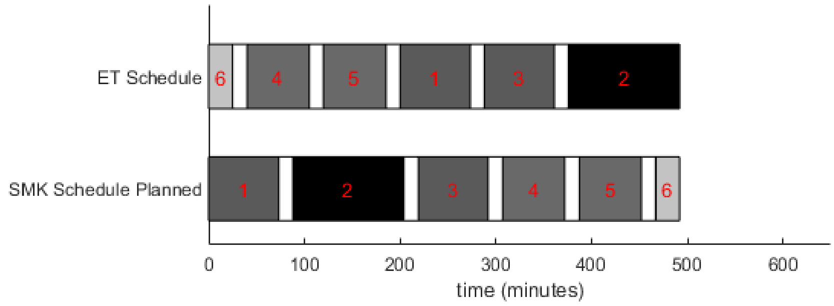

Figure 4.8.: The planned schedule against an ET schedule for a session in 2020. The color represents the standard deviation (dark means high std).

Table 4.10.: Expected ET costs per start of a surgery, belonging to the schedules of Figure (4.8)

	Start s2	Start s3	Start s4	Start s5	Start s6	End of session	Total
SMK Schedule	4.4	8.1	9.2	10.0	10.8	10.9	53.4
ET Schedule	1.6	4.3	5.9	7.3	8.5	10.9	38.5

We apply the ET algorithm to multiple sessions to show the improvements that can be achieved compared to the current schedule over a longer period. The data exists of eight sessions of one surgeon in the period January 2020. First, we look at applying SVF to each session, without interchanging surgeries between sessions. The results in terms of total ET costs are given by Table (4.11). For each session there is a possibility to decrease the earliness and tardiness during the session, which would lead to less variability in the schedule. The total ET costs can be decreased from 483 minutes to 391 minutes. The probability that the realised end times are within 15 minutes from the planned end times are calculated in the same way as was done for risk-pooling, by (4.2). These probabilities do really differ from each other when surgeries are booked onto the sessions in the way the SMK did, since the highest probability equals 0.42, and the lowest equals 0.27. This range will be smaller when ET scheduling is applied.

We look at the possibility to book the surgeries onto the sessions in such a way that minimal total ET costs are achieved. The optimal ET schedule is accomplished by applying the method illustrated by Figure 4.5. This assumes there are no lead-time constraints, such that each surgery can be booked onto each session. Results are given by Table 4.12. This schedule would decrease the ET costs even further, compared to schedule of Table 4.11. Next to this, the probability that the realised end times are within 15 minutes from planned, is roughly spread out over the sessions. The highest probability now equals 0.37 and the lowest equals 0.29. The average values stays the same.

Next, we apply secondary makespan optimization to the optimal ET schedule, given by (4.6), such that the planned duration more or less is the same for each session. This would yield the results given by Table 4.13. In this schedule we have interchanged surgeries between sessions, that fall in the same ranking group. The total ET costs for the eight sessions together would be 380 minutes, which is an improvement compared to the schedule of Table 4.11, but slightly worse than the optimal ET schedule of Table 4.12. The probabilities are spread out even further, whereas the average stays the same.

Applying the risk-spreading model (4.7) to the data gives a feasible solution within the capacity of the sessions, and yields a highest probability of ending within 15 minutes from the end time equal to 0.3249, and a smallest probability of 0.3245. Compared to the above described scheduling methods, the risk of deviating from the end time of the session is heavily spread out over the sessions.

In conclusion we see that in the application of ET scheduling to SMK data the total expected ET costs can be decreased in case changing the order of surgeries is feasible for the SMK. ET scheduling does however not affect the expected deviation of the end times of sessions. To only look at the end times of sessions, we introduced the method of risk-spreading. The method risk-spreading minimizes the earliness and tardiness costs of the end time of the sessions per session. To that end, personnel would experience less deviation from their working times, since the lowest probability of ending in time is has been brought up due to the spreading of the risk. After the risk-spreading is applied, it is still advised to book the surgeries in order of shortest variance first, to reduce the total earliness and tardiness costs on the day.

Table 4.11.: Expected ET costs per session of the SMK schedule against an SVF schedule, and the probability that the sessions end within a range of 15 minutes from planned.

Session	1	2	3	4	5	6	7	8	**Total**
Planned duration w/o change-overs (min)	304	416	521	414	373	357	364	403	3152
ET costs SMK schedule (min)	44	53	83	68	67	57	59	57	483
ET costs when SVF is applied to each session separately (min)	40	39	69	54	51	44	52	42	391
Probability that realised end time is within 15 min of the planned	0.36	0.42	0.30	0.27	0.33	0.34	0.31	0.33	0.33 (av.)

Table 4.12.: Expected ET costs per session for the optimal ET-schedule, with the same set of surgeries as of Table 4.11, and the probability that the sessions end within a range of 15 minutes from planned.

Session	1	2	3	4	5	6	7	8	**Total**
Planned duration w/o change-overs (min)	321	387	364	390	469	453	386	383	3152
ET costs for optimal ET schedule with as second objective makespan (min)	41	41	44	45	48	51	53	55	378
Probability that realised end time is within 15 min of the planned	0.37	0.36	0.33	0.33	0.32	0.31	0.30	0.29	0.33 (av.)

Table 4.13.: Expected ET costs per session for the optimal ET-schedule with as second objective makespan optimization, with the same set of surgeries as of Table 4.11, and the probability that the sessions end within a range of 15 minutes from planned.

Session	1	2	3	4	5	6	7	8	**Total**
Planned duration w/o change-overs (min)	392	388	395	396	394	395	396	396	3152
ET costs for optimal ET schedule with as second objective makespan (min)	45	47	45	43	47	47	53	53	380
Probability that realised end time is within 15 min of the planned	0.36	0.34	0.33	0.33	0.33	0.30	0.30	0.32	0.33 (av.)

4.4 Comparison Risk-pooling and ET algorithm

In this section we will calculate the overshoot probability for the scheduling by the method of risk-pooling and the method of ET optimization. These methods will be applied on data of January and February 2020. This dataset consists of 66 patients and 14 sessions.

The schedule made by the SMK for this period yields the following results, shown by Table 4.14. The booking accuracy is the amount of sessions that end with less than 15 minutes deviation from the intended end times of the schedule. The probability of over and under shoot per session was $(0.586, 0.634, 0.635, 0.638, 0.672, 0.674, 0.683, 0.691, 0.712, 0.732, 0.733, 0.759)$.

Table 4.14.

Average booking accuracy	0.320
Average probability of over and under shoot	0.680
Smallest probability of over and under shoot	0.586
Highest probability of over and under shoot	0.759

The method of ET optimization with secondary objective makespan, defined by (4.6) yields the following results on the given dataset, shown by Table 4.15. The probability of over and under shoot per session was $(0.674, 0.674, 0.681, 0.681, 0.683, 0.684, 0.687, 0.688, 0.693, 0.696, 0.702, 0.707, 0.716)$.

Table 4.15.

Average booking accuracy	0.310
Average probability of over and under shoot	0.690
Smallest probability of over and under shoot	0.674
Highest probability of over and under shoot	0.716

The method of risk-pooling, defined by (4.1), yields the following results shown by Table 4.16. This method accepts a few outliers, but minimizes the total deviation from the intended end times, therefore more sessions should deviate less from their intended end times. The probability of over and under shoot per session was $(0.433, 0.544, 0.620, 0.658, 0.661, 0.680, 0.694, 0.709, 0.720, 0.743, 0.772, 0.793)$.

Table 4.16.

Average booking accuracy	0.330
Average probability of over and under shoot	0.670
Smallest probability of over and under shoot	0.433
Highest probability of over and under shoot	0.793

The average booking accuracy of the SMK schedule is in between the minimal ET schedule and the risk-pooling schedule. The method of ET optimization with secondary objective makespan spread the variance over multiple sessions, since the smallest probability of over and under shoot and the highest probability of over and under shoot are close to each other. Risk-pooling has the highest difference between the those probabilities, which leads to a small improvement in booking accuracy. To leave the choice open on whether on average a higher booking accuracy is preferred, on the expense of having session with real high deviation and some with very low (risk-pooling), or on having a comparable value of the booking accuracy to the current situation, with less deviation of the schedule during the day (ET-scheduling), we continue with implementing both scheduling methods into a stochastic model, which is explained in the next section.

4.5 Conclusion on the methodologies

We use this section to conclude on the methodologies chapter.

From literature we see, that we can model the decision of booking a patient to an OR session, taking into account the uncertain future, with the use of Markov Decision Theory [8]. The cost function of the Markov Decision Model, will be used to incorporate the ideas of risk-pooling and ET-scheduling. First, we explain how we can model risk-pooling through MDP, then we explain how we can model the idea of ET scheduling through MDP.

Section 4.2 explained the method of risk-pooling. With the MINLP (4.1) we could determine the schedule for which the sum of the expected deviation from all of the sessions was minimized. The objective of (4.1) can be included as part of the cost function of the MDP, that is the sum over the sessions of the square root of the sum of the variances of each session, with the underlying assumption that surgery times are independent and normally-distributed. The average booking accuracy of the sessions can then be improved, which means that the personnel will experience less deviation in total, on the expense of having some sessions with very high variance.

Section 4.3 explained the method of ET scheduling. ET scheduling affects the order of surgeries during the day, whereas the expected deviation of the end times does not depend on the order of surgeries. This research focuses on the expected deviation of the end times. To that end, we do not use the ET scheduling method, since that yields pre-computation of ranking groups, which is difficult to use in a dynamic process, or the involvement of complex constraints that determine the standard deviation of the completion times of surgeries, while it would not contribute to the goal of this research. The method of risk-spreading affects the expected deviation of the end times of sessions. Compared to ET scheduling, the method of risk-spreading minimizes the expected earliness and tardiness costs of the end times of the sessions per session. It does not improve the average value of the KPI for correct end times of sessions, but it can decrease the risk per session, that is having less sessions with high risk, since the risk has been averaged over all the sessions. This could create a schedule where the probability that the personnel experiences sessions with high deviation from the end times is less compared to the current schedule or risk-pooling. The objective of (4.7) can be incorporated in the cost function of the MDP. Chapter (5) explains how the method of risk-spreading and risk-pooling are incorporated in the MDP.

5 Model

In this section we present a Markov Decision model. In Section 5.1 we give a Model Description. In Section 5.2 we formulate a Markov decision problem and, due to the curse of dimensionality, we formulate an ADP approach to obtain approximate optimal policies in Section 5.3.

5.1 Model Description

We propose a model for the advanced patient scheduling of the OR. The decision to be made, is to book a patient on a session or leave the patient on the waiting list. This decision should be made in compliance with the uncertain future. With the information of the waiting list and the information of the sessions in the future, and the probability of having events such as cancellations or arrivals, the model has to determine the optimal decision to take. For this, a Markov Decision model is well suited.

The goal of the model is to determine a strategy that takes into account the following objectives:

- Minimization of earliness and tardiness of the end times of sessions.
- Minimization of the access times of patients in an equitable manner.
- Giving as quickly as possible an indication of the day/time of surgery for the patient, that is having an acceptable invitation time.
- Minimization of the deviation from the capacity of sessions.

The optimal decision exists of whether a certain patient(group) is allocated to a specific session or kept on the waiting list (postponing the decision to book).

We list the assumptions of the model:

- *ORs are identical.*
  The OR-complex of the SMK has identical ORs, such that each surgery can be performed at each OR. In case this model is used at other healthcare facilities, this assumption should be taken into account, since no operating-room related restrictions are taken into account.
- *ORs are independent of each other.*
  This means no OR pooling is allowed. OR pooling allows to carry out surgeries in parallel as the main surgeon only needs to be present during the critical part of the surgery and can move to the next patient before closing the patient. The model is solved for one surgeon, and therefore taking into account the possibility of OR pooling cannot be done. At the SMK, OR-pooling is barely done. In most of the cases the surgeon performs at one OR during the whole session.
- *Downstream resources are available (i.e. recovery beds, PACU, etc.)*
  No restrictions are added on available resources of the downstream departments. For the SMK,

this is a reasonable assumption, since the SMK, has enough beds in the combined holding, recovery and PACU.

- *The surgery times are independent and normally-distributed.*
  We prefer the normal distribution over the log normal distribution, even though literature suggests surgery times are often log normally distributed [17]. This assumption simplifies the calculation of the standard deviation of the end times of sessions, since the sum of independent normal distributed random variables is again normal distributed. For log normal distributed random variables, this property does not hold, but it can be approximated by a log normal distribution [43]. This approximation could worsen the quality of the solution, and next to that, the approximation is impossible to do while solving the MDP.
- *The equipment is available.*
  No equipment related restrictions are added to the model. Including equipment restrictions would yield saving information about the type of surgery in the model, this would increase the complexity of the model. Next to this, the order of surgeries on the day itself is still free to decide on, which could be used to solve the problem of possible equipment restrictions.

The model creates a policy for booking patient groups of a certain urgency category. The patient groups are defined by the expected surgery time and the expected standard deviation of the surgery time. The booking agent still has to choose which patient on the waiting list, belonging to this patient group, is booked on the session, since then the surgery date can be communicated to the patient. A suitable decision rule for this is the First Come First Serve rule (FCFS), where the patient with the longest waiting time is booked first, since this rule would minimize access times of patients. The booking of patient groups also leaves some freedom to the scheduler in case a patient does not accept the proposal.

5.2 The Markov Decision Problem

We solve a stochastic optimization problem in which decision making is included by a Markov Decision Process (MDP). An MDP is defined by its epochs, states, actions, transition probabilities, and rewards.

This section presents a discounted infinite horizon Markov Decision Process (MDP) for booking patients into an uncertain session roster in such a way as to minimize a combination of patient access times and invitation times, un-used OR capacity and deviation of the planned end times. The discount factor motivates the decision maker to favor taking decisions early, rather not postponing them indefinitely. The used sets, variables, and parameters can be found in Table 5.1.

Table 5.1.: Sets and parameters used in the MDP Model

Sets	**Description**
$\mathcal{N} = \{0, 1, \ldots, N\}$	Set of days in the booking horizon, indexed by n, with N the length of the booking horizon in days.
$\mathcal{T} = \{1, \ldots, T\}$	Set of patient types, indexed by t, with T the number of different patient types.
$\mathcal{U} = \{1, \ldots, U\}$	Set of urgency categories, indexed by u, with U the number of different urgency categories.

Variables	**Description**
$x_{t,n}$	State variable, indicating the number of patients of type t that has been booked on a session on day n.
$m_{t,u}$	State variable, indicating the number of patients of type t with urgency category u urgency category u on the waiting list.
b_n	State variable, indicating if a session exists at day n or not.
$d_{t,u,n}$	Decision variable, indicating the number of patients of type t with urgency category u that is booked on day n.
Parameters	**Description**
μ_t	Represents the expected surgery time of patient type t.
σ_t	Represents the expected standard deviation of the surgery time of patient type t.
n_u	Represents the maximum days before treatment of a patient with urgency category u.
C	Represents the capacity of a session in minutes.
δ, ζ, η	Parameters representing the ratio of importance of the earliness and tardiness of end times, equitably booking with minimal access times, and overtime costs, respectively.
D	Represents the maximum number of patients that can be booked on a session.
Q	Represents the maximum number of patients that can be on the waiting list of one type of patients of one urgency category.
C^{max}	Represents the maximum fraction of the capacity of a session that can be utilized.
r_{RP}	Represents the re-scale factor in case the objective of risk-pooling is incorporated in the cost function.
r_{RS}	Represents the re-scale factor in case the objective of risk-spreading is incorporated in the cost function.

5.2.1 Epochs and horizon

We are dealing with an infinite horizon. The amount of sessions that have been planned at a tactical level is finite given a moment in time, however the horizon is rolling, and therefore we need an infinite horizon. Define N to be the length of the booking horizon in days. We have $\mathcal{N} = \{0, 1, \ldots, N\}$, the set of days in the booking horizon indexed by n. Saturdays and sundays are excluded in this set.

We assume that booking decisions are made once a day. Therefore a decision epoch is at the beginning of each day.

5.2.2 States

We define the set of type of patients to be $\mathcal{T} = \{1, \ldots, T\}$, indexed by t. The type of a patient is defined by its expected surgery time μ_t and its expected standard deviation of its surgery time σ_t. We define the set of urgency categories to be $\mathcal{U} = \{1, \ldots, U\}$, indexed by u.

The state should include information on the number of patients of type t whom are already booked on a session on day n. We need information on the type of patient in case a patient is cancelled. The state should include information on the number of patients of type t with urgency category u on the waiting list, those whom still have to be booked. Next to this information should be included on whether or not a session exists on day n.

We define the state space to be:

$$\mathbf{s} = (\mathbf{x}, \mathbf{m}, \mathbf{b}) = (x_{t,n}(\mathbf{s}), m_{t,u}(\mathbf{s}), b_n(\mathbf{s}))_{t \in \mathcal{T}, n \in \mathcal{N}, u \in \mathcal{U}},$$

where $x_{t,n}(\mathbf{s})$ defines the number of patients of type t who are booked to a session on day n. Where $m_{t,u}(\mathbf{s})$ is the number of patients of type t with urgency category u who are on the waiting list. Where $b_n(\mathbf{s})$ is the state denoting if a session exists on day n or not.

In order to use the mathematical programming model we need a finite state space. Therefore, we assume upper bounds to the number of patients that can be booked on one day, given by D, and to the number of patients on the waiting list, given by Q. These upper bounds will be chosen sufficiently high so they are of little influence on the final policy. The set of states will be denoted by,

$$\mathcal{S} = \left\{ (\mathbf{x}, \mathbf{m}, \mathbf{b}) \;\middle|\; \begin{array}{ll} \sum_{t \in \mathcal{T}} x_{t,n}(\mathbf{s}) \leq D \cdot b_n(\mathbf{s}), & \forall n \in \mathcal{N}; \\ x_{t,N}(\mathbf{s}) = 0 & \forall t \in \mathcal{T}; \\ m_{t,u}(\mathbf{s}) \leq Q, & \forall t \in \mathcal{T}, u \in \mathcal{U}; \\ b_n(\mathbf{s}) \in \{0, 1\}, & \forall n \in \mathcal{N}; \\ (\mathbf{x}, \mathbf{m}, \mathbf{b}) \in \mathbb{N}_0^{TN} \times \mathbb{N}^T \times \{0,1\}^N & \end{array} \right\}. \tag{5.1}$$

It is important to note that the booking horizon is not static but rolling and thus at the beginning of each decision epoch $x_{t,N} = 0$ and b_N is either zero or one, defined by the probability that a session has been assigned to the surgeon or not.

5.2.3 Actions

At each decision epoch, the task is to decide the number of patients of a specific patient type with a specific urgency category that is going to be booked on a day. Therefore, the vector of possible decisions can be defined as

$$\mathbf{d} = (d_{t,u,n}(\mathbf{s}))_{t\in\mathcal{T},u\in\mathcal{U},n\in\mathcal{N}},$$

where $d_{t,u,n}(\mathbf{s})$ is the number of patients of patient type t with urgency category u that is booked on a session of day n. Once patients groups are booked on sessions and placed in increasing order of variance, the appointment times are known.

It is important to note that we are not assigning patients to specific days, we are just allocating capacity to each type of patient with a certain urgency category. Once capacity is allocated, a second level of scheduling is needed which assigns patients to specific days. The cancellation of surgeries is assumed not to be an option as the SMK wishes to determine a policy to avoid such measures. Therefore we seek to determine a policy that takes future events into account, such that cancellations do not have to occur, and that only in the most unlikely convergence of circumstances a cancellation has to be done manually.

For the action to be valid, any action in **s** must satisfy the following constraints.

- A patient of type t with urgency category u can only be booked when he/she is in the waiting list, and a patient can only be booked on one day:

$$0 \leq \sum_{n\in\mathcal{N}} d_{t,u,n}(\mathbf{s}) \leq m_{t,u}(\mathbf{s}), \qquad \forall t\in\mathcal{T}, u\in\mathcal{U}.$$

- The bookings agent cannot book patients on a day with no session:

$$\sum_{t\in\mathcal{T},u\in\mathcal{U}} d_{t,u,n}(\mathbf{s}) \leq b_n(\mathbf{s})\cdot D, \qquad \forall n\in\mathcal{N}.$$

- The bookings agent can only book patients on a day if there are enough spots left:

$$\sum_{t\in\mathcal{T},u\in\mathcal{U}} d_{t,u,n}(\mathbf{s}) \leq D - \sum_{t\in\mathcal{T}} x_{t,n}(\mathbf{s}), \qquad \forall n\in\mathcal{N}.$$

- It is not allowed to use over C^{max} of the capacity:

$$\sum_t \mu_t(x_{t,n}(\mathbf{s}) + \sum_u d_{t,u,n}(\mathbf{s})) \leq C^{max}\cdot C, \qquad \forall n\in\mathcal{N}.$$

The last constraints restricts on the use of capacity. The direct cost function will increase costs if capacity is exceeded. Therefore booking in overtime is non-desirable. Also, note that no constraint is formulated on booking a patient after their maximum days of treatment, this is also incorporated in the direct cost function.

We define the set of feasible actions, depending on the state $\mathbf{s}\in\mathcal{S}$ as,

$$\mathcal{D}(\mathbf{s}) = \left\{ (\mathbf{d}) \;\middle|\; \begin{array}{ll} \sum_{n\in\mathcal{N}} d_{t,u,n}(\mathbf{s}) \leq m_{t,u}(\mathbf{s}), & \forall t\in\mathcal{T}, u\in\mathcal{U} \\ \sum_{t\in\mathcal{T},u\in\mathcal{U}} d_{t,u,n}(\mathbf{s}) \leq b_n(\mathbf{s}), & \forall n\in\mathcal{N}, \\ \sum_{u\in\mathcal{U},t\in\mathcal{T}} d_{t,u,n}(\mathbf{s}) \leq D - \sum_{t\in\mathcal{T}} x_{t,n}(\mathbf{s}), & \forall n\in\mathcal{N}, \\ \sum_t \mu_t(x_{t,n}(\mathbf{s}) + \sum_u d_{t,u,n}(\mathbf{s})) \leq C^{max}C & \forall n\in\mathcal{N}, \\ d_{t,u,n}(\mathbf{s}) \in \{0,1,\ldots D\}, & \forall t\in\mathcal{T}, u\in\mathcal{U}, n\in\mathcal{N}. \end{array} \right\} \tag{5.2}$$

5.2.4 Transition probabilities

Once a decision is made in state **s**, the system moves to some new state **s'**. This transition deals with stochastic elements. We first introduce four random variables in Table 5.2 and elaborate on them. Some of the stochastic variables are based on the state. Afterwards we define the transition probabilities based on these stochastic events.

Table 5.2.: List of random variables used in the MDP.

Random Variable	Description
$A_{t,u} \sim \text{Pois}(\lambda_{t,u})$	The number of new arrivals of patient type t with urgency category u.
$C_{t,n}(\mathbf{s}) \sim \text{Bin}(i, p^{cancel})$	The number of cancelled patients due to non-planning related issues.
$UC_{t,u}(\mathbf{s}) \sim \text{Bin}(i, p^{uc}_u)$	The number of patients of type t and urgency category u that get an higher priority.
$S \sim \text{Bernoulli}(p^{session})$	A binary number to define whether or not day $N+1$ has a session.

The first stochastic element has to do with the uncertain arrival process. The random variable $A_{t,u}$ represents the number of arrivals of patients of type t with urgency category u at a new decision epoch. The random variable is Poisson distributed with rate $\lambda_{t,u}$, where the rate is the number of patients of type t with urgency category u that arrive at a new decision epoch on average. Literature suggests surgical requests can best be modelled as a Poisson proces [44].

The second stochastic element includes the number of cancelled patients of type t on day n due to non-planning issues, given by the random variable $C_{t,n}(\mathbf{s})$, depending on the state. The patients were already booked, but were cancelled due to several possible reasons. These could be patient-related reasons, such as: the patient wants to reschedule, the patient has no more complaints, the patient is ill, the patient did not show up or the patients screening is not completed. Reasons could also be SMK-related, such as: administrative mistakes, the surgeon or the surgical team is not available or equipment is not available. Note that the random variable $C_{t,n}(\mathbf{s})$ does not include cancellations due the priority of an emergency arrival. The random variable $C_{t,n}$ has a binomial distribution with i as the number of trials and p^{cancel} the probability that an already booked patient is cancelled because of the above summed up reasons. The binomial distributed is well suited to model cancellations [44].

The third stochastic element is the event of having a new session planned at the last day of the new booking horizon, for this we define the binary random variable S. This random variable is Bernoulli distributed with probability $p^{session}$. We assume that the probability of having a session planned does not depend on the time (season) of the year, nor on the day of the week.

The last stochastic element has to do with the possibility that the urgency category of a patient is updated. We define the random variable $UC_{t,u}(\mathbf{s})$ to be the number of patients of type t that leave the urgency category u and enter the urgency category $u-1$, due to an urgency update, which depends on the number of patients in the waiting list. The random variable $UC_{t,u}(\mathbf{s})$ has a binomial distribution with i is the number of trials and p^{uc}_u the probability that a patient of urgency category u gets a higher priority. Note that we assume that patients cannot be labelled a lower priority. Moreover, we assume that the probability of a priority change does not depend on the patient type.

The described random variables are independent of each other. Note that the random variable of the cancellations depends on the values of state **x**. The more patients are booked, the higher the probability of a cancellation. Next to this the random variable for urgency updates, depends on the number of patients on the waiting list, that is state **m**. The more patients are on the waiting list, the higher the probability of an urgency change.

As a result of choosing booking action **d** in state **s**, and having $A_{t,u}(\mathbf{s})$ new arrivals of type t with urgency category u, having $C_{t,n}(\mathbf{s})$ cancellations of type t on day n, having $UC_{t,u}(\mathbf{s})$ patients of type t, u leaving category u to $u-1$, and S whether or not a new session will exists at the end of the new booking horizon, the state of the system next day, denoted by **s**, will be determined by the following probability distribution:

$$p(\mathbf{s}'|\mathbf{s},\mathbf{d}) = \begin{cases} \prod_{t,u,n} \text{Pr}^{\text{arrival}}(A_{t,u}) \cdot \text{Pr}^{\text{cancel}}(C_{t,n}(\mathbf{s})) \cdot \text{Pr}^{\text{uc}}(UC_{t,u}(\mathbf{s})) \cdot \text{Pr}^{\text{session}}(S) \\ \qquad\qquad\qquad \text{if } \mathbf{s}' \text{ satisfies Eqs. (5.4)-(5.10)}, \\ 0 \qquad\qquad\qquad \text{otherwise}, \end{cases} \tag{5.3}$$

$$x_{t,n}(\mathbf{s}') = x_{t,n+1}(\mathbf{s}) + \sum_{u\in\mathcal{U}} d_{t,u,n+1}(\mathbf{s}) - C_{t,n+1}(\mathbf{s}), \qquad n < N, \tag{5.4}$$

$$x_{t,N}(\mathbf{s}') = 0. \tag{5.5}$$

$$m_{t,u}(\mathbf{s}') = m_{t,u}(\mathbf{s}) - \sum_{n\in\mathcal{N}} (d_{t,u,n}(\mathbf{s})) + A_{t,u} + UC_{t,u+1}(\mathbf{s}) + \sum_{n\in[0,\ldots,n_u]} C_{t,n}(\mathbf{s}), \qquad u = 1, \tag{5.6}$$

$$m_{t,u}(\mathbf{s}') = m_{t,u}(\mathbf{s}) - \sum_{n\in\mathcal{N}} (d_{t,u,n}(\mathbf{s})) + A_{t,u} + UC_{t,u+1}(\mathbf{s}) - UC_{t,u}(\mathbf{s}) + \sum_{n\in\langle(n_{u-1}),\ldots,n_u]} C_{t,n}(\mathbf{s}), \qquad 1 < u < U, \tag{5.7}$$

$$m_{t,u}(\mathbf{s}') = m_{t,u}(\mathbf{s}) - \sum_{n\in\mathcal{N}} (d_{t,u,n}(\mathbf{s})) + A_{t,u} - UC_{t,u}(\mathbf{s}) + \sum_{n\in\langle(n_{u-1}),\ldots,n_u]} C_{t,n}(\mathbf{s}), \qquad u = U, \tag{5.8}$$

$$b_n(\mathbf{s}') = b_{n+1}(\mathbf{s}), \qquad n < N, \tag{5.9}$$

$$b_n(\mathbf{s}') = S, \qquad n = N. \tag{5.10}$$

The Pr's in Equation (5.3) correspond to the respective probabilities. Equation (5.4) defines the new number of booked patients of type t on day n by adding the newly booked patients and subtracting the number of cancelled patients. Due to the rolling horizon we set the number of booked patients on the new day equal to zero with Equation (5.5). Equation (5.6),(5.7), and (5.8) update the waiting list by subtracting the newly booked patients and by subtracting the patients that enter a higher urgency category, and by adding the new patient arrivals, by adding the patients who come from an lower urgency category, and by adding the patients that were already booked but were cancelled. The difference between equation (5.6),(5.7), and (5.8) is the urgency category for which the equation holds. Firstly, the waiting list of patients of urgency category U cannot gain patients from above (5.7, and the waiting list for $u = 1$ cannot lose patients to a higher urgency category. Secondly, In case the urgency category equals one (5.7), only patients who were booked between the current day and the maximum days of treatment of type u, given by n_u, and were eventually cancelled, are added to the waiting list. For the urgency categories where $u > 1$ ((5.6) and (5.7)), patients are added to the waiting list when they where booked between the days $n_u - 1$ and n_u and their treatment was cancelled. Equation (5.8) updates state variable **b** by one day. For the last day of the booking horizon it should be determined whether or not a session will be planned or not, this is given by Equation (5.9).

The MDP does not include the probability that patients do not accept the proposal. If the they do not accept, the SMK can change the proposal by hand, for example by choosing the second best (sub-optimal) decision found by the MDP, or by interchanging the patient with a similar patient (i.e. similar surgery time and standard deviation, feasible within their maximum days of treatment). Including several decisions and the probability that a patient would accept these, would increase the size of the action space and therefore the size of the MDP. It could be possible to solve the MDP again, after the patient declined, but now with this action as non-feasible. Solving the MDP would take a lot of time, thus solving again is non-desirable.

5.2.5 Direct costs

Costs should take the following into account.

- Minimal access times of patients in a equitable manner (a cost associated with booking a patient beyond their maximum access times), where date/time of surgery is announced to patients as soon as possible, i.e. booking a patient is done as soon as possible (a cost associated with delaying the decision to book a patient).
- Minimal deviation from the capacity of sessions (a cost associated with booking less or more time than the the total time of a session).
- Minimal expected deviation from the end times of the sessions (a cost associated with the expected deviation of the end times).

We define the following cost function that includes the above mentioned objectives,

$$c(\mathbf{s},\mathbf{d}) = \zeta \sum_{u\in\mathcal{U}} A_u + \eta \sum_{n\in\mathcal{N}} O_n + \delta \sum_{n\in\mathcal{N}} D_n, \tag{5.11}$$

where $(\zeta{:}\eta{:}\delta)$ are parameters that represent the ratio of importance for the different objectives, where A_u represents the costs function for booking a patient after its maximum days of treatment, and the costs for delaying the decision to book, depending on the urgency category. Where O_n represents the costs for the difference in total booking time of a session and its capacity, depending on the day. Where D_n represents the expected standard deviation costs of a session on day n. These cost functions will be specified below.

We define the costs A_u, representing costs for booking equitable, in the following way, in a similar fashion as [8],

$$A_u = \sum_{n\in\mathcal{N}} f_{u,n}^{LATE} \sum_{t\in\mathcal{T}} d_{t,u,n}(\mathbf{s}) + f_u^{DELAY}\Big(\sum_t m_{t,u}(\mathbf{s}) - \sum_{n\in\mathcal{N}} d_{t,u,n}(\mathbf{s})\Big), \tag{5.12}$$

where,

$$f_{u,n}^{LATE} = \begin{cases} \sum_{k=1}^{n-n_u} \lambda^{k-1} f_u^{DELAY}, & \text{if } n > n_u, \\ 0 & \text{otherwise.} \end{cases} \tag{5.13}$$

In this cost function costs are incorporated for delaying the booking decision, where for each patient that is not booked, while on the waiting list, a cost of f_u^{DELAY} is incorporated. Intuitively the costs for delaying a patient with high priority should be higher than for a patient with low priority. Costs are incorporated for booking a patient after its maximum days of treatment with the aid of $f_{u,n}^{LATE}$, in this function n_u is the maximum days before treatment for a patient of urgency category u. The $f_{u,n}^{LATE}$ function depends on f_u^{DELAY}, since this insures that delaying the booking for n days and then booking within the target results in the same cost as booking a patient n days past the target initially, where λ can be chosen smaller than one if the delaying costs should be higher than the costs for booking late.

We define the costs of not using full capacity or using too much capacity, given by O_n, in the following way,

$$O_n = \begin{cases} |(\sum_t \mu_t(x_{t,n}(\mathbf{s}) + \sum_{u\in\mathcal{U}} d_{t,u,n}(\mathbf{s}))) - b_n(\mathbf{s})C| & \text{if } n \in \{0,1\} \\ \max\{\sum_t \mu_t(x_{t,n}(\mathbf{s}) + \sum_u d_{t,u,n}(\mathbf{s})) - C, 0\} & \text{else.} \end{cases} \tag{5.14}$$

This function includes costs for the upcoming two days if the session's capacity is not completely used or over-used and there are costs included for booking more capacity than available on sessions in the future. Notice that these costs are nonlinear, since the absolute value and the maximum function is incorporated. Fortunately, we can convert the absolute value and the maximum function into a linear program through LP-modelling tricks [31]. The parameter η is multiplied with O_n in the objective function, if it is chosen larger than 1, costs for crossing capacity will increase faster. We do not use a non-decreasing convex function for exceeding capacity, since this would increase model complexity, and the proposed cost function seems to behave as wished. Note that session's utilization can never be higher than C^{max} due to the constraint on the action.

The costs function D_n depends on the methodology we use, differing between the risk-pooling or risk-spreading approach.

(1) Risk pooling.
For normal distributed independent surgery times the addition in standard deviation costs of the end time of booking one or more patients extra on day n can be calculated as follows,

$$D_n = r_{RP}\left[\frac{1}{\sqrt{2\pi}}\sqrt{\sum_t \sigma_t^2\left[x_{t,n}(\mathbf{s}) + \sum_u d_{t,u,n}(\mathbf{s})\right]} - \frac{1}{\sqrt{2\pi}}\sqrt{\sum_t \sigma_t^2 x_{t,n}(\mathbf{s})}\right], \tag{5.15}$$

where r_{RP} is a scale factor, to have the let D_n behave within the same range as O_n and A_n. Note that the cost for the uncertainty of the end time of the session is only calculated for the current decision epoch and previous costs will not be counted again.

(2) Risk-spreading.
For normal distributed independent surgery times the risk-spreading effect is obtained by using the following cost function:

$$D_n = r_{RS}\left[\frac{1}{\sqrt{2\pi}}\left(\sum_{t\in\mathcal{T}} \sigma_t^2\left[x_{t,n}(\mathbf{s}) + \sum_{u\in\mathcal{U}} d_{t,u,n}(\mathbf{s})\right]\right)^2 - \frac{1}{\sqrt{2\pi}}\left(\sum_{t\in\mathcal{T}} \sigma_t^2 x_{t,n}(\mathbf{s})\right)^2\right], \tag{5.16}$$

where r_RS is the re-scale factor to have the D_n costs in the same range as the O_n and A_u costs. Note that the costs are only counted for the current decision epoch and previous costs will not be counted again. The power of 2 makes sure that booking a patient to a session with already high variance, leads to much more costs than booking a patient to a session with low variance.

For the three cost functions it should hold that they behave within the same range, such that the ratio's (δ, ζ, η) can be changed if differentiation in importance of objectives is desired. The cost function for risk-pooling creates that booking a patient is non-desirable, since that would induce a cost of σ_t, in case of an empty session. It would be less in case the session was partly filled, due to the square root. In order to still book patients, we multiply D_n with r_{RP}, which we set to the equal to one over maximal standard deviation, i.e. $r_{RP} = \frac{1}{\max_{t\in\mathcal{T}}\{\sigma_t\}}$. The cost function for risk-spreading creates a cost for booking a patient as well, next to this the cost function is not in the same range as O_n. To overcome this, we multiply D_n of (5.16) by r_{RS}, which we set equal to one over the maximal addition in costs that booking an extra patient could give in D_n, that is $r_{RS} = \frac{1}{(\frac{1}{\sqrt{2\pi}}\max_{t\in\mathcal{T}}\{\sigma_t^2\}\cdot D)^2 - (\frac{1}{\sqrt{2\pi}}\max_{t\in\mathcal{T}}\{\sigma_t^2\}\cdot(D-1))^2}$.

5.2.6 ■ Size of the MDP

The MDP faces the curses of dimensionality. In this Section we show we are dealing with a large state space and a large decision space.

The state has $(T \times N + T \times U + N)$ variables. The state variable $x_{t,n}$ can take integer values up to D including zero, but is set to zero when $b_n = 0$, since no patients can be booked on a day with no session. The state $m_{t,u}$ can take integer values up to Q excluding zero, and b_n is a binary state variable, thus can take the values 0 or 1.

The number of states is upper bounded by:

$$|S| = (1 + D)^{NT} \times (Q)^{TU} \times (2)^N. \tag{5.17}$$

The above equation is an upper bound for the number of states, since the actual number of states decreases in size if not all days have a session, i.e. $b_n = 0$ for some $n \in \mathcal{N}$. For these n, $x_{t,n}$ can only be equal to zero, instead of the integers from zero up to D. The decision variable has $(T \times U \times N)$ dimensions. The decision variable can take integer values in the range of $[0, \dots, \min\{D, Q\}]$.

The total number of feasible decisions per state is upper bounded by:

$$|D(\mathbf{s})| = (1 + \min\{D, Q\})^{TU(N+1)}.$$

The above equation is an upper bound for the number of decisions, since the decision should satisfy multiple constraints, given in Section 5.2.3. The number of feasible decisions depends on the state, that is the current number of patients on the waiting list and the current number of sessions in the future.

The number of feasible actions and feasible states can be decreased by decreasing the maximum number of surgeries per session. This maximum differs per type of surgery and per surgeon. The SMK data shows that the maximum number of back surgeries per session equals four, whereas more than four knee surgeries can be performed during one session. Some surgeons are known to perform more surgeries than others; the fastest surgeons perform maximal seven surgeries per session.

5.2.7 ■ Optimality equations

Our objective is to find decisions that minimize total discounted expected costs over an infinite horizon. For this, the optimilaty equations are as follows,

$$v(\mathbf{s}) = v(\mathbf{x}, \mathbf{m}, \mathbf{b}) = \min_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \left\{ c(\mathbf{s}, \mathbf{d}) + \gamma \sum_{\mathbf{s}'\in\mathcal{S}} p(\mathbf{s}'|\mathbf{s}, \mathbf{d}) v(\mathbf{s}') \right\}, \qquad \forall \mathbf{s} \in \mathcal{S}. \tag{5.18}$$

with discount factor $\gamma \in [0, 1)$, and $v(\mathbf{s})$ representing the expected discounted costs of state $\mathbf{s}$, $v(\mathbf{s}) \in \mathbb{R}^{|S|}$. From these optimal values the optimal decision per state can be found.

It can be proven that for $\gamma \in [0, 1)$, for a finite state space $\mathcal{S}$, and a bounded cost function $c(\mathbf{s}, \mathbf{d})$, forall $\mathbf{s} \in \mathcal{S}$ and $d \in \mathcal{D}(\mathbf{s})$, it holds that there exists a unique solution $\mathbf{v}^* = \{v^*(\mathbf{s})\}_{\mathbf{s}\in\mathcal{S}}$ to the above equation (5.18) [45, p.151, Theorem 6.2.5].

From (5.1) it follows that the state space $\mathcal{S}$ is finite. From (5.2) it follows that the action set ($\mathcal{D}(\mathbf{s})$) is finite for each $\mathbf{s} \in \mathcal{S}$. The direct costs $c(\mathbf{s}, \mathbf{d})$, defined with (5.11, are bounded, since we can find some $W \in \mathbb{R}$ such that $|c(\mathbf{s}, \mathbf{d})| \leq W \leq \infty$, forall $\mathbf{s} \in \mathcal{S}$ and $\mathbf{d} \in \mathcal{D}(\mathbf{s})$.

It can be proven that an optimal deterministic stationary policy exists when the state space $\mathcal{S}$ is discrete and the action space $\mathcal{D}(\mathbf{s})$ is finite for each $\mathbf{s} \in \mathcal{S}$ [45, p.154, Theorem 6.2.10]. The optimal decision rule is then gained by determining the optimal decision that was made at each state in the Bellman equation, making use of the following formula.

$$d(\mathbf{s}) = d^*_{\mathbf{s}} \in \arg \min_{\mathbf{d} \in \mathcal{D}(s)} \left\{ c(\mathbf{s}, \mathbf{d}) + \gamma \sum_{\mathbf{s}' \in \mathcal{S}} p(\mathbf{s}' | \mathbf{s}, \mathbf{d}) v(\mathbf{s}') \right\}$$

There are different ways to solve the MDP. Well known exact-solution methods are value iteration, policy iteration, and the linear programming formulation [45]. However, if the size of the MDP grows, the curses of dimensionality could arise and exact methods become intractable. Solution methods as approximate dynamic programming are needed to solve the problem for larger instances.

5.3 Solution Approach

The large size of state space combined with the large action space make it difficult to find a direct solution to the optimality equations (5.18). To overcome the curses of dimensionality, we make use of the field of approximate dynamic programming (ADP). We will use a linear programming approach to the ADP, that is solved through column generation. Section 5.3.1 explains the approximate linear program (ALP) for solving the MDP. Section 5.3.2 explains the column generation algorithm to solve the ALP. To run the column generation algorithm an initial feasible solution is needed. Section 5.3.3 explains the method for finding this initial solution. Section 5.3.4 explains how the approximate optimal policy is gained from the approximate value function. Section 5.3.5 states the software and the solvers that are used for the several programs.

5.3.1 Approximate Dynamic Programming

With the aid of ADP an approximate optimal strategy for achieving our goal, as formulated in Section 5.1, can be computed. This means that for each possible state that the system can be in, the approximate optimal decision to take is given by the solution of the ADP. If we assume static transition probabilities, the ADP only has to be solved one time, where after for each possible state the best choice is known. The ADP has to be solved again, when important components (i.e. parameters) of the model differ, due to for example seasonal components as holidays.

We start the explanation of our solution approach by writing the exact LP for solving the MDP. Each MDP induces an LP [45], the exact LP for solving the optimality equations is as follows,

$$\begin{aligned} \max_{v} \quad & \sum_{\mathbf{s} \in S} \omega(\mathbf{s}) v(\mathbf{s}) \\ \text{s.t.} \quad & c(\mathbf{s}, \mathbf{d}) + \gamma \sum_{\mathbf{s}' \in S} p(\mathbf{s}' | \mathbf{s}, \mathbf{d}) v(\mathbf{s}') \geq v(\mathbf{s}), \quad \forall \mathbf{d} \in D(\mathbf{s}), \forall \mathbf{s} \in S \end{aligned} \tag{5.19}$$

where $\sum_{\mathbf{s} \in S} \omega(\mathbf{s}) = 1$, which represents the weight of state $\mathbf{s} \in S$ in the objective function. Solving (5.19) yields the same solution as the optimality equations whenever $\omega(\mathbf{s}) > 0, \mathbf{s} \in S$. Next to this, the choice of $\omega(\mathbf{s})$ does not influence the solution, as long as it is strictly positive [45]. This LP has one variable for each state $\mathbf{s} \in S$ and one constraint for each state-action pair, i.e. $(\mathbf{s}, \mathbf{d})$, where

$\mathbf{s} \in S$ and $\mathbf{d} \in D(\mathbf{s})$, therefore the LP (5.19) gets an intractable number of variables and constraints.

The dual of (5.19) is given by,

$$\begin{aligned} \min_{x} \quad & \sum_{\mathbf{s}\in S}\sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s}} c(\mathbf{s},\mathbf{d})x(\mathbf{s},\mathbf{d}) \\ \text{s.t.} \quad & \sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} x(\mathbf{s},\mathbf{d}) - \gamma\sum_{\mathbf{s'}\in S}\sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} p(\mathbf{s}'|\mathbf{s},\mathbf{d})x(\mathbf{s},\mathbf{d}) = \omega(\mathbf{s}), \quad \forall \mathbf{s}\in S, \\ & x(\mathbf{s},\mathbf{d}) \geq 0, \qquad \forall \mathbf{s}\in\mathcal{S}, \forall \mathbf{d}\in\mathcal{D}(\mathbf{s}). \end{aligned} \tag{5.20}$$

Strong duality tells us that if a linear programming problem has an optimal solution, then so has its dual, and the respective optimal costs are equal [46, p.148, Theorem 4.4]. In order to show that a linear programming problem has an optimal solution, we have to show that the LP is feasible and bounded. In Section 5.2.7 it was shown that a feasible solution to the optimality equations (5.18) exists, and since (5.19 is the equivalent form of the optimality equations a feasible and bounded solution to 5.19 exists. Note that the dual has one constraint for each state **s** and one variable for each state-action pair. The problem however remains intractable.

To overcome the curses of dimensionality we use an approximate linear programming approach (ALP). The use of ADP in linear programming was first introduced by De Farias and Van Roy [27]. The large state space, resulting into too many variables, is then solved by introducing an approximate value function, and the large action space, together with the state space, results into too many constraints, can be solved through column generation [30]. We prefer this approach above the approximate dynamic programming approach, which are inefficient due to the iterative computation.

The ALP approach makes use of an approximated value function. We approximate the value function $v(\mathbf{s})$ with the basis functions ϕ_f and weights $\theta_f \in \mathbb{R}$, such that $v(\mathbf{s}) \in \text{span}\{\phi_0, \dots, \phi_F\}$:

$$v(\mathbf{s}) \approx \sum_{f\in\mathcal{F}} \theta_f \phi_f(\mathbf{s}). \tag{5.21}$$

The goal is to find a good setting of weights that approximates the optimal value function, given a set of basis functions. By approximating the value function, a reduction can take place in the dimensionality of the problem.

Various basis functions $\{\phi_f\}$ are available (e.g. polynomials, splines etc.). When choosing we should consider the approximation accuracy of the linear combinations of the basis functions and we should consider the effort involved in computing with such approximations [28]. To select a proper set of basis functions that have significant impact on the value function, we could use a regression analysis. However, this analysis would only indicate the quality of certain basis functions for a small instance, which does not indicate the quality of the approximation for larger instances. Moreover, obtaining a direct solution for a small instance that incorporates the structure of the MDP is impossible. Unfortunately, it is difficult in practice to select a set of basis functions that contains the optimal cost-to-go function within its span. Instead, basis functions must be based on heuristics and simplified analysis. One can only hope that the span comes close to the desired cost-to-go function [27]. Regression analysis done by Bikker et al. [47] would indicate an affine approximation to $v(\mathbf{s})$ as (5.22). A similar value function approximation, with an affine structure, was done by Sauré et al. [23]. Based on their observations, we define:

$$\mathcal{F} = \{1, (t,n), (t,u), n | t = 1, \dots, T, n = 0, \dots, N, u = 1, \dots U\},$$

then we have $(1+T\times N + T\times U + N)$ basis functions, and the value function is approximated by,

$$v(\mathbf{s}) = Z_0 + \sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}} X_{t,n}x_{t,n}(\mathbf{s}) + \sum_{t\in\mathcal{T}}\sum_{u\in\mathcal{U}} M_{t,u}m_{t,u}(\mathbf{s}) + \sum_{n\in\mathcal{N}} B_n b_n(\mathbf{s}), \quad \forall \mathbf{s}\in\mathcal{S}, \tag{5.22}$$

where $Z_0 \in \mathbb{R}$ and $X_{t,n}, M_{t,u}, B_n \geq 0 \; \forall n \in \mathcal{N}, t \in \mathcal{T}$, where $x_{t,n}, m_{t,u}, b_n$ follow from the state description and Z_0 is a constant. The goal is to find a good setting of the coefficients $Z_0, X_{t,n}, M_{t,u},$, and B_n that approximates the value function.

The advantage of this approximation is that the parameters are easily interpreted. $X_{t,n}$ represents the marginal infinite-horizon discounted cost of booking a patient of type t on day n, $M_{t,u}$ represents the marginal infinite-horizon discounted costs of having one more patient of type t with urgency category u on the waiting list and B_n represents the marginal infinite-horizon discounted costs of having a session on day n.

We substitute the approximate value function (5.22) into the linear program (5.19):

$$\begin{aligned}
\max_{Z_0, X, M, B} \quad & Z_0 + \sum_{n \in \mathcal{N}} \sum_{t \in \mathcal{T}} \left[\sum_{\mathbf{s} in \mathcal{S}} \omega(\mathbf{s}) x_{t,n}(\mathbf{s}) \right] X_{t,n} + \sum_{t \in \mathcal{T}} \sum_{u \in \mathcal{U}} \left[\sum_{\mathbf{s} \in \mathcal{S}} \omega(\mathbf{s}) m_{t,u}(\mathbf{s}) \right] M_{t,u} \\
& + \sum_{n \in \mathcal{N}} \left[\sum_{\mathbf{s} \in \mathcal{S}} \omega(\mathbf{s}) b_n(\mathbf{s}) \right] B_n, \\
\text{s.t.} \quad & Z_0 + \sum_{t \in \mathcal{T}, n \in \mathcal{N}} X_{t,n} x_{t,n}(\mathbf{s}) + \sum_{t \in \mathcal{T}, u \in \mathcal{U}} M_{t,u} m_{t,u}(\mathbf{s}) + \sum_{n \in \mathcal{N}} B_n b_n(\mathbf{s}) \\
& - \gamma \sum_{\mathbf{s}' \in \mathcal{S}} p(\mathbf{s}'|\mathbf{s}, \mathbf{d}) (Z_0 + \sum_{t \in \mathcal{T}, n \in \mathcal{N}} X_{t,n} x_{t,n}(\mathbf{s}') + \sum_{t \in \mathcal{T}, u \in \mathcal{U}} M_{t,u} m_{t,u}(\mathbf{s}') \\
& + \sum_{n \in \mathcal{N}} B_n b_n(\mathbf{s}')) \qquad \leq c(\mathbf{s}, \mathbf{d}), \quad \forall \mathbf{s} \in \mathcal{S}, \mathbf{d} \in \mathcal{D}(\mathbf{s}),
\end{aligned} \tag{5.23}$$

rearranging terms, to see a clear structure of each weight coefficient with its feature, yields,

$$\begin{aligned}
\max_{Z_0, X, M, B} \quad & Z_0 + \sum_{n \in \mathcal{N}} \sum_{t \in \mathcal{T}} \left[\sum_{\mathbf{s} \in \mathcal{S}} \omega(\mathbf{s}) x_{t,n}(\mathbf{s}) \right] X_{t,n} + \sum_{t \in \mathcal{T}} \sum_{u \in \mathcal{U}} \left[\sum_{\mathbf{s} \in \mathcal{S}} \omega(\mathbf{s}) m_{t,u}(\mathbf{s}) \right] M_{t,u} \\
& + \sum_{n \in \mathcal{N}} \left[\sum_{\mathbf{s} \in \mathcal{S}} \omega(\mathbf{s}) b_n(\mathbf{s}) \right] B_n, \\
\text{s.t.} \quad & (1-\gamma) Z_0 + \sum_{t,n} X_{t,n} \left(x_{tn}(\mathbf{s}) - \gamma \sum_{\mathbf{s}' \in \mathcal{S}} p(\mathbf{s}'|\mathbf{s}, \mathbf{d}) x_{t,n}(\mathbf{s}') \right) \\
& + \sum_{t \in \mathcal{T}, u \in \mathcal{U}} M_{t,u} \left(m_{t,u}(\mathbf{s}) - \gamma \sum_{\mathbf{s}'} p(\mathbf{s}'|\mathbf{s}, \mathbf{d}) m_{t,u}(\mathbf{s}') \right) \\
& + \sum_{n \in \mathcal{N}} B_n \left(b_n(\mathbf{s}) - \gamma \sum_{\mathbf{s}' \in \mathcal{S}} p(\mathbf{s}'|\mathbf{s}, \mathbf{d}) b_n(\mathbf{s}') \right) \leq c(\mathbf{s}, \mathbf{d}) \quad \forall \mathbf{s} \in \mathcal{S}, \mathbf{d} \in \mathcal{D}(\mathbf{s}).
\end{aligned} \tag{5.24}$$

For easier reference, we introduce the following notation:

$$\begin{aligned}
\chi_{t,n}(\mathbf{s}, \mathbf{d}) &= x_{t,n}(\mathbf{s}) - \gamma \sum_{\mathbf{s}'} p(\mathbf{s}'|\mathbf{s}, \mathbf{d}) x_{t,n}(\mathbf{s}'), \\
\upsilon_{t,u}(\mathbf{s}, \mathbf{d}) &= m_{t,u}(\mathbf{s}) - \gamma \sum_{\mathbf{s}'} p(\mathbf{s}'|\mathbf{s}, \mathbf{d}) m_{t,u}(\mathbf{s}'), \\
\beta_n(\mathbf{s}, \mathbf{d}) &= b_n(\mathbf{s}) - \gamma \sum_{\mathbf{s}'} p(\mathbf{s}'|\mathbf{s}, \mathbf{d}) b_n(\mathbf{s}'),
\end{aligned}$$

where the transition probabilities are defined as in Section 5.2.4. The above expressions represent the current state values minus the expected discounted values of the future state.

We rewrite (5.24), making use of $\chi_{t,n}(\mathbf{s},\mathbf{d})$, $\upsilon_{t,u}(\mathbf{s},\mathbf{d})$, and $\beta_n(\mathbf{s},\mathbf{d})$ to the following LP,

$$\begin{aligned}
\max_{Z_0, X, M, B} \quad & Z_0 + \sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}}\left[\sum_{\mathbf{s}\in\mathcal{S}}\omega(\mathbf{s})x_{t,n}(\mathbf{s})\right]X_{t,n} + \sum_{t\in\mathcal{T}}\sum_{u\in\mathcal{U}}\left[\sum_{\mathbf{s}\in\mathcal{S}}\omega(\mathbf{s})m_{t,u}(\mathbf{s})\right]M_{t,u} + \sum_{n\in\mathcal{N}}\left[\sum_{\mathbf{s}\in\mathcal{S}}\omega(\mathbf{s})b_n(\mathbf{s})\right]B_n \\
\text{s.t.} \quad & (1-\gamma)Z_0 + \sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}}\chi_{t,n}(\mathbf{s},\mathbf{d})X_{t,n} + \sum_{t\in\mathcal{T}}\sum_{u\in\mathcal{U}}\upsilon_{t,u}(\mathbf{s},\mathbf{d})M_{t,u} + \sum_{n\in\mathcal{N}}\beta_n(\mathbf{s},\mathbf{d})B_n \le c(\mathbf{s},\mathbf{d}) \quad \forall\mathbf{s}\in\mathcal{S},\mathbf{d}\in\mathcal{D}(\mathbf{s}),
\end{aligned} \tag{5.25}$$

The choice of $\omega(\mathbf{s})$ did not influence the solution for the MDP, since for any state $\mathbf{s}$ and any feasible solution $\mathbf{v}$ it holds that $v^*(\mathbf{s}) \geq v(\mathbf{s})$, where $\mathbf{v}^*$ is the optimal solution. In the ADP setting, this phenomenon is lost. Therefore, the weight of a state is best chosen proportional to its frequency of occurrence over time [47]. We use the following notation for easier reference,

$$\begin{aligned}
\mathbb{E}_\omega[x_{t,n}] &= \sum_{\mathbf{s}\in\mathcal{S}}\omega(\mathbf{s})x_{t,n}(\mathbf{s}), \quad \forall t,n, \\
\mathbb{E}_\omega[m_{t,u}] &= \sum_{\mathbf{s}\in\mathcal{S}}\omega(\mathbf{s})m_{t,u}(\mathbf{s}), \quad \forall t,u, \\
\mathbb{E}_\omega[b_n] &= \sum_{\mathbf{s}\in\mathcal{S}}\omega(\mathbf{s})b_n(\mathbf{s}), \quad \forall n.
\end{aligned}$$

which denotes the expected value of $x_{t,n}, m_{t,u}, b_n$ under the probability distribution ω, respectively. This notation can be introduced into our LP (5.25). We achieve,

$$\begin{aligned}
\max_{Z_0, X, M, B} \quad & Z_0 + \sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}}\mathbb{E}_\omega[x_{t,n}]X_{t,n} + \sum_{t\in\mathcal{T}}\sum_{u\in\mathcal{U}}\mathbb{E}_\omega[m_{t,u}]M_{t,u} + \sum_{n\in\mathcal{N}}\mathbb{E}_\omega[b_n]B_n, \\
\text{s.t.} \quad & (1-\gamma)Z_0 + \sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}}\chi_{t,n}(\mathbf{s},\mathbf{d})X_{t,n} + \sum_{t\in\mathcal{T}}\sum_{u\in\mathcal{U}}\upsilon_{t,u}(\mathbf{s},\mathbf{d})M_{t,u} + \sum_{n\in\mathcal{N}}\beta_n(\mathbf{s},\mathbf{d})B_n \le c(\mathbf{s},\mathbf{d}), \quad \forall\mathbf{s}\in\mathcal{S},\mathbf{d}\in\mathcal{D}(\mathbf{s}).
\end{aligned} \tag{5.26}$$

Model (5.26) has fewer variables than the original problem (5.19) , $(T \times N + T \times U + N + 1)$, but computationally an intractable number of constraints. To solve this, we will use column generation on its dual. The same approach, to overcome the intractable number of constraints, was done by Bikker et al. [47] and Saure et al. [23], who used a similar affine structure for the value function approximation. The approach of column generation is explained in the Section 5.3.2.

The dual of (5.26) is as follows, let us call (5.27) the *Master problem (MP)*,

$$\begin{aligned}
\min_{Y} \quad & \sum_{\mathbf{s}\in\mathcal{S}}\sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} c(\mathbf{s},\mathbf{d})Y(\mathbf{s},\mathbf{d}), \\
\text{s.t.} \quad & (1-\gamma)\sum_{\mathbf{s}\in\mathcal{S}}\sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} Y(\mathbf{s},\mathbf{d}) = 1, \\
& \sum_{\mathbf{s}\in\mathcal{S}}\sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \chi_{t,n}(\mathbf{s},\mathbf{d})Y(\mathbf{s},\mathbf{d}) \geq \mathbb{E}_\omega[x_{t,n}], \quad \forall t\in\mathcal{T}, n\in\mathcal{N}, \\
& \sum_{\mathbf{s}\in\mathcal{S}}\sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \upsilon_{t,u}(\mathbf{s},\mathbf{d})Y(\mathbf{s},\mathbf{d}) \geq \mathbb{E}_\omega[m_{t,u}], \quad \forall t\in\mathcal{T}, u\in\mathcal{U}, \\
& \sum_{\mathbf{s}\in\mathcal{S}}\sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \beta_n(\mathbf{s},\mathbf{d})Y(\mathbf{s},\mathbf{d}) \geq \mathbb{E}_\omega[b_n], \quad \forall n\in\mathcal{N}, \\
& Y(\mathbf{s},\mathbf{d}) \geq 0, \quad \forall\mathbf{s}\in\mathcal{S},\mathbf{d}\in\mathcal{D}(\mathbf{s}).
\end{aligned} \tag{5.27}$$

The dual variable $Y(\mathbf{s},\mathbf{d})$ can be interpreted as the frequency that action $\mathbf{d}$ is chosen in state $\mathbf{s}$. To explain the column generation method in the next section, let us call the *Restricted Master Problem*

(RMP) to be the program of (5.27), where we do not use the complete set of states $\mathcal{S}$ and set of actions $\mathcal{D}(\mathbf{s})$, but we consider a subset of the states $\mathcal{S}'$ and $\mathcal{D}'(\mathbf{s})$.

5.3.2 Column Generation

Column generation is a method for dealing with LPs that have a huge number of variables compared to a moderate number of constraints [31]. The idea of column generation is to only use the variables associated with state-action pairs that contribute to the optimal solution, since most variables are non-basic and have value zero in the optimal solution. Column generation needs an initial set of feasible state-action pairs, that satisfy the constraints of (5.27). This initial set can be found manually or by an optimization model. Section 5.3.3 states the method for finding the initial set of feasible state-action used in this research. By starting with this small set of feasible state-action pairs, the column generation algorithm adds the state-action pair associated with the most violated constraint in (5.26). This process continues until no primal constraint is violated anymore. Column generation does not guarantee a computational advantage; it could still be the case that all state-action pairs are needed in the basis to get to the optimal solution.

In order to identify the state-action pair that should enter the basis, we solve the following optimization problem, given by (5.28), subject to the constraints belonging to the state (5.1) and action space (5.2). Let us call (5.28) the *Pricing Problem (PP)*:

$$\arg \min_{\mathbf{s}\in\mathcal{S},\mathbf{d}\in\mathcal{D}(\mathbf{s})} \left\{ c(\mathbf{s},\mathbf{d}) - \left[(1-\gamma)\widetilde{Z}_0 + \sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}} \chi_{t,n}(\mathbf{s},\mathbf{d})\widetilde{X}_{t,n} + \sum_{t\in\mathcal{T},u\in\mathcal{U}} \upsilon_{t,u}(\mathbf{s},\mathbf{d})\widetilde{M}_{t,u} + \sum_{n\in\mathcal{N}} \beta_n(\mathbf{s},\mathbf{d})\widetilde{B}_n \right] \right\}, \tag{5.28}$$

where $(\widetilde{Z}_0, \widetilde{X}_{t,n}, \widetilde{M}_{t,u}, \widetilde{B}_n)$ denote the shadow prices of the RMP under the current set of state-action pairs. A state-action pair enters the basis if the solution to (5.28) is less than zero.

The column generation algorithm terminates when no negative solution to (5.28) exists, since then no primal constraint is violated anymore. It is important to incorporate numerical inaccuracies in the computed shadow prices . For this reason it is adviced to use a small tolerance $\epsilon > 0$ when verifying with state-action pair will lead to improvement. The value of ϵ is typically in the order of 10^{-4}[48]. If ϵ is chosen too small, the column generation algorithm may not converge and the pricing problem will produce the same state-action pair every time it is solved. It is said that the solution is "close enough" to optimality to stop iterating when the tolerance of ϵ is accomplished.

Note that the cost function is part of the objective, and the cost function (5.11) is nonlinear in case the risk-pooling objective (5.15) or the risk-spreading objective (5.16) is incorporated. Both cost functions are nonconvex, therefore the pricing problem is a Mixed Integer Non-Linear Program (MINLP). Section 5.3.5 provides information on how we deal with this type of programming.

It is not necessary for column generation to add the column with lowest reduced costs. Any column with negative reduced costs will do. This means, that if finding the optimal column with the above program (5.28) takes too long, we can use approximation algorithms to find a column with reduced costs. If this approximation algorithm cannot find a solution anymore, the above program should be used to prove optimality. Such a scheme can reduce the computation time per iteration, however, the number of iterations may increase and it is not certain that the overall effect is positive [30]. It may also be possible to generate more than one column with negative reduced costs. This could reduce the number of iterations, but increase the computation time per iteration. The solver we use returns a local optimum, therefore it is not certain that the state-action pair with most negative reduced costs is added. By experimenting we do not see a decrease in total computation time when feasible state-action pairs with negative reduced costs, which are worse than the local optimum, are added to the set of state-action pairs, due to the increase of the number iterations.

5.3.3 Initial feasible solution

As stated in Section 5.3.2, the column generation needs an initial feasible solution. We need a diverse set of initial state-action pairs, in order to get a feasible solution to the RMP. We could not find one state-action pair for which a feasible solution to the RMP exists. Therefore, we first create a small set of state-action pairs manually. Then we check if the initial set of state-actions pair is feasible, else we iteratively solve two optimization models to add state-action pairs until feasibility to the RMP can be achieved.

The manual selection of initial state-action pairs is as follows. Since we expect a higher booking percentage in the near future and a lower booking percentage in the far future, we let the initial value of the state variable **x** decrease in time, while we leave space on each day for a possible booking. We set the initial value of the waiting list state variable **m** equal to one in the waiting list for some t and u, and set the other values equal to zero. Next to this we set the state variable **b** equal to one for all days. The list of actions now exists of planning the only patient in the waiting list on a certain day, or not. Therefore the set of initial state-action pairs, created manually, exists of $N+1$ pairs.

Using the above initialization of state-action pairs could still cause in-feasibility of the MP. To overcome this, new state-action pairs should be added until the MP can be solved to feasibility. The method comes down to a column generation method, where the pricing problem for the initialisation is given by (5.30) and the master problem for the initialisation by (5.29). Once θ is equal to zero, it is certain that with the set of state-actions pairs a feasible solution exists for the RMP. The initial set of state-action pairs is manually found and is the start of the column generation algorithm for the initialisation. In order to identify the state-action pairs that should be added, we solve the program (5.30), subject to the constraints on the action- and state-space, which finds the state-action pair for which the constraints in the primal of (5.30) are violated the most. In (5.30) the values of $\widetilde{Z}_0, \widetilde{X}_{t,n}$, $\widetilde{M}_{t,u}$, and $\widetilde{B}_n$ are the shadowprices of the constraints in (5.29).

In order to identify the state-action pairs that should be added, we solve the program (5.30) given by (5.29) and the pricing problem program given by (5.30). When the objective, the minimization of θ, is equal to zero, we know that with the current set of state-actions pairs a feasible solution to (5.27 can be found.

$$
\begin{aligned}
\min_{Y,\theta} \quad & \theta, \\
\text{s.t.} \quad & (1-\gamma)\sum_{\mathbf{s}\in\mathcal{S}}\sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} Y(\mathbf{s},\mathbf{d}) = 1-\theta, \\
& \sum_{\mathbf{s}\in\mathcal{S}}\sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \chi_{t,n}(\mathbf{s},\mathbf{d})Y(\mathbf{s},\mathbf{d}) \geq \mathbb{E}_\omega[x_{t,n}] - \theta, && \forall t\in\mathcal{T}, n\in\mathcal{N}, \\
& \sum_{\mathbf{s}\in\mathcal{S}}\sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \upsilon_{t,u}(\mathbf{s},\mathbf{d})Y(\mathbf{s},\mathbf{d}) \geq \mathbb{E}_\omega[m_{t,u}] - \theta, && \forall t\in\mathcal{T}, u\in\mathcal{U}, \\
& \sum_{\mathbf{s}\in\mathcal{S}}\sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \beta_n(\mathbf{s},\mathbf{d})Y(\mathbf{s},\mathbf{d}) \geq \mathbb{E}_\omega[b_n] - \theta, && \forall n\in\mathcal{N}, \\
& Y(\mathbf{s},\mathbf{d}) \geq 0, && \forall \mathbf{s}\in\mathcal{S}, \mathbf{d}\in\mathcal{D}(\mathbf{s}).
\end{aligned}
\tag{5.29}
$$

$$
\arg\min_{\mathbf{s}\in\mathcal{S},\mathbf{d}\in\mathcal{D}(\mathbf{s})} \Big\{(1-\gamma)\widetilde{Z}_0 + \sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}} \chi_{t,n}(\mathbf{s},\mathbf{d})\widetilde{X}_{t,n} + \sum_{t\in\mathcal{T},u\in\mathcal{U}} \upsilon_{t,u}(\mathbf{s},\mathbf{d})\widetilde{M}_{t,u} + \sum_{n\in\mathcal{N}} \beta_n(\mathbf{s},\mathbf{d})\widetilde{B}_n\Big\}. \tag{5.30}
$$

5.3.4 Approximate Optimal Policy

When the column generation algorithm has terminated, i.e. when no primal constraints are violated anymore, the approximate optimal policy can be obtained, by inserting the optimal values $(Z*_0, X^*_{t,n}, M^*_{t,u}, B^*_n)$ into the right-hand side of the optimality equations (5.18). This would yield the following approximate optimal policy:

$$d^*(\mathbf{s}) \in \arg\min_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \left\{ c(\mathbf{s},\mathbf{d}) + \gamma \left[Z_0^* + \sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}} x_{t,n}(\mathbf{s}')X^*_{t,n} + \sum_{t\in\mathcal{T}}\sum_{u\in\mathcal{U}} m_{t,u}(\mathbf{s}')M^*_{t,u} + \sum_{n\in\mathcal{N}} b_n(\mathbf{s}')B^*_n \right] \right\}, \quad \forall \mathbf{s}\in\mathcal{S}. \tag{5.31}$$

In (5.31) some terms are independent of the decision, therefore we can rearrange the formula into:

$$d^*(\mathbf{s}) \in \arg\min_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \left\{ c(\mathbf{s},\mathbf{d}) + \gamma \left[\sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}} x_{t,n}(\mathbf{s}')X^*_{t,n} + \sum_{t\in\mathcal{T}}\sum_{u\in\mathcal{U}} m_{t,u}(\mathbf{s}')M^*_{t,u} \right] \right\}, \quad \forall \mathbf{s}\in\mathcal{S}. \tag{5.32}$$

The expression (5.32) represents the balance between the direct costs and the combination of the loss of available treatment capacity in the future and the costs of having a patient on the waiting list in the future.

To analyse the quality of the ADP, we can solve the ADP and MDP for a small instance and compare the value function, however if the basis functions are well suited for a small instance, this does not indicate that the same holds for larger instances. In our case we cannot find a direct solution to the MDP for instances where the structure of the model is preserved. Analysing the quality of the AOP is still an ongoing research topic. Section 5.3.4 will analyse the behaviour of the coefficients of the features to indicate if the approximate policy behaves as expected.

5.3.5 Solver

The ALP is modelled in AIMMS 4.72. The column generation algorithm switches between solving the Restricted Master Problem (RMP), with the updated set of state-action pairs, and solving the pricing problem with the shadow prices calculated by the RMP. We solve the RMP with the GUROBI 9.0 solver for Mixed Integer Programming. The pricing problem is an non-convex MINLP. If the pricing problem was convex, a convex solver for MINLP could have been used. Convex solvers use the convexity of the relaxed problem, since a mixed integer problem is by definition non-convex. Nevertheless, the goal of risk-pooling results into a non-convex problem, even though a single square root is concave. Equally the goal of risk-spreading results into a non-convex problem. We use the GMP-AOA module of AIMMS to solve the MINLP, that is the pricing problem. The GMP-AOA module used in this study is not guaranteed to find a global optimal solution [49]. The GMP-AOA-module instead returns a local optimal solution, which could cause more iterations in the column generations algorithm, since not the state-action pair with most negative reduced costs is added. For this, we tried the solver GUROBI 9.0, that returns a global optimal solution, for Mixed Integer Quadratically Constrained Programs (MIQCP), since we can rewrite our MINLP into a MIQLP. By experimenting the choice was made to continue with the GMP-AOA solver, since the column generation algorithm terminated faster in most of the cases than when the pricing problem was solved by GUROBI 9.0. A MINLP can be solved faster by providing an initial solution that help interior point methods to solve the NLP part. Finding an initial feasible solution is not affected by the non-linearity in the objective. For this a dummy objective can be used and one can find a feasible solution from the feasible region and then evaluate the objective function at that point. The method of using an initial solution is incorporated within the GMP AOA module of AIMMS that is used.

5.3.6 A second approach to the value function approximation

In this section we conduct a second approach on the value function approximation. We use again an affine structure of the approximation, however, this time we do not include the state vector **b** as a feature. First we explain why the change of basis functions can lead to a better prediction. In the value function approximation of (5.22), we observed that the optimal choice of weights in most of the cases existed of setting $X^*_{t,n}$ and $M^*_{t,n}$ equal to zero, and having positive values for Z^*_0 and B^*_n. The last two coefficients do however not influence the approximate optimal decision, see (5.32). Therefore, this would indicate that the optimal policy is a myopic policy, where the expected future costs are not taken into account. It is unlikely that the myopic policy is the optimal policy, therefore we expect that we did not make the best choice of basis functions in (5.22).

The fact that $X^*_{t,n}$ and $M^*_{t,u}$ are set equal to zero can be better understood when looking at the approximate linear program of (5.26). The cost function on the right hand side is a non-negative function. The constant Z^*_0 is set to a value to fill some of the slack in the constraint. To fill the remaining slack, the other coefficients can be used. The observed values of the coefficients show that the coefficient B^*_n can optimally fill the remaining slack. To overcome this, we choose to leave out the feature belonging to the **b** state, such that the remaining slack has to be decreased with the values of $X_{t,n}$ and $M_{t,u}$.

The value function is then approximated by:

$$v(\mathbf{s}) = Z_0 + \sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}} X_{t,n}x_{t,n}(\mathbf{s}) + \sum_{t\in\mathcal{T}}\sum_{u\in\mathcal{U}} M_{t,u}m_{t,u}(\mathbf{s}), \quad \forall \mathbf{s}\in\mathcal{S}, \tag{5.33}$$

where $Z_0 \in \mathbb{R}$ and $X_{t,n}, M_{t,u}, \geq 0\ \forall n \in \mathcal{N}, t \in \mathcal{T}$, where $x_{t,n}, m_{t,u}$ follow from the state description and Z_0 is a constant. We have $(1{+}T \times N{+}T \times U)$ basis functions. The goal is to find a good setting of the coefficients $Z_0, X_{t,n}$,and $M_{t,u}$, that approximates the value function. $X_{t,n}$ represents the marginal infinite-horizon discounted cost of booking a patient of type t on day n, $M_{t,u}$ represents the marginal infinite-horizon discounted costs of having one more patient of type t with urgency category u on the waiting list.

We perform similar steps as were performed in Section 5.3.1, to end up with the following approximate linear program:

$$\begin{aligned}
\max_{Z_0, X, M} \quad & Z_0 + \sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}} \mathbb{E}_\omega[x_{t,n}]X_{t,n} + \sum_{t\in\mathcal{T}}\sum_{u\in\mathcal{U}} \mathbb{E}_\omega[m_{t,u}]M_{t,u}, \\
\text{s.t.} \quad & (1-\gamma)Z_0 + \sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}} \chi_{t,n}(\mathbf{s},\mathbf{d})X_{t,n} + \sum_{t\in\mathcal{T}}\sum_{u\in\mathcal{U}} \upsilon_{t,u}(\mathbf{s},\mathbf{d})M_{t,u} \leq c(\mathbf{s},\mathbf{d}), \quad \forall \mathbf{s}\in\mathcal{S}, \mathbf{d}\in\mathcal{D}(\mathbf{s}).
\end{aligned} \tag{5.34}$$

We use the same approach to solve (5.34) as was done for the first approximate linear program (5.26). The complete list of programs and procedures used for solving (5.34) are given in Appendix A.4.

The optimal choice of the coefficients can still be found zero when solving (5.34). This is the case, when there are several state-action pairs in the basis for which the cost function takes on a value close to zero. In the constraints of (5.34), the right hand side is then close to zero for some state-action pairs. This does not directly suppress the coefficients, since the values of $\chi_{t,n}(\mathbf{s},\mathbf{d})$ and $\upsilon_{t,n}(\mathbf{s},\mathbf{d})$ can be less than zero. However, if multiple state-actions pairs are in the basis, for which the cost function is close or equal to zero, then it is likely that the only feasible solution to the coefficients remains the zero solution, since the coefficients do not depend on the state-action pair. This indicates that the initialisation method, where the initial state-action pairs are added to the basis, and the column generation algorithm, that adds state-actions pair associated with the most violated constraint, have a huge influence on the value function approximation. Moreover, it is likely that for larger instances, where on average more state-action pairs contribute to the optimal solution, state-actions pair are in the basis for which the cost function is close to zero. Therefore, the larger the instance gets, the higher the probability that the algorithm returns the zero solution for the coefficients. In the column generation algorithm, we observe that often state-action pairs are added to the basis, which do not occur often in practice. An example of such a state-action pair is when the cost function is low, since this would yield a schedule with a small or empty waiting list, and a completely filled schedule within capacity, where the action is to book none or book the remaining patient on the waiting list onto sessions in which they perfectly fit. It is desired to find the weights of the coefficients through a method in which the cost function is not often close to zero. We leave this problem in this study as a discussion point for further research.

6 Results

In this chapter we present the results of the experiments we carry out to evaluate the performance of the proposed solution approach. The AOP is tested against the FIFO policy and the myopic policy through discrete event simulation. Section 6.1 analyses the AOP by looking at the behaviour of the coefficients of the features for different cases. Since the AOP for the value function approximation of (5.22) shows an undesired behaviour, we use the AOP belonging to the value function approximation of (5.33) through out this chapter. Section 6.2 presents the simulation model and Section 6.3 discusses the results obtained from the simulation model in terms of several performance indicators.

In Section 5.2.5 we formulated two different cost functions for the MDP. The first cost function included an objective for risk-pooling (5.15), the second cost function included an objective for risk-spreading (5.16). Due to the non-linearity that these cost functions have, the complexity of the problem increases. It is in our interest to also investigate the performance of a policy when none of these cost functions of risk-pooling or risk-spreading was used. Therefore, we define the following three AOP policies:

(1) We refer to the first AOP-policy with *AOP-AO*. This AOP policy is found for the MDP where the cost function does not include the D_n, the term to achieve minimal deviation from the end times. The cost function therefore incorporates the objective for minimal access times in an equitable manner where date/times are announced as soon as possible (A_n), and the objective for minimal deviation from the capacity of sessions (O_n).

(2) We refer to the second AOP-policy with *AOP-RP*. This AOP policy is found for the MDP with the cost function of (5.11), where D_n represents the cost function for risk-pooling (5.15).

(3) We refer to the third AOP-policy with *AOP-RS*. This AOP policy is found for the MDP with the cost function of (5.11), where D_n represents the cost function for risk-spreading (5.16).

The AOP strategy incorporates the booking of patient groups. In the simulation we book the patients of this group according to the First-In-First-Out rule.

We present two heuristics to book patients to OR sessions, each motivated by a slightly modified version of the problem. Since these heuristics are derived when some of the assumptions of the original problem are violated, their solutions may not be optimal for the original problem we aim to solve. We compare the approximate optimal policy against the following described policies.

- **FIFO:** The FIFO policy is a reasonable approximation of the current surgical scheduling policy of hospitals [8]. The policy books the surgery with longest waiting time into the first available sessions, where the addition of the expected surgical time does not cause the total surgery time booked in the session to exceed the capacity. The FIFO policy is slightly modified to approximate the current policy in a better way. Patients are booked 30 days (based on the norm for invitation time by the SMK) in advance to an available OR session, where the patient with longest waiting time is booked first, and where patients can be booked until 90% of the capacity is booked. Patients with lower access time target than 30 days, which is the case for acute and emergency patients, are booked when they arrive in the first available session, where capacity does not exceed C^{max}. If the non-elective patient is not treated within their maximal internal access time, a cancellation of a lower priority surgery takes place to create capacity for the higher priority patient.

- **Myopic policy:** The myopic policy is the policy that decides on the best decision to take at one moment in time, without taking into account future events. To obtain the myopic decision in the simulation we solve the program of (6.1) at each decision epoch.

$$d^*(\mathbf{s}) \in \arg\min_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \Big\{c(\mathbf{s},\mathbf{d})\Big\}, \quad \forall \mathbf{s} \in \mathcal{S}, \tag{6.1}$$

where the cost function can differ. We refer to the cost function in the name of the myopic policy in the same way used for the AOP, that is: *Myopic-AO*, *Myopic-RP*, *Myopic-RS*.

The performance metrics of interest collected during the simulation are given in Table 6.1. These are based on the KPIs the SMK uses, introduced in Section 2.2.

Table 6.1.: List of performance metrics collected during the simulation.

Performance metric
Average surgeries per session (Productivity)
Fill rate per session
Average booking accuracy of the sessions
Access times
Invitation times
Number of cancellations due to priority of an emergency

Since the myopic policy and the AOP policy do not incorporate the decision to cancel a patient, a cancellation is counted if the acute patient is not treated within the maximal internal access time.

6.1 Approximate policy insights

In this section, we analyse some scenarios to illustrate some relevant properties of the coefficients of the features. First we look at the behaviour of approximate value function of (5.22). In specific we look at the values of $X^*_{t,n}, M^*_{t,u}, B^*_{t,n}$ for several test cases. Note that B^*_n does not influence the approximate optimal decision. $X^*_{t,n}$ represents the optimal marginal infinite-horizon discounted cost of booking a patient of type t on day n and $M^*_{t,u}$ represents the optimal marginal infinite-horizon discounted costs of having one more patient of type t with urgency category u on the waiting list. During the analysis of the coefficients of the AOP for several scenarios, we observed that in most of the cases the algorithm returns values of the coefficients $X^*_{t,n}$, and $M^*_{t,u}$ equal to zero, which would indicate that the optimal policy does not take into account future events, and thus that the optimal policy is a myopic policy. We carefully examined if a specific characteristic (i.e. more elective than acute patients, high utilization, etc.) of the scenarios could be pointed out to indicate when a zero solution is returned, or when not, however, we could not draw any conclusion from this analysis. In Section 5.3.6 we explained why such an approximation is undesired, and therefore we proposed a second approach to approximate the value function. The analysis in the remainder of this section is done for the coefficients of the value function approximation of (5.33).

The first analysis is for the case of having 1 patient type, with 1 urgency category and having a booking horizon of 10 days. We differ the maximum internal access time of the urgency category, to see how the coefficients behave. In case (1) the internal access time is set to 5 days (that is within booking horizon), in case (2) the internal access time is set to 20 days (outside the booking horizon). We analyse the difference of the coefficients for the different policies of the *AOP-AO*, *AOP-RP*, and *AOP-RS*. The results are shown by Figure 6.1, 6.2, 6.3, and Table 6.2.

We observe that the behaviour of $X^*_{t,n}$ is approximately the same for each approximate optimal policy and acts in the same range. Due to the fact that for case (1) the patients should be treated within 5 days, the value of $X^*_{t,n}$ is larger in the beginning of the booking horizon, and decreases after 5 days, since it brings more expected infinite-horizon discounted costs when a patient is booked in the coming 5 days, in case a new patient arrives. For case (2), $X^*_{t,n}$ tends to behave the same for each day, since it brings cost if a patient is booked in the next ten days, since that leaves less capacity available for a new arrival that should be treated within ten days. $X^*_{t,n}$ equals zero for the last day, due to the definition of the rolling horizon. The value of $X^*_{t,n}$ slightly decreases in n, due to the discount factor. At day zero the value of $X^*_{t,n}$ is relatively low, since booking an extra patient on day zero does not result into extra costs in the next time-step due to the rolling horizon. For $M^*_{t,u}$ we observe that the value is higher for case (1) than for case (2). This is caused by the fact that a patient in case (1) brings more expected discounted costs due to its shorter maximum internal access time. Furthermore, we observe that the values of $X^*_{t,n}$ and $M^*_{t,n}$ become higher when the expected surgery time is increased, or when the expected standard deviation of the surgery time is increased. The marginal infinite-horizon discounted costs for having a patient extra booked on day n that is higher, which is reasonable, since the patient claims a bigger spot, the same reasoning holds for $M^*_{t,u}$.

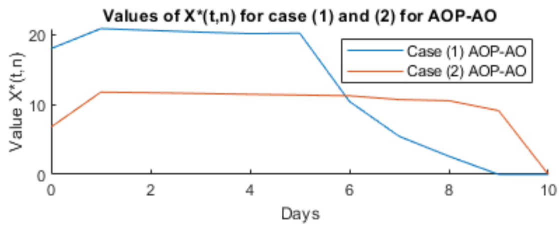

Figure 6.1.: Values of $X^*_{t,n}$ for case (1) and (2) for the AOP-AO policy

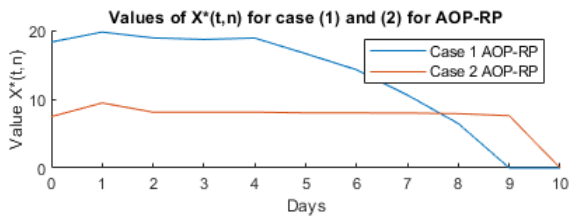

Figure 6.2.: Values of $X^*_{t,n}$ for case (1) and (2) for the AOP-RP policy

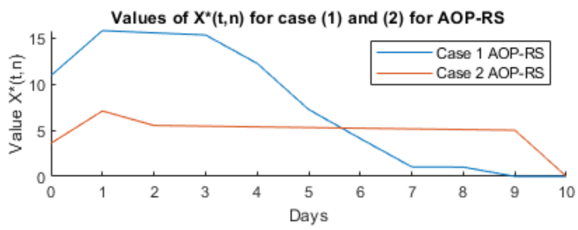

Figure 6.3.: Values of $X^*_{t,n}$ for case (1) and (2) for the AOP-RS policy

Table 6.2.: Values of $M^*_{t,u}$ for case (1) and case (2) per policy.

Policy	$M^*_{t,u}$ (1)	$M^*_{t,u}$ (2)
$AOP-AO$	13.0	10.5
$AOP-RP$	16.0	7.75
$AOP-RS$	7.89	3.28

Analysis of case (1) and (2), with a change in the number of patients to 2 types of patients, show that the behaviour of the coefficients remains the same. The difference of the values per type depend on the expected surgery time and the expected standard deviation of the surgery time. In case $\mu_1 > \mu_2$ and $\sigma_1 = \sigma_2$, we observe that $X^*_{1,n} \geq X^*_{2,n}$ for all days in the booking horizon and that $M^*_{1,u} \geq M^*_{2,u}$. The observation is in line with the desired behaviour, since a patient with a higher expected surgery time results into claiming more time of the session. In case $\mu_1 = \mu_2$ and $\sigma_1 > \sigma_2$, we observe the same behaviour of the coefficients, except for the *AOP-AO* approximation, since the cost function then does not include a cost for the expected standard deviation of the end times. For other combinations of μ and σ a trade-off in the coefficients is observed.

We continue the analysis of the coefficients by introducing more urgency categories and a second patient type. The analysis is done for the coefficients of the *AOP-AO*, since we observed a similar behaviour of the coefficients for each policy, where only the magnitude of the values of the coefficients differ. We define case (3) to have two patient types, five urgency categories with maximum internal access times equal to 2, 14, 30, 60, 180 days, where the booking horizon equals 60 days. Patient type 1 has an expected surgery time of 80 minutes, and patient type 2 has an expected surgery time of 120 minutes. Both patient types have the same expected standard deviation of the surgery time, equal to 25. The results are shown in Figure 6.4 and Figure 6.5. We observe an exponential decay in for $X^*_{t,n}$, which can be explained by the use of the discount factor. The closer we are to the operating day, the more marginal infinite-horizon discounted costs there are for having an extra patient booked. The magnitude of $X^*_{t,n}$ is higher for patient type 2, due to the larger expectation of the surgery time. For $M^*_{t,u}$ we observe that the values are equal for u when the maximum number of treatment days is within the booking horizon ($u = 1, u = 2, u = 3$), and for the ones after the booking horizon ($u = 4, u = 5$). It leaves fewer expected future costs when patients should be treated after the booking horizon, since these patients are not forced to be booked onto sessions within the booking horizon, since there is no cost for booking late, and thus leave more freedom in the schedule than more urgent patients do. We would expect that $M^*_{t,u}$ is higher for the most urgent patient, and decreases in u, since having an acute patient ($u = 1$) on the waiting list has larger impact on the schedule (booking within 2 days) than having an emergency patient ($u = 2$) (booking within 14 days). We could not think of a reasonable explanation for this behaviour of the coefficient.

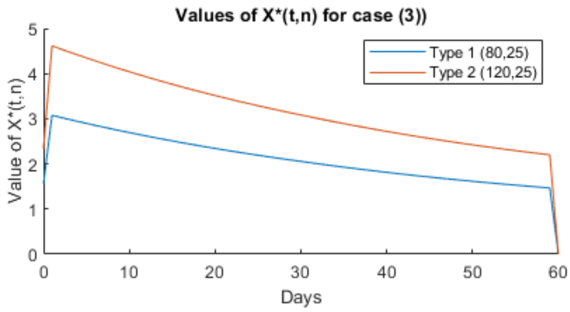

Figure 6.4.: Values of $X^*_{t,n}$ for case (3) for the AOP-AO policy.

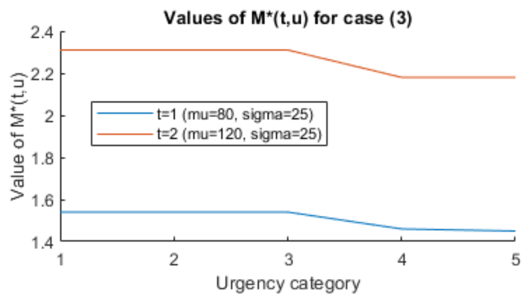

Figure 6.5.: Values of $M^*_{t,u}$ for case (3) for the AOP-AO policy.

Furthermore, we observe that the optimal values are highly influenced by the probability of a cancellation. If this probability is chosen larger, the value of $X^*_{t,n}$ decreases faster in n. Intuitively, this is explained by the fact having an extra patient booked further in the future, has a higher probability to be cancelled, and therefore the future costs for having a patient booked is further in the future decrease faster. Next to this, the other uncertainties can influence the values of the coefficients, however, their influence is observed to be much less, and in most cases none.

For the second value function approximation, we observed that for larger instances, the optimal weights of the coefficients ($X^*_{t,n}$ and $M^*_{T,u}$) were still found to be zero, which would indicate that again the myopic policy is the optimal policy. A large instance is seen as an instance with more than 5 different patient types, 5 urgency categories, and a booking horizon of 60 days. In Section 5.3.6 we discussed that the coefficients are heavily affected by the state-action pairs that are added to the basis in the initialization and during the column generation algorithm: the larger the instance becomes, the higher the probability that some state-action pairs are added for which the value of the cost function is very small, which causes the coefficients to be equal to zero. Large instances for which the coefficients were unequal to zero were not found in this study. However, we do realize that the instance can vary on several characteristics, and that we could not analyse each possible instance within time for the second value function approximation. As we will also state in the discussion in Chapter 7, we recommend to do further analysis on several instances and to do research in finding methods that do not add state-action pairs for which the cost function adopts low values.

The values of the optimal coefficients $X^*_{t,n}$ an $M^*_{t,u}$ that represent the expected discounted future costs compete against the value of the direct cost function in determining the approximate optimal policy through the Bellman equations (5.32). Therefore, the analysis of the coefficients cannot conclude on the complete behaviour of the approximate optimal policy, but it does show some insights on the effect of several choices on the marginal infinite-horizon discounted costs as was discussed in this section.

6.2 Simulation Model

In this section we explain the simulation model that will be used to evaluate the proposed strategies in terms of the performance indicators.

The decision epochs of the MDP are defined to be at the start of each day. The days only consists of the working days, thus excluding Saturday and Sunday. Arrivals during the weekend are spread out over Monday till Friday, since we assumed arrivals are not depending on the day of the week in the MDP. Given an initial state, and given the approximate optimal decision per state, we can simulate the booking process for a given booking horizon. In the simulation uncertainties in the cancellation, urgency updates, arrivals, and session availability are incorporated.

The arrival proces is assumed to be Poisson distributed with arrival rate $\lambda_{t,u}$, depending on the type of patient and the urgency category of the patient. We use the Poisson distribution since the events of an arrival occur independently, and we assume that the probability that an event occurs in the given time does not change through time. Moreover, literature suggest that surgical requests can be best modelled as a Poisson process [44]. The data available in this study is not sufficient to investigate whether or not the arrival process can be modelled by a Poisson process. The Poisson distribution is a special case of the negative binomial distribution. The Poisson distribution assumes that the variance is equal to the expectation, whereas if the variance is larger, this could lead to over dispersion. In such a case, the negative binomial should be used, this is not done within this simulation.

To determine the number of patient types for a respresentative instance of the SMK, we identify the number of unique combinations of the surgery time and standard deviations of the patients of a surgeon. The expected surgery time and expected standard deviation are determined by the HiX-module that the SMK uses. Since this system only uses the surgery time of the last five similar surgeries that the specific surgeon performed for prediction, a lot of patients get the same surgery time and standard deviation assigned, which reduces the number of different patient characteristics. The number of patient types is therefore based on the number of unique combinations of these two characteristics. The number of different patient types has a huge influence on the size of the problem, see Section 5.2.6, therefore it is advised to reduce the number of different patient types, by combining patients with similar characteristics, if possible.

The arrival rate per type of patient t with urgency level u is defined as follows:

$$\lambda_{t,u} = \frac{N_{t,u}}{T}$$

where $N_{t,u}$ is the number of surgeries of patient t with urgency level u during T days. The arrival rate is used for the Poisson process.

We determine the probability of having a session per surgeon by the following calculation:

$$p_{session} = \frac{N}{5W},$$

where N is the number of sessions that the surgeon had during a period of W weeks. With this fraction we determine the probability that the surgeon has a session on a new day. The simulation is most interesting when the server utilization is close to one, too see how the different policies deal with very limited capacity available.

The probability of non-policy-cancellations and the probability of urgency updates is determined in consultation with the OR-personnel. The probability of non-policy cancellations is set to 0.01, meaning that each booked surgery has a probability of 0.01 to be cancelled during each decision epoch. This value is set such that around one third of the patients is cancelled in a common schedule due to non-policy related issues. The probability of urgency updates differs per urgency category. For the highest urgency category the probability to enter a higher urgency category is zero, for the other categories the probability is set equal to one over the difference in maximum days before treatment between the urgency categories, to make sure that elective patients will be booked in time due to an increase in priority level.

$$p_u^{uc} = \begin{cases} 0 & \text{if } \ u = 1 \\ \frac{1}{(n_u - n_{u-1})} & \text{else.} \end{cases}$$

The cost function f_u^{DELAY} is defined such that delaying a patient with higher priority costs more than delaying a patient with low priority, even though the actual values are fairly subjective as it is difficult to quantify the impact of a delay on a patient's health. The choice for f_u^{DELAY} in turn affects $f_{u,n}^{LATE}$ in such a way that booking a patient with higher priority after his or her maximum internal access time, results into a higher costs than doing the same for a patient with lower priority. We chose to let f_u^{DELAY} be a linear function in u: $f_u^{DELAY} = (6 - u)$, such that a patient with highest priority $(u = 1)$ has a delay cost of 5, and with lowest priority $((u = 5)$ a delay cost of 1. We set $\lambda = 0.99$ in the $f_{u,n}^{LATE}$ function (5.13), which induces that is it slightly preferred to book a patient late, that is booking after the maximum internal access time, than delaying the decision to book, in order to increase invitation times.

The simulation is initialized by representative values of the states $\mathbf{x}, \mathbf{m}$, and $\mathbf{b}$, and with starting values for the waiting times of the patients in the system. A representative work stock of a surgeon for the SMK is one where the number of patients on the waiting list is more ore less twice the number of patients that are booked. We initialize $b_n(\mathbf{s})$ by drawing from a Bernoulli distribution with probability $p_{session}$. With the information of $\mathbf{b}$, we initialize $x_{t,n}(\mathbf{s})$. The sessions in the first 30 days are booked for 90%, and the days after for 50% with randomly selected patient types. We initialize $m_{t,u}(\mathbf{s})$ by adding patients to the patient list, for which the arrival rate is larger than zero (i.e. $\lambda_{t,u} > 0$), until twice the number of booked patients, defined by the initialization of $\mathbf{x}$, is reached. The initial waiting times of the patients on the waiting list are uniformly picked between zero and n_u.

Given an initial work stock (a waiting list similar to the waiting list at one point in time of a surgeon) and an initial session plan that is partly booked with surgeries, as defined earlier, we will simulate the proposed strategies: the AOP policy, the FIFO-policy and the myopic-policy. We simulation the schedule of one surgeon. We use a warming up period of the simulation equal 100 days. After the warming up period we simulate the process for one year and keep track of the performances. To generalize the results of the simulation, we run the simulation 20 times. It is preferred to have statistics collected for over 100 runs, however, due to the size of the problem, the simulation takes too long to perform that many runs in time. Therefore, we should be careful with drawing conclusions from results.

A list of the configurations that are the same for the analysed cases in Section 6.3 in the simulation is given in Table 6.3. These values are used, unless stated differently. Other values are mentioned in the text.

Since the simulation model includes a mixed integer non-linear program to determine the AOP at every decision epoch, we implement the simulation model in AIMMS. The simulations in AIMMS were carried out on a Intel(R) Core(TM) i5-5200U CPU (2.19 gigahertz, 8 gigabyte RAM) running Windows 10.

Table 6.3.: Configurations for the simulation.

Parameter	Value	Explanation
C	510	Capacity of the sessions (min).
C^{max}	1.2	Maximum fill-rate of sessions.
D	10	Maximum patients on waiting list per t, u.
Q	10	Maximum number of patients planned per session.
λ	0.99	Trade-off parameter in delaying the booking decision or booking late.
$(\delta : \eta : \zeta)$	$(1:1:1)$	Ratios in the cost function of (5.11) for A_n, O_n, and D_n, respectively.

6.3 Numerical Results

This section shows and compares the performance values found by the simulation for the described policies. Section 6.3.1 focuses on a case, for which the AOP policy can be determined and the simulation can be completed in a suitable computation time. The case consists of 3 patient types, 3 urgency categories, and a booking horizon of 30 days. This does not completely resemble the problem of the SMK, since that would include around 50 different patient types (belong to one surgeon), 5 urgency categories and a booking horizon of 60 days. We could not perform simulations for such a case, since only running the column generation would take an approximated duration of a few weeks. We therefore reduced the instance to a size for which we could the obtain results in time.

Due to the small size of the cases that we test, the simulation model is not a valid representation of the current practice. Therefore, we have to be very careful with drawing conclusions from the simulation results. A second reason to be careful in drawing conclusions is the small number of runs that is performed, which causes the results to be less generalizable.

6.3.1 Case study

We analyse a case consisting of three different patient types, three urgency categories and a booking horizon of 30 days. The patient types and arrival rates for each patient type are given by Table 6.4. The maximal internal access times per urgency category (n_u) are set to 2 days, 14 days and 30 days. The number of arrivals per urgency category are chosen to be equal to the current number of arrivals per urgency category in the SMK, where the 2-month and elective patients are within urgency category 3. Sessions have a capacity of 510 minutes and the probability of having a session is set to 0.49, such that the server utilization ρ equals 0.95.

Table 6.4.: Arrival rates per type of patient for case (1)

Type(μ,σ)	$u = 1$	$u = 2$	$u = 3$
(75,10)	0.041421	0.016543	0.750554
(100,20)	0.006857	0.050756	0.638064
(125,30)	0.0034	0.055092	0.760098
Percentage	3.50%	5.20%	91.30 %

A simulation of 20 runs for this case for the *AOP-AO* policy takes around half an hour, and for the *AOP-RS* or *AOP-RP* policy the simulation takes around 2 hours per policy. The column generation algorithm to find the optimal coefficients of the value function approximation takes around 2 hours for the *AOP-AO* policy, and takes around 10 hours for the other two approximate optimal policies per policy. However, due to the non-linear objective function, used for to the risk-spreading or risk-pooling objective, the computation time can heavily increase by changing some parameters slightly. The mentioned computation times are belonging to the computer described at the end of Section 6.2.

We show the results obtained in the simulation for the seven described policies in Table 6.5.

Table 6.5.: The results of the simulation for the seven described policies.

Performance indicator		*FIFO*	*AOP-AO*	*AOP-RP*	*AOP-RS*	*Myopic-AO*	*Myopic-RP*	*Myopic-RS*
Productivity		4.59	4.88	4.99	4.89	4.89	4.86	4.87
Average fill-rate		0.89	0.96	0.98	0.97	0.96	0.96	0.96
Cancellations per month		1.58	2.53	0.775	2.62	2.19	1.01	1.57
% of acute resulting in cancel		40.60%	68.50%	23.30%	67.00%	59%	28.2%	41.20%
Av. booking accuracy session		0.43	0.41	0.40	0.40	0.40	0.40	0.40
Min. booking accuracy session		0.29	0.30	0.29	0.30	0.30	0.29	0.30
Max. booking accuracy session		0.96	0.70	0.75	0.67	0.70	0.81	0.64
Access - and invitation times								
Performance indicator	u	*FIFO*	*AOP-AO*	*AOP-RP*	*AOP-RS*	*Myopic-AO*	*Myopic-RP*	*Myopic-RS*
Av. Invitation time	1	1.90	1.48	1.25	1.63	1.23	1.35	1.40
Av. Invitation time	2	8.70	8.18	7.15	7.71	7.71	7.64	7.02
Av. Invitation time	3	25.00	12.64	9.84	12.23	14.70	13.44	19.41
Av. Access time	1	2.30	54.24	8.82	58.37	47.16	20.32	25.37
Av. Access time	2	9.10	8.93	7.25	8.738	8.35	7.65	7.37
Av. Access time	3	26.00	12.68	9.86	12.28	14.76	13.44	19.48
% w/i access time target	1	69.00%	25.45%	70.65%	25.12%	34.25%	62.35%	50.79%
% w/i access time target	2	94.00%	88.80%	97.75%	86.59%	91.41%	99.11%	95.75%
% w/i access time target	3	97.00%	98.42%	99.31%	98.31%	97.91%	99.93%	98.28%

With the aid of Table 6.5 we can analyse whether or not the AOP outperforms the FIFO strategy and the myopic policy for this case, under this choice of settings. Moreover, we can analyse the influence that the risk-pooling and risk-spreading objective have compared to when their objectives are not included. The results show that the performances of the AOP policies and the myopic policies are comparable, whereas we would expect that the AOP outperforms the myopic policy in terms of cancellations per month and access times, since it does include the future costs in its decisions. The average booking accuracy of a session does not differ when risk-pooling, risk-spreading or none of both is applied. However, there is a slight difference observed in the maximum booking accuracy of a session, which seems higher when risk-pooling is applied. This observation is in line with what risk-pooling does: creating some session with high variance, and some with low. We analyse the effect of risk-pooling and risk-spreading in more detail later in this section with Table 6.7. The access time targets are not met with the AOP and myopic policies for the acute patients. The *AOP-RP* policy seems to have a positive influence on the number of cancellations that occur due to an arrival of an acute patient that cannot be booked within his or her access time. This leads to a relative high percentage of patients that can be treated within their access time target. The FIFO policy is only outperformed by the AOP in terms of the productivity and the fill-rate, except for the *AOP-RP* policy, that can serve more patients within their access times than the FIFO policy does, and has fewer cancellations. From these results we can conclude that the AOP in the current setting of parameters does not outperform the FIFO or the myopic policy for this specific case. However, the model does give the decision maker some freedom in choice of parameters and importance of objectives, that could affect the performance of the AOP in a positive manner. Therefore we look at some more cases.

The model gives the decision maker some freedom in deciding on the importance of each objective with the ratios defined as $(\delta : \eta : \zeta)$, representing the ratios in the cost function of (5.11) for A_n, O_n, and D_n, respectively. To see the affect of changing the ratio's, we look at two cases. First we analyse what occurs when the value of δ is set higher, such that the objective of treating patients within their maximal internal access times, and booking as soon is possible becomes more important.

Then we analyse what occurs when the value of ζ is set higher, to see if the effect of risk-pooling and risk-spreading occurs.

We first look at the case when the ratio's are set to $(1000 : 1 : 1)$. Here δ is set to 1000. We experimented with some deltas smaller than 1000, however, the difference in performance becomes more clear when a large δ is chosen. Results are given by Table 6.6. The results show that changing the ratio's this way improves the values of most performances compared to the setting of ratio's used in Table 6.5, which was set to $(1 : 1 : 1)$. Still, the access times for the acute patients are on average higher than the norm. However, the percentage of patients from this group that were treated within their access time target is relatively high. The AOP outperforms the FIFO policy slightly on the access time targets, on the expense of having a lower invitation time. In the results of Table (6.6) there is no clear difference between the two policies, which indicates that the approximate optimal policy does not take into account the future costs in the correct way, for this case and this choice of settings. There is a difference in the average access times for urgency category 1 an 3, where the access times for the *Myopic-AO* strategy are higher, however, this is difference in performance is not reflected in the percentage of patients who are booked within their access time target.

Table 6.6.: The results of the simulation for a ratio setting of $(1000 : 1 : 1)$ for the *AOP-AO* policy and the *Myopic-AO* policy.

Performance indicator		*AOP-AO*	*Myopic-AO*
Value of the ratio's $(\delta : \eta : \zeta)$		$(1000 : 1 : 0)$	$(1000 : 1 : 0)$
Productivity		5.00	5.07
Average fill-rate		0.98	1.00
Cancellations per month		0.65	0.611
% of acute resulting in cancel		19.4%	18.8%
Av. Booking accuracy session		0.39	0.38
Min. Booking accuracy session		0.29	0.30
Max. Booking accuracy session		0.70	0.64
Access - and invitation times			
Performance indicator	u	*AOP-AO*	*Myopic-AO*
Av. Invitation time	1	1.26	1.27
Av. Invitation time	2	7.29	7.14
Av. Invitation time	3	11.64	18.64
Av. Access time	1	7.37	8.49
Av. Access time	2	7.35	7.23
Av. Access time	3	11.65	18.66
% w/i access time target	1	73.06%	72.77%
% w/i access time target	2	98.90%	98.00%
% w/i access time target	3	99.78%	99.43%

Since Table 6.5 did not directly show a clear difference in the effect that risk-pooling or risk-spreading has, we look at the case when the ratio for the cost for the expected deviation of the end times is set higher than 1. To maintain that the cost for not booking a patient is higher than the cost that are incorporated for achieving the risk-pooling or risk-spreading effect, we cannot choose the ratio ζ too high compared to the ratio η. For risk-pooling we analyse the results for the ratios $(1 : 1 : 100)$. For risk-spreading we analyse the results for the ratios $(1 : 1 : 2500)$. The results of the performances are shown in Table 6.7.

The *Myopic-AO* policy performs better in productivity, but worse in cancellations compared to the *Myopic-RP* and *Myopic-RS* policies. The last two policies are delaying their decision to book, which causes very low invitation times, but they can serve patients within their maximal internal access time better than *Myopic-AO*. Interesting in this comparison of policies is the booking accuracy

Table 6.7.: The results of the simulation in case the objective for decreasing the earliness and tardiness costs of the end times of session is more important.

Performance indicator		*Myopic-AO*	*Myopic-RP*	*Myopic-RS*
Value of the ratio's ($\delta : \eta : \zeta$)		$(1:1:0)$	$(1:1:100)$	$(1:1:2500)$
Productivity		4.89	4.76	4.62
Average fill-rate		0.96	0.93	0.90
Cancellations per month		2.17	1.94	2.43
% of acute resulting in cancel		58%	35.62%	36.95%
Av. Booking accuracy session		0.39	0.42	0.41
Min. Booking accuracy session		0.31	0.31	0.33
Max. Booking accuracy session		0.65	0.89	0.72
Access - and invitation times				
Performance indicator	u	*Myopic-AO*	*Myopic-RP*	*Myopic-RS*
Av. Invitation time	1	1.18	0.92	0.99
Av. Invitation time	2	7.815	1.07	1.57
Av. Invitation time	3	19.38	0.96	5.74
Av. Access time	1	36.14	4.85	4.52
Av. Access time	2	8.40	2.63	3.12
Av. Access time	3	19.46	2.56	6.63
% w/i access time target	1	39.26%	57.26%	53.28%
% w/i access time target	2	93.58%	99.52%	99.8%
% w/i access time target	3	97.05%	100%	100%

of the sessions. The average booking accuracy only shows a small difference between *Myopic-RP* and *Myopic-RS*, where risk-pooling performs better, however, the minimum and maximum booking accuracy show a larger difference: risk-pooling takes on a lower minimum booking accuracy ($0.31 < 0.33$), and takes on a higher maximum booking accuracy ($0.89 > 0.72$). Moreover, when we closely look at the observed booking accuracy of the sessions, we see that the risk-pooling and risk-spreading effect has taken place. Figure 6.6 (a) shows the occurrences of booking accuracy in one year for the risk-pooling policy, Figure 6.6 (b) shows the occurrences of booking accuracy in one year for the risk-spreading policy, and Figure 6.6 (c) shows the occurrences of booking accuracy in one year when none of the risk-pooling or risk-spreading objectives are applied. Observe that the risk-spreading method Figure 6.6(b) heavily concentrates the booking accuracy's around 0.41, whereas the risk-pooling method Figure 6.6(a) has some sessions with really low booking accuracy, and some session with really high booking accuracy. Furthermore, the risk-spreading method spreads the risk a bit more than when the objective is not applied. This can be seen since in Figure 6.6 (c) the booking accuracy close to 0.30 occurs more than 10 times, whereas the risk-spreading approach yields no occurrences of the booking accuracy equal to 0.30 or lower. Note that the occurrences of very high booking accuracy are mainly for sessions that were not completely filled and/or existed of surgeries with low expected standard deviation. To verify that risk-pooling and risk-spreading is really applied, a more in-depth study of multiple (larger) cases should be done.

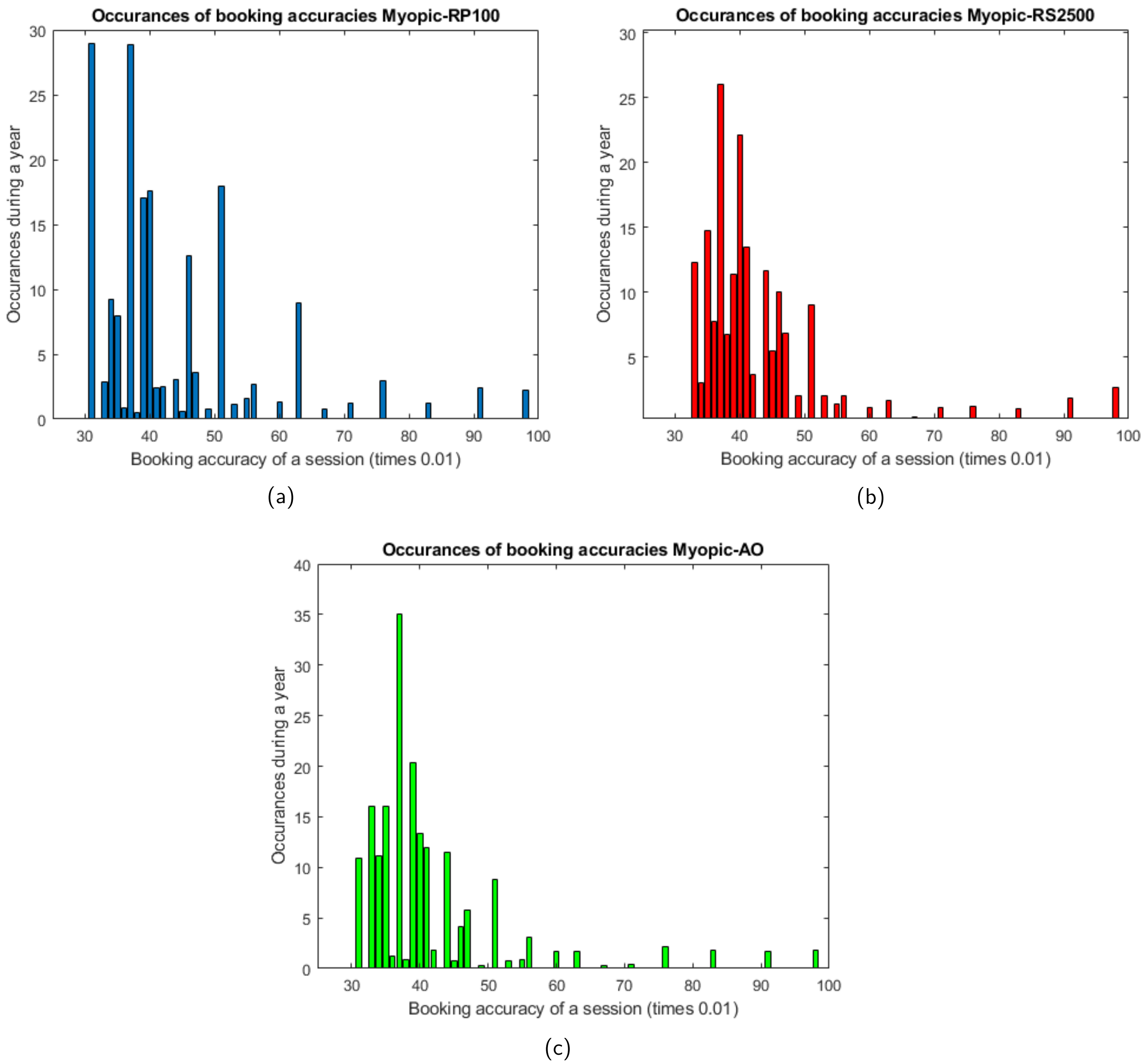

Figure 6.6.: The number of occurrences of the values of the booking accuracy for the three different policies *Myopic-RP* (a), *Myopic-RS* (b), and *Myopic-AO* (c) of Table 6.7.

In this section we looked at applying the different policies to one case through simulation, where we differed the ratios of the objectives. We observed that for this specific case, and for the specified settings, the AOP outperforms the FIFO policy in terms of productivity and access times for non-acute patients, but performs worse in terms of cancellations and access times of acute patients. The AOP does not outperform the myopic policy. A change in ratio's showed an improvement in performance of the AOP. More different settings of ratio's should be analysed, to see if the AOP can outperform the FIFO policy. Moreover, we do realize that there are more parameters that can be differed in the model, that create new situations for which it is also interesting to see how the AOP acts against the FIFO and myopic policy. Interesting scenarios are changing the workload of the system, changing the number of arrivals per urgency category, or changing the patient types for a case having only long surgeries, or a case having only short surgeries, or for a more diverse set of surgeries. We therefore recommend to perform a more detailed analysis on multiple (larger) cases, using a validated simulation model, and using more simulation runs, in order to generalize and verify whether or not the AOP outperforms the FIFO strategy and myopic policy or not. The first simulations tend to show that the risk-pooling approach really pools variance, and thereby decreases the average booking accuracy of a session. Next to this, the risk-spreading approach shows the spreading of risk, such that less sessions have a high probability of ending late. However, these observations were made when the ratio for the expected and earliness tardiness cost of the end time of a session in the objective was set really high. For a smaller ratio, a more detailed analysis should be performed to see how much the risk-pooling and risk-spreading approach affect the schedule, since incorporating these objectives highly affect the complexity of the model by making it non-linear, and therefore it is important to examine the effect both approaches.

7 Conclusion & Discussion

In this thesis, we presented a methodology for the offline operational OR-patient booking. The method was designed to incorporate a combination of equitable patient booking with minimal access times, maximal productivity to be cost-effective, and minimal earliness and tardiness costs of the end times of sessions. We proposed two methods to affect the expected deviation from the end times of sessions in the schedule, referred to as the risk-pooling and the risk-spreading method. The model considers offline decision-making regarding multi-priority and patient booking. The problem is formulated as an MDP, for which an ADP algorithm is developed to obtain approximate solutions. We used the linear programming approach to ADP. For this we chose an affine approximation to the value function, for which we found the optimal setting of the coefficients by the use of column generation. With the optimality equations the approximate optimal policy was determined. We provided insights in the approximated value function and evaluated the performance of the AOP through simulation.

The model is computationally found to be expensive, due to the size of the problem, and the non-linearity of the objective function. Therefore we applied the model to a small case study, for which results could be obtained in time. The results intend to show that the AOP does outperform the FIFO-policy in terms of productivity and access times for non-acute patients, but performs worse in terms of cancellations and access times of acute patients. However, the results do indicate that the AOP is very sensitive to the choice of the ratio's in the objective, and that performances can be improved if the settings of the ratio's is chosen differently by the decision-maker. The evaluation of the AOP through simulation does not show a significant improvement over the myopic policy. This can indicate that the choice of the value function approximation was wrong. However, one should consider the results of the simulation non-conclusive, even though it is an indication for how the AOP tends to behave for the analysed case. To obtain a general conclusion on the performance of the AOP, a validation of the simulation with more runs and a detailed analysis of the performance on the AOP on multiple larger instances is needed.

In this research we proposed two methods to deal with the expected earliness and tardiness costs of the end times of the sessions: risk-pooling and risk-spreading. Both methods were implemented in the model. The complexity of the model increases due to the non-linearity in the objective function. This causes a trade off in the computation time against the quality of the solution. The simulation shows, that for a high ratio of the objective for the costs of the expected earliness tardiness of the end times of session in comparison to the other ratio's, both the risk-pooling and the risk-spreading method affect the schedule. Risk-pooling improves the average booking accuracy of the sessions by clustering the surgeries with high variance and the surgeries with low variance. Risk-spreading decreases the difference between the smallest probability booking accuracy and the highest booking accuracy, such that less sessions occur with a very high expected deviation of the end time. It is left to the decision maker to decide on the importance of this objective. We recommend to do a more detailed analysis on the influence of risk-pooling and risk-spreading for multiple cases.

For future research, we recommend to do research on methods that can solve the master problem, where no state-action pairs are used for which the cost function is equal to zero, since these state-action pairs heavily affect the coefficients of the value function approximation. We recommend a more in depth study on the analytical structure of the ADP. A clear structure, such as a threshold policy, could lead to easy-to-use heuristics. Furthermore, future research should look at the benefit of some more complex approximation architectures to improve the value function approximation, that could decrease the number of variables further, since the experiences in this study show that master problem still had to many variables for larger instances. We do recommend to investigate the allowance for rescheduling capacity in the decision-making process, where a cost for rescheduling is incorporated. This could improve on treating the acute patients within their maximum internal access times, since the results of the simulation show that the AOP finds it difficult to incorporate the arrival of acute patients. With respect to the implementation of this method for the SMK, we would advice to first perform a more detailed analysis on the performance of the AOP.

Bibliography

[1] M. Samudra, C. Van Riet, E. Demeulemeester, B. Cardoen, N. Vansteenkiste, and F. E. Rademakers. Scheduling operating rooms: achievements, challenges and pitfalls. *Journal of Scheduling*, 19(5):493–525, 2016.

[2] Françoise Johansen, Derk Loorbach, and Annemiek Stoopendaal. Exploring a transition in dutch healthcare. *Journal of health organization and management*, 2018.

[3] Brecht Cardoen, Erik Demeulemeester, and Jeroen Beliën. Operating room planning and scheduling: A literature review. *European journal of operational research*, 201(3):921–932, 2010.

[4] Van Enk J. Van Houdenhoven, M. *Jaarverslag 2019: Stichting Sint Maartenskliniek*, 2019.

[5] ICT and Health. Sint maartenskliniek neemt hypermodern ok-complex in gebruik. https://www.icthealth.nl/nieuws/sint-maartenskliniek-neemt-hypermoderne-ok-complex-in-gebruik/, 2019.

[6] A. Testi and E. Tànfani. Tactical and operational decisions for operating room planning: Efficiency and welfare implications. *Health Care Management Science*, 12(4):363, 2009.

[7] G. Wullink, M. Van Houdenhoven, E. W. Hans, J. M. van Oostrum, M. van der Lans, and G. Kazemier. Closing emergency operating rooms improves efficiency. *Journal of Medical Systems*, 31(6):543–546, 2007.

[8] D. Astaraky and J. Patrick. A simulation based approximate dynamic programming approach to multi-class, multi-resource surgical scheduling. *European Journal of Operational Research*, 245(1):309–319, 2015.

[9] J. van Oostrum. *Applying mathematical models to surgical patient planning*. Number EPS-2009-179-LIS. Erasmus Research Institute of Management, 2009.

[10] A. J. T. Schneider, J. T. van Essen, M. Carlier, and E. W. Hans. Scheduling surgery groups considering multiple downstream resources. *European journal of operational research*, 282(2):741–752, 2020.

[11] J. T. van Essen, E. W. Hans, J. L. Hurink, and A. Oversberg. Minimizing the waiting time for emergency surgery. *Operations Research for Health Care*, 1(2-3):34–44, 2012.

[12] M. Heng and J. G. Wright. Dedicated operating room for emergency surgery improves access and efficiency. *Canadian Journal of Surgery*, 56(3):167, 2013.

[13] B. Roland, C. Di Martinelly, F. Riane, and Y. Pochet. Scheduling an operating theatre under human resource constraints. *Computers & Industrial Engineering*, 58(2):212–220, 2010.

[14] B. Denton, J. Viapiano, and A. Vogl. Optimization of surgery sequencing and scheduling decisions under uncertainty. *Health care management science*, 10(1):13–24, 2007.

[15] M. J. C. Eijkemans, M. Van Houdenhoven, T. Nguyen, E. Boersma, E. W. Steyerberg, and G. Kazemier. Predicting the unpredictablea new prediction model for operating room times using individual characteristics and the surgeon's estimate. *Anesthesiology: The Journal of the American Society of Anesthesiologists*, 112(1):41–49, 2010.

[16] J.W.M. Otten, A. Braaksma, and R. J. Boucherie. Minimizing earliness/tardiness costs on multiple machines with an application to surgery scheduling. *Operations research for health care*, 22:100–194, 2019.

[17] D. P. Strum, J. H. May, and L. G. Vargas. Modeling the uncertainty of surgical procedure times: comparison of log-normal and normal models. *Anesthesiology*, 92(4):1160–1167, 2000.

[18] M. Van Houdenhoven, J. M. Van Oostrum, E. W. Hans, G. Wullink, and G. Kazemier. Improving operating room efficiency by applying bin-packing and portfolio techniques to surgical case scheduling. *Anesthesia & Analgesia*, 105(3):707–714, 2007.

[19] E. Hans, G. Wullink, M. Van Houdenhoven, and G. Kazemier. Robust surgery loading. *European Journal of Operational Research*, 185(3):1038–1050, 2008.

[20] S. M. Ross. Introduction to probability models 10th edition academic press. 2010.

[21] M. L. Puterman. *Markov Decision Processes Discrete Stochastic Dynamic Programming*. John Wiley Sons, Inc., Hoboken, New Jersey, 2005.

[22] W. B. Powell. *Approximate Dynamic Programming: Solving the curses of dimensionality*, volume 703. John Wiley & Sons, 2007.

[23] A. Saure, J. Patrick, S. Tyldesley, and M. L. Puterman. Dynamic multi-appointment patient scheduling for radiation therapy. *European Journal of Operational Research*, 223(2):573–584, 2012.

[24] J. Patrick, M. L. Puterman, and M. Queyranne. Dynamic multipriority patient scheduling for a diagnostic resource. *Operations research*, 56(6):1507–1525, 2008.

[25] A. Erdelyi and H. Topaloglu. Approximate dynamic programming for dynamic capacity allocation with multiple priority levels. *IIE Transactions*, 43(2):129–142, 2010.

[26] Y. Gocgun and M. L. Puterman. Dynamic scheduling with due dates and time windows: an application to chemotherapy patient appointment booking. *Health care management science*, 17(1):60–76, 2014.

[27] D. P. De Farias and B. Van Roy. The linear programming approach to approximate dynamic programming. *Operations research*, 51(6):850–865, 2003.

[28] J. W. Daniel. Splines and efficiency in dynamic programming. *Journal of Mathematical Analysis and Applications*, 54(2):402–407, 1976.

[29] Y. Gocgun and A. Ghate. Lagrangian relaxation and constraint generation for allocation and advanced scheduling. *Computers & Operations Research*, 39(10):2323–2336, 2012.

[30] C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh, and P. H. Vance. Branch-and-price: Column generation for solving huge integer programs. *Operations research*, 46(3):316–329, 1998.

[31] B. Manthey. *Optimization Modelling*. University of Twente, without date.

[32] C. Combes, N. Meskens, C. Rivat, and J-P. Vandamme. Using a kdd process to forecast the duration of surgery. *International Journal of Production Economics*, 112(1):279–293, 2008.

[33] S. P. Devi, K. S. Rao, and S. S. Sangeetha. Prediction of surgery times and scheduling of operation theaters in optholmology department. *Journal of medical systems*, 36(2):415–430, 2012.

[34] V. Bellini, M. Guzzon, B. Bigliardi, M. Mordonini, S. Filippelli, and E. Bignami. Artificial intelligence: A new tool in operating room management. role of machine learning models in operating room optimization. *Journal of Medical Systems*, 44(1):20, 2020.

[35] M. Fairley, D. Scheinker, and M. L. Brandeau. Improving the efficiency of the operating room environment with an optimization and machine learning model. *Health care management science*, 22(4):756–767, 2019.

[36] Z. ShahabiKargar, S. Khanna, N. Good, A. Sattar, J. Lind, and J. O'Dwyer. Predicting procedure duration to improve scheduling of elective surgery. In *Pacific Rim International Conference on Artificial Intelligence*, pages 998–1009. Springer, 2014.

[37] S. Dreiseitl and L. Ohno-Machado. Logistic regression and artificial neural network classification models: a methodology review. 35(5-6):352–359, 2002.

[38] I. Kuin. Internship operating room planning zgt. 2020. University of Twente.

[39] N. Master, Z. Zhou, D. Miller, D. Scheinker, N. Bambos, and P. Glynn. Improving predictions of pediatric surgical durations with supervised learning. *International Journal of Data Science and Analytics*, 4(1):35–52, 2017.

[40] I. H. Wright, C. Kooperberg, B. A. Bonar, and G. Bashein. Statistical modeling to predict elective surgery time. comparison with a computer scheduling system and surgeon-provided estimates. *Anesthesiology*, 85(6):1235–1245, 1996.

[41] J. Friedman, T. Hastie, and R. Tibshirani. *The elements of statistical learning*, volume 1. Springer New York, 2001.

[42] Harish Guda, Milind Dawande, Ganesh Janakiraman, and Kyung Sung Jung. Optimal policy for a stochastic scheduling problem with applications to surgical scheduling. *Production and Operations Management*, 25(7):1194–1202, 2016.

[43] N. B. Mehta, J. Wu, A. F. Molisch, and J. Zhang. Approximating a sum of random variables with a lognormal. *IEEE Transactions on Wireless Communications*, 6(7):2690–2699, 2007.

[44] Belinda Spratt, Erhan Kozan, and Michael Sinnott. Analysis of uncertainty in the surgical department: durations, requests and cancellations. *Australian Health Review*, 43(6):706–711, 2019.

[45] M. L. Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, Hoboken, New Jersey, The address, 2014.

[46] D. Bertsimas and J. N. Tsitsiklis. *Introduction to linear optimization*, volume 6. Athena Scientific Belmont, MA, 1997.

[47] I. A. Bikker, M. R. K. Mes, A. Sauré, and R. J. Boucherie. Online capacity planning for rehabilitation treatments: an approximate dynamic programming approach. *Probability in the Engineering and Informational Sciences*, pages 1–25, 2018.

[48] J. Bisschop. *AIMMS optimization modeling*. Lulu. com, 2006.

[49] M. Hunting. The aimms outer approximation algorithm for minlp. *Paragon Decision Technology, Haarlem*, 2011.

Appendices

A Appendix

A.1 Capacity levels of the operating rooms

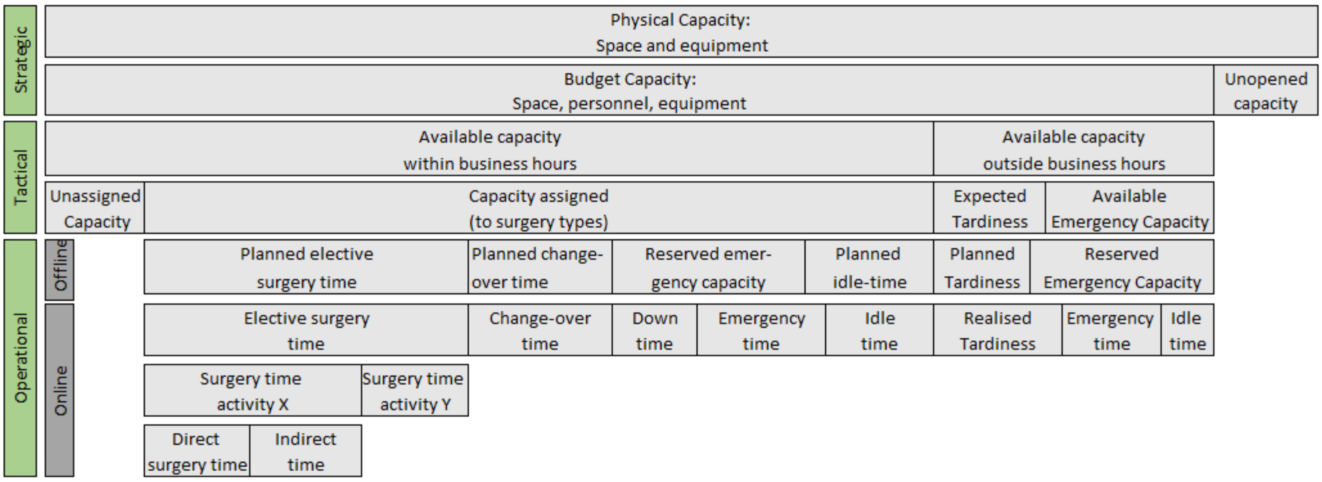

Figure A.1.: Structure of the utilisation of the capacity of the operating rooms.

A.2 Data features

This section sums up a list of features that possibly influence surgery times and thus could be used as predicting variables for the forecasting of surgery times, which in turn could improve the booking accuracy. Measurements of a few variables can be used for this study.

- Location (Nijmegen/Boxmeer).
- Date of surgery.
- Treatmentcode.
- Surgery unit.
- Main transactioncode.
- Cutting surgeon.
- Surgery time (existing of: preparation time, cutting time, discharge-time).
  - Proposed by HiX.
  - Expected by surgeon.
  - Planned by the planner.
  - Realised.
- Startime of the surgery.
- Endtime of the surgery.
- Change-over time.
- Idle time before surgery.
- Acute surgery (Yes/No).
- Planned start time session.
- Planned end time session.
- Responsible anaesthesiologist.
- Assistent.
- Attendence of a doctor in training.
- Number of weeks since the surgeon performed a similar surgery.

Features that include patient information:

- Weight of the patient.
- Length of the patient.
- Age of the patient.
- Gender of the patient.
- ASA classifiction patient.
- Mallampati Classification patient (has influence on the preparation time).
- Medical history of the patient.

A.3 Derivation ET costs formula for normal distributed surgery times

This appendix derives the function for calculating the ET costs of an operating room, given a schedule and assuming that surgery times are independent and normally-distributed.

Given a schedule, the optimal due date of job j with completion time C_j is

$$d_j^* = G^{-1}\left(\frac{\beta_j}{\alpha_j + \beta_j}\right), \tag{A.1}$$

with G the distribution function of C_j [16]. If we assume $\alpha_j = \beta_j, \forall j$, then the optimal due dates are given by $d_j^* = G^{-1}(\frac{1}{2})$. If we further assume normal distributed operating times, then the completion times are also normal distributed, and thus the due dates are always equal to $d_j^* = \mathbb{E}[C_j]$, since $G^{-1}(\frac{1}{2})$ equals the mean of the distribution function G.

We then calculate the ET costs with the following formula, where G represents the distribution function of the completion time of surgery j, which is assumed to be normally distributed.

$$\begin{aligned}
ET_{costs} &= \sum_j F_j^{\pi}(d_j^*) \\
&= \sum_j \left(\beta_j \int_{d_j^*}^{\infty} x dG(x) - \alpha_j \int_{-\infty}^{d_j^*} x dG(x) \right) \\
&= \sum_j \left(\beta_j \int_{d_j^*}^{\infty} x \frac{1}{\sqrt{2\pi}\sigma_j} e^{-\frac{(x-\mu_j)^2}{2\sigma_j^2}} dx - \alpha_j \int_{-\infty}^{d_j^*} x \frac{1}{\sqrt{2\pi}\sigma_j} e^{-\frac{(x-\mu_j)^2}{2\sigma_j^2}} dx \right) \\
&= \sum_j \left(\beta_j \left(\sigma_j^2 \frac{1}{\sqrt{2\pi}\sigma_j} e^{-\frac{(d_j^*-\mu_j)^2}{2\sigma_j^2}} + \mu_j \frac{\alpha_j}{\alpha_j + \beta_j} \right) - \alpha_j \left(-\sigma_j^2 \frac{1}{\sqrt{2\pi}\sigma_j} e^{-\frac{(d_j^*-\mu_j)^2}{2\sigma_j^2}} + \mu_j \frac{\beta_j}{\alpha_j + \beta_j} \right) \right) \\
&= \sum_j (\alpha_j + \beta_j)\sigma_j^2 g(d_j^*).
\end{aligned}$$

We have assumed symmetry (i.e. $\alpha_j = \beta_j, \forall j$), lets say $\alpha = \beta = \frac{1}{2}$. Now the ET costs are

$$ET_{costs} = \sum_j \sigma_j^2 g(d_j^*),$$

where g is the probability density function of the completion time of surgery j and σ_j^2 the variation of the completion time of surgery j.

$$g(d_j^*) = \frac{1}{\sqrt{2\pi}\sigma_j} e^{-\frac{(d_j^*-\mu_j)^2}{2\sigma_j^2}}.$$

Now we have $d_j^* = G^{-1}(\frac{1}{2}) = \mathbb{E}[C_j] = \mu_j$, and thus the e gets a power equal to zero, and we are only left with the following ET costs:

$$\begin{aligned}
ET_{costs} &= \sum_j \sigma_j^2 \frac{1}{\sqrt{2\pi}\sigma_j} \\
&= \frac{1}{\sqrt{2\pi}} \sum_j \sigma_j
\end{aligned}$$

This is just the sum over all surgeries of all the standard deviations of the completion times of each surgery.

A.4 A second approach to the value function approximation

The approximation of the value function is given in Section 5.3.6. Next to this the approximate linear program was given by Model (5.34). Model (5.27) has a tractable number of variables ($1+T \times N+T \times U$), but still an intractable number of constraints. Therefore, we solve its dual, given by (A.2), using column generation.

$$\begin{aligned}
\min_{Y} \quad & \sum_{\mathbf{s}\in\mathcal{S}} \sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} c(\mathbf{s},\mathbf{d})Y(\mathbf{s},\mathbf{d}), \\
\text{s.t.} \quad & (1-\gamma)\sum_{\mathbf{s}\in\mathcal{S}} \sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} Y(\mathbf{s},\mathbf{d}) = 1, \\
& \sum_{\mathbf{s}\in\mathcal{S}} \sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \chi_{t,n}(\mathbf{s},\mathbf{d})Y(\mathbf{s},\mathbf{d}) \geq \mathbb{E}_\omega[x_{t,n}], \quad \forall t\in\mathcal{T}, n\in\mathcal{N}, \\
& \sum_{\mathbf{s}\in\mathcal{S}} \sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \upsilon_{t,u}(\mathbf{s},\mathbf{d})Y(\mathbf{s},\mathbf{d}) \geq \mathbb{E}_\omega[m_{t,u}], \quad \forall t\in\mathcal{T}, u\in\mathcal{U}, \\
& Y(\mathbf{s},\mathbf{d}) \geq 0, \qquad \forall \mathbf{s}\in\mathcal{S}, \mathbf{d}\in\mathcal{D}(\mathbf{s}).
\end{aligned} \tag{A.2}$$

In order to identify the state-action pair that should enter the basis, we solve the following optimization problem, given by (A.3), subject to the constraints belonging to the state (5.1) and action space (5.2).

$$\arg\min_{\mathbf{s}\in\mathcal{S},\mathbf{d}\in\mathcal{D}(\mathbf{s})} \left\{ c(\mathbf{s},\mathbf{d}) - \left[(1-\gamma)\widetilde{Z}_0 + \sum_{n\in\mathcal{N}} \sum_{t\in\mathcal{T}} \chi_{t,n}(\mathbf{s},\mathbf{d})\widetilde{X}_{t,n} + \sum_{t\in\mathcal{T},u\in\mathcal{U}} \upsilon_{t,u}(\mathbf{s},\mathbf{d})\widetilde{M}_{t,u} \right] \right\}, \tag{A.3}$$

where $(\widetilde{Z}_0, \widetilde{X}_{t,n}, \widetilde{M}_{t,u})$ denote the optimal values of the dual LP (A.2) under the current set of state-action pairs.

The manual initialization of state-actions pair is done the same as in Section 5.3.3. To add further state-action pairs two optimization models are used, that slightly differ from the ones mentioned in Section 5.3.3, since the feature $b_n(\mathbf{s})$ has been left out of the approximation. We solve the program (A.5) given by (A.4) and the pricing problem program given by (A.5) to add state-actions pairs to the basis. In (A.5) the values of $\widetilde{Z}_0$, $\widetilde{X}_{t,n}$, $\widetilde{M}_{t,u}$, and are the shadow prices of the constraints in (A.4). When the objective, the minimization of θ, is equal to zero, we know that with the current set of state-actions pairs a feasible solution to (A.2) can be found.

$$\begin{aligned}
\min_{Y,\theta} \quad & \theta, \\
\text{s.t.} \quad & (1-\gamma)\sum_{\mathbf{s}\in\mathcal{S}} \sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} Y(\mathbf{s},\mathbf{d}) = 1-\theta, \\
& \sum_{\mathbf{s}\in\mathcal{S}} \sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \chi_{t,n}(\mathbf{s},\mathbf{d})Y(\mathbf{s},\mathbf{d}) \geq \mathbb{E}_\omega[x_{t,n}] - \theta, \quad \forall t\in\mathcal{T}, n\in\mathcal{N}, \\
& \sum_{\mathbf{s}\in\mathcal{S}} \sum_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \upsilon_{t,u}(\mathbf{s},\mathbf{d})Y(\mathbf{s},\mathbf{d}) \geq \mathbb{E}_\omega[m_{t,u}] - \theta, \quad \forall t\in\mathcal{T}, u\in\mathcal{U}, \\
& Y(\mathbf{s},\mathbf{d}) \geq 0, \qquad \forall \mathbf{s}\in\mathcal{S}, \mathbf{d}\in\mathcal{D}(\mathbf{s}).
\end{aligned} \tag{A.4}$$

$$\arg \min_{\mathbf{s}\in\mathcal{S},\mathbf{d}\in\mathcal{D}(\mathbf{s})} \left\{(1-\gamma)\widetilde{Z}_0 + \sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}} \chi_{t,n}(\mathbf{s},\mathbf{d})\widetilde{X}_{t,n} + \sum_{t\in\mathcal{T},u\in\mathcal{U}} \upsilon_{t,u}(\mathbf{s},\mathbf{d})\widetilde{M}_{t,u}\right\}. \tag{A.5}$$

After termination of the column generation algorithm, the approximate optimal policy can be found by inserting the optimal values $(Z_0^*, X_{t,n}^*, M^*t, u)$ into the right-hand side of the optimality equations, and leaving out the terms that are independent of the decision. The approximate optimal policy is then determined as follows, for the formula does not differ from (5.32):

$$d^*(\mathbf{s}) \in \arg \min_{\mathbf{d}\in\mathcal{D}(\mathbf{s})} \left\{c(\mathbf{s},\mathbf{d}) + \gamma\left[\sum_{n\in\mathcal{N}}\sum_{t\in\mathcal{T}} x_{t,n}(\mathbf{s}')X_{t,n}^* + \sum_{t\in\mathcal{T}}\sum_{u\in\mathcal{U}} m_{t,u}(\mathbf{s}')M_{t,u}^*\right]\right\}, \quad \forall \mathbf{s}\in\mathcal{S}. \tag{A.6}$$