Aalto University
School of Science
Master's Programme in Mathematics and
Operations Research

Katja Metsäkoivu

A Capacity allocation method for appointment scheduling in public healthcare

Master's Thesis
Espoo, February 22, 2021

Supervisor:	Professor Antti Punkka, Aalto University
Advisor:	Juuso Rantala M.Sc. (Tech.)

Aalto University
School of Science
Master's Programme in Mathematics and Operations Research

ABSTRACT OF MASTER'S THESIS

Author:	Katja Metsäkoivu		
Title:	A Capacity allocation method for appointment scheduling in public healthcare		
Date:	February 22, 2021	**Pages:**	40
Major:	Systems and Operations Research	**Code:**	SCI3055
Supervisor:	Professor Antti Punkka		
Advisor:	Juuso Rantala M.Sc. (Tech.)		

This thesis develops an advance scheduling policy for a non-complex, but large scale healthcare service provider. A literature review of current advance scheduling and capacity allocation methods reveals a possibility to combine these two methodologies for this purpose. An integer linear program minimizing patient wait time and unused resource capacity is formulated for past demand data and known future capacities. A rolling horizon heuristic is utilized to solve the program over a full year and the results are used as a base for a capacity allocation structure. The performance of the capacity allocation scheduling is numerically tested in a case study of a public dental care provider. The simulation results show signs of overall improvement in patient wait times in accordance to the policy goals. Simple predictive details, such as expected rise in overall demand, could be included in the optimization model data to further improve the scheduling results.

Keywords:	healthcare, scheduling, optimization, capacity allocation, rolling horizon, simulation
Language:	English

Aalto-yliopisto
Perustieteiden korkeakoulu
Matematiikan ja operaatiotutkimuksen maisteriohjelma

DIPLOMITYÖN TIIVISTELMÄ

Tekijä:	Katja Metsäkoivu		
Työn nimi: Kapasiteetin allokointimenetelmä ajanvarausten aikataulutukseen julkisessa terveydenhuollossa			
Päiväys:	22. helmikuuta 2021	**Sivumäärä:**	40
Pääaine:	Systeemi- ja operaatiotutkimus	**Koodi:**	SCI3055
Valvoja:	Professori Antti Punkka		
Ohjaaja:	DI Juuso Rantala		

Tässä työssä kehitetään aikataulutusmenetelmä mittavalle terveydenhuollon palveluntarjoajalle. Nykyisiin menetelmiin tehty kirjallisuuskatsaus tuo esiin mahdollisuuden yhdistää etukäteisaikataulutuksen ja kapasiteetin allokoinnin lähestymistapoja tähän tarkoitukseen. Menneelle kysynnälle ja tunnetuille tuleville kapasiteeteille muodostetaan lineaarinen kokonaislukuoptimointitehtävä, joka minimoi asiakkaiden odotusaikaa sekä palveluntarjoajan käyttämätöntä kapasiteettia. Rullaavan horisontin heuristiikkaa käyttäen optimointitehtävä voidaan ratkaista kokonaisen vuoden ajalta. Tulosten perusteella muodostetaan rakenne kapasiteetin allokointiin. Allokointirakenteeseen perustuvaa aikataulutusta tarkastellaan numeerisesti julkisen hammashoidon tapaustutkimuksessa. Simuloinnin tulokset viittavaat yleiseen parannukseen potilaiden odotusajoissa mallin tavoitteiden mukaisesti. Tulokset voisivat parantua entisestään, jos optimoinnin dataan lisättäisiin ennustettuja yksityiskohtia, kuten odotettu yleinen kysynnän nousu.

Asiasanat:	terveydenhuolto, aikataulutus, optimointi, rullaava horisontti, simulointi
Kieli:	Englanti

Acknowledgements

Corona epidemic, self-isolation, and Master's thesis, what a way to spend a winter. I would like to thank my family for keeping me sane through this time and supporting me throughout my studies.

Thank you to my supervisor Prof. Antti Punkka for honest and encouraging feedback along the way, and to my advisor Juuso Ratala for finding this opportunity to finish my studies.

Espoo, February 22, 2021

Katja Metsäkoivu

Contents

Chapter 1

Introduction

Offering a constant customer experience in services where demand surpasses a restricted capacity requires special attention from scheduling. Environments like this are common in public services such as public healthcare. As a (nearly) free and ever relevant service, public healthcare will unceasingly have more demand than supply, resulting in queues and long wait times. Improvements in queue and resource management can help offer acceptable wait times for each customer group.

Healthcare customer groups, or *service types*, have different urgency levels and resource requirements. The urgency level of a service type defines its *service time goal*. A sudden toothache, for example, calls for a same day appointment, whereas a routine check up can wait up to several months. Some service time goals are imposed by the government as statutory treatment time guarantees (suom. hoitotakuu), while some goals are set by the *service providers* for their own operations. In Finland, for example, urgent care has to be accessible during the same day and non-urgent wait time should not exceed 3 months. Additionally, some service providers have set service time goals for semi-urgent service types (3 weeks) and follow-up appointments (1 month), to give a few examples. Healthcare resources vary from medical personnel to operating rooms and specialized machinery. The service type of an appointment determines which resources it requires to be served. A public healthcare provider's access to these resources depends on a government dictated budget. *Resource capacity* can be measured in time (e.g. working hours of medical personnel) or by number (e.g. number of operating rooms). The incoming demand of a healthcare provider is usually in the form of *appointment request* calls. During a call, the service type of the request is determined and an appointment time is scheduled. This type of scheduling, where scheduling decisions have to be made without any knowledge of later requests, is called *online advance scheduling* (Gupta and Denton 2008).

So how far in the future should a non-urgent appointment be scheduled to leave enough capacity for more urgent appointments requested later? This question has been answered with intricate mathematical algorithms in the case of a single resource (Patrick et al. 2008, Saure et al. 2012, Truong 2015, Erdelyi and Topaloglu 2009) or a single service type (Conforti et al. 2009, Liu et al. 2010). However, the case of a public healthcare provider in charge of hundreds of resources and dozens of service types is too large to be tackled with the existing methods.

This thesis develops a capacity allocation method that will enforce the service time goals of a large scale healthcare provider by guiding the advance scheduling process. To accomplish this task, the thesis first develops an extensive schedule optimization model. The solution of this optimization model is used as a base for dividing the service provider's available resources between its service types. The effectiveness of this resource allocation is numerically tested in a dental care example case and the resulting wait times are compared to the actual wait times achieved with their current scheduling policy. The comparison is based on metrics formulated for the objective function of the optimization model.

This thesis focuses on a case where the shape of the demand repeats at a known cycle or can otherwise be predicted in detail. The resource capacities of the next demand cycle are also assumed to be known. Patient wait times during the visit to the clinic are thoroughly studied in many papers (e.g. Erdogan and Denton 2013, Kuiper et al. 2015, Kemper et al. 2014, Castro and Petrovic 2012) and thus excluded from the scope of this thesis. The effects of appointment cancellations and patient "no shows" are only considered in passing, since they have inspired many studies of their own (e.g. Schuetz and Kolisch 2013, Wang and Gupta 2011) and, based on the case data used in this thesis, they do not have a significant impact in public dental care.

The rest of this thesis is divided into five chapters. Chapter 2 presents a literature review of appointment scheduling in healthcare. Chapter 3 introduces the theory behind mathematical methods used in the thesis. Chapter 4 provides a detailed description of the developed schedule optimization model and how the capacity allocation is derived from its results. Chapter 5 reviews an implementation of the capacity allocation method to a real life case in public dental care. Chapter 6 concludes the thesis and presents ideas for future development.

Chapter 2

Healthcare scheduling in literature

Bailey's (1952) study on patient wait time and medical personnel idle time is widely considered the first study where mathematical methods are used to model and improve healthcare operations. Since then methods of operations research have become important tools in the development of healthcare operations and they have inspired several studies from greatly varying perspectives. Historically, appointment scheduling methods have chiefly focused on the appointment time and the wait time of patients during their visit to the clinic, also known as *direct* wait time (Gupta and Denton 2008). Recently, there has been more studies focusing on the appointment day and the *indirect* wait time of accessing treatment. This literature review will focus on studies concerning indirect wait times, specifically previous works in *advance scheduling* and *capacity allocation*.

In advance scheduling systems, stochastically and dynamically arriving jobs are scheduled to future slots without any knowledge of requests arriving in later time slots (Parizi and Ghate 2016). This is very a common operating practice in healthcare; patients call a service line to book an appointment and are given one immediately during the call or by a call back service. The most common method used in development of advance scheduling policies for healthcare is a Markov decision process (MDP) combined with an approximation or heuristic for easier implementation.

Patrick et al. (2008) use MDP with approximate dynamic programming (ADP) to model the scheduling of diagnostic services for one type of resource, one appointment duration, and multiple urgency levels. Their model minimizes used overtime, appointment lateness, and the number of unscheduled appointments. The implementation of the policy is mostly possible with a set of simplified steps, but in some cases an integer program needs to be solved every day.

Saure et al. (2012) apply an extension to Patrick et al.'s (2008) work in a radiation therapy case with consideration of consecutive appointments and multiple appointment durations. The solution is stated as an approximate policy of four steps. Parizi and Ghate (2016) utilize a similar method combining MDP and ADP in the scheduling of elective surgeries with multiple service types and resource categories, but in implementation the developed mathematical program needs to be solved every day.

Liu et al. (2010) use MDP to form an index policy for appointment scheduling with a single service type. The method requires initial parameter estimations and some daily calculation. Feldman et al. (2014) extend this method with patient preference input.

Gocgun and Puterman (2014) use MDP and ADP to derive a scheduling policy in a chemotherapy case (one resource type, multiple service types) where different tolerance limits are set to the patient wait time. No scheduling cost is incurred for scheduling patients within their tolerance limits, linear cost for scheduling early or late. Implementation requires solving the developed ADP every scheduling day. Zhang et al. (2019) combine MDP, ADP, and stochastic programming in an elective surgery scheduling case where requests can be rejected and a full week of demand is collected before making scheduling decisions.

Besides MDP based methods there are some other advance scheduling approaches. Conforti et al. (2009) introduce an integer programming solution for scheduling consecutive radiation therapy appointments. A larger set of appointment requests is scheduled over the next weeks. Truong (2015) found an optimal dynamic programming solution for a case with one resource type and two service types (urgent and non-urgent).

The advance scheduling works described above all assume a service provider operating *offline* in the sense that they collect all appointment requests of a day (or other collection period) before making the scheduling decisions. This gives the scheduling policies more data to operate on, but it does not work for service providers who schedule their appointments *online*, before knowledge of any appointments requested later. Erdelyi and Topaloglu (2009) define statistically approximated protection level policies suited for online scheduling in a case with one resource type. And, more recently, Dai et al. (2020) detail two heuristics based on their MDP model, that could be used for immediate scheduling in a case with one resource type, one service type, and a moving booking window.

Another approach used to guide appointment scheduling is capacity allocation. It is a method of dividing available resources between different service types and it can be set in place far in advance while still offering purposeful structure for scheduling decisions. In the field of healthcare, capacity allocation is most frequently used in operating room scheduling and solved with linear programming.

Ozkarahan (2000) develops an integer linear program (ILP) for minimizing personnel idle time and over time during one operating day after an advance scheduling process has scheduled possible surgeries for that day. Ogulata and Erol (2003) tackle a general surgery scheduling case in a large hospital. Three hierarchical ILP formulations are used to decide weekly operating room schedules, maximize capacity utilization, balance allocation distribution, and minimize patient wait time. Beliën and Demeulemeester (2007) develop a mixed integer linear program (MILP) based heuristics to minimize expected total bed shortage with stochastic urgent cases and surgery duration. Mannino et al. (2012) tackle another cyclic master surgery schedule with a MILP formulation minimizing overtime and balancing patient queue lengths. Holte and Mannino (2013) utilize robust optimization to find a policy that minimizes queues in the worst demand scenario. Tang and Wang (2015) take a robust optimization model to an elective surgery case to minimize worst case revenue loss.

This thesis will use the notation and formulation of capacity allocation ILP methods to form an online advance scheduling policy for a large scale service provider.

Chapter 3

Methods

3.1 Linear Programming

Linear programming (LP) is a subcategory of mathematical programming, or mathematical optimization, which is used to determine the best numerical outcome of a mathematical model. All linear programs can be described in three lines as presented in (3.1) (Bertsimas and Tsitsiklis 1997):

$$
\begin{array}{rrcl}
\min & \mathbf{c}^T & & \mathbf{x} \\
s.t. & \mathbf{Ax} & \leq & \mathbf{b} \\
 & \mathbf{x} & \geq & 0 \ .
\end{array}
\tag{3.1}
$$

In the formulation above a linear objective function $\mathbf{c}^T\mathbf{x}$ is minimized subject to linear constraints $\mathbf{Ax} \leq \mathbf{b}$ and $\mathbf{x} \geq 0$. Vector $\mathbf{x}$ is the vector of decision variables, while vector $\mathbf{b}$, vector $\mathbf{c}$ and matrix $\mathbf{A}$ are parameters with known values. In a basic linear program the decision variables are continuous. If any of the decision variables are defined for a set of integers the problem is categorized as a mixed integer linear programming (MILP). Integer linear programming (ILP) which is a special case of linear programming where all model variables are integers. The most commonly used integer variables are binaries, $x \in \{0, 1\}$, which can only take two different values, 0 or 1.

3.2 Rolling horizon

Rolling horizon is a solution heuristic for optimization models extending over long periods of time. The need for a solving heuristic can arise either from uncertainty of input data or from the large scale of the model (Bischi et al. 2019). In both cases the full time horizon is divided into multiple shorter intervals and each interval is solved separately to find a sub-optimal (or optimal) approximation of the full horizon result. Figure 3.1 shows an example of commonly used rolling horizon structure. In the case of uncertain input data, each interval will have an active period to be optimized and a prediction period with predictive data to be considered in the optimization. In the case of a large scale model, both active and prediction period are optimized, but only the active period results are locked as final. Additionally, some information from past intervals can be considered to keep the result coherent in both cases. To give a few examples: in a production planning case, future demand forecasts are considered in the prediction horizon in order to anticipate production needs (Silvente et al. 2015); in an operational scheduling case, the past values of a yearly-basis performance constraint have to be taken into account in every interval as past information (Bischi et al. 2019).

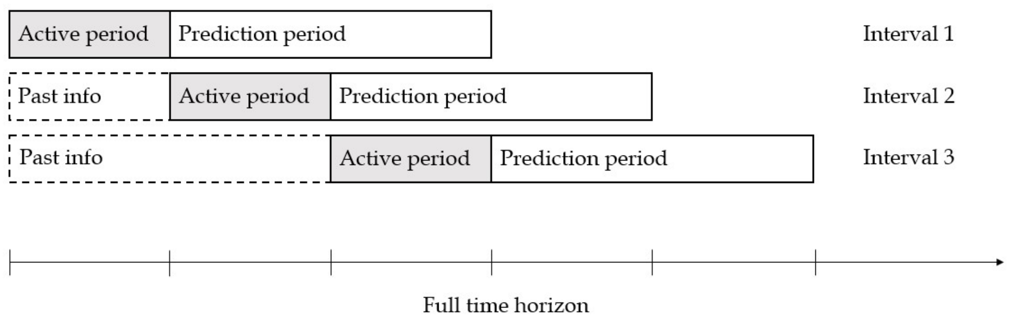

Figure 3.1: Example of rolling horizon structure.

The use of a heuristic will most likely lead to sub-optimality of the result and the scale of the inaccuracy might be difficult to measure. While the use of a heuristic cannot always be avoided, this should be addressed when considering the results.

3.3 Discrete-event simulation

Simulation is used for numerical estimation of real-life systems. There are many subcategories of simulation, but the focus of this section is to introduce the basics of a dynamic and discrete simulation, also called discrete-event simulation. Dynamic simulation refers to systems that evolve over time and discrete dynamic simulation to models where the time is depicted in discrete intervals rather than continuously (Law and Kelton 2000).

Building blocks of discrete-event simulation models are system state, simulation clock, event list, statistical counters, initialization routine, event routine, and library routines (Law and Kelton 2000). A simple example is to study the queue of a customer service desk. The systems state includes variables needed to describe the current state of the system. In this case they would be the number of customers in the queue, number of service desks, and whether each service desk is currently occupied by a customer. A simulation clock keeps track of the current simulation time. The event list contains the arrival times of new customers to the queue. Statistical counters store information about the system performance. For example, queuing time of each customer and the number of customers served. The initialization routine of the simulation sets the system state for simulation time 0. At the opening of the service desk there might be no queue and two empty service desks. The event routine updates the system state when events occur. When a new customer arrives the event routine first checks if there is a queue. If there is no queue, the routine advances to check if any service desks are free. If there is a free service desk the customer is moved to occupy that desk. Library routines generate a service time for the customer from a given distribution and add a "leaving customer" event to the event list.

A simulation is stochastic if it includes any probabilistic components, such as the customer service time described above. Stochastic simulations produce a random output and should be run multiple times before drawing any conclusions. Static, or deterministic, simulations have no stochastic elements and their output is also predetermined by the event list and the initial system state.

There are two main approaches for advancing the discrete-event simulation clock: fixed-increment and next-event (Law and Kelton 2000). With fixed-increment approach the simulation clock is advanced by the same increment every step and events that occur during the same time slot appear to happen simultaneously. With next-event approach the simulation clock is advanced to the next event time and the increments can vary in length.

Chapter 4

A model for finding a good capacity allocation for appointment scheduling

Figure 4.1 presents a simplified overview of the capacity allocation method developed in this Chapter. Initially, the available capacities (Figure 4.1i.) and past appointment requests are known. Next, appointments requests of different service types (pattern) and different duration (size) are scheduled to the available capacities with an optimization model (Figure 4.1ii.). The optimization results are summarized to daily service type slots by combining the durations of same type appointments scheduled to the same resource (Figure 4.1iii.). When using the capacity allocation, a scheduler can simply book appointments to the first available slot allocated to the right service type (Figure 4.1iv.).

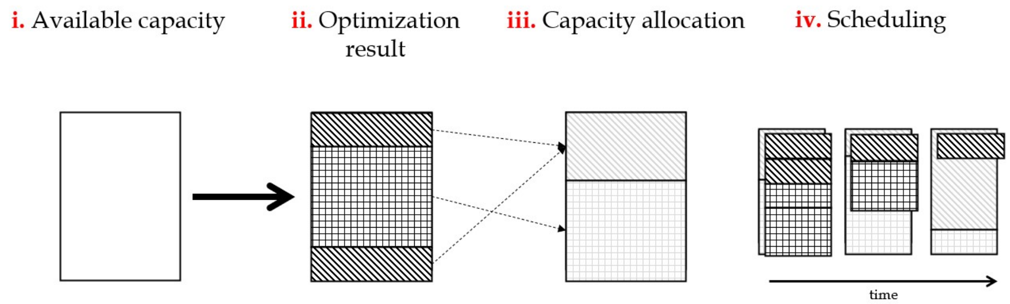

Figure 4.1: Developed capacity allocation method in simple terms

4.1 Assumptions

From the models' perspective all appointments scheduled for the same time slot happen at the same time. This is acceptable as long as the constraints make sure that resource requirements of appointments scheduled for a time slot do not exceed its resource capacities.

This thesis focuses on a healthcare network where resources in the near future (within a year) can be considered set, because their budget is determined well in advance. In the model the resource capacities are considered immutable, meaning no overtime or additional personnel are considered.

Based on the data of the dental care case of this thesis, the cancellation and patient "no show" rate in public dental care is very low. While appointment cancellations and "no shows" can be important to plan for in scheduling cases with high probabilities for them, they are not in the scope of this thesis.

Data analysis of the case service provider's requested appointments over past three years showed that weekly and even daily demand distributions have stayed near identical each year. This study will consider the appointments requested on a past year a fair approximation of future demand.

4.2 Schedule optimization model

The schedule optimization model schedules a set of appointments, $I = \{1, ..., n_I\}$, over a set of time slots, $J = \{1, ..., n_J\} \cup \{3000\}$, in such a way that overall appointment lateness and unused resource capacity are minimized. The time slot $j = 3000$ represents an unknown future time where appointments that do not fit to the optimization capacity can be scheduled. Each requested appointment i is allocated one appointment time that cannot be before the request time, $b_i \in J$. A binary decision variable $x_{ij} \in \{0, 1\}$ defines whether appointment i is booked for time slot j or not. Constraint (4.1) ensures that each appointment is booked exactly once, and constraint (4.2) makes sure that appointments cannot be booked before their request time:

$$\sum_{j \in J} x_{ij} = 1 \quad \forall\, i \in I \ , \tag{4.1}$$

$$\sum_{j=1}^{b_i - 1} x_{ij} = 0 \quad \forall\, i \in I \ . \tag{4.2}$$

Besides being scheduled to a certain time slot, the appointments are assigned to certain resources from the set of all resources, $R = \{1, ..., n_R\} \cup \{\phi\}$. The resource $r = \phi$ represents an additional resource that is only active in the additional time slot $j = 3000$. A binary decision variable $y_{ijr} \in \{0, 1\}$ defines whether appointment i is booked for time slot j and resource r. Each resource r has a capacity, $a_{jr} \geq 0$, for each time slot j. The additional resource, ϕ, has an unlimited capacity when $j = 3000$ and no capacity in the other time slots ($a_{3000,\phi} = \infty$ and $a_{j,\phi} = 0 \ \forall\ j \in J \backslash \{3000\}$). The sum of appointment durations, $d_i \geq 0$, booked for the resource cannot exceed this capacity, as stated in constraint (4.3):

$$\sum_{i \in I} y_{ijr} d_i \leq a_{jr} \quad \forall\, r \in R \backslash \{\phi\},\ \ j \in J \backslash \{3000\} \ . \tag{4.3}$$

Furthermore, the resource r assigned for appointment i has to be the right kind. The resources of the model are classified to n_C different resource categories, $C = \{1, ..., n_C\}$. A binary parameter $z_{ic} \in \{0, 1\}$ states whether appointment i requires a resource of category $c \in C$ or not, and binary parameter $k_{rc} \in \{0, 1\}$ whether resource r belongs to category c. The additional resource, ϕ, belongs to all resource categories ($k_{\phi,c} = 1 \ \forall\ c$). A resource can be part of multiple resource categories and it can serve appointments that require one or more of the categories it belongs to, as stated in constraint (4.4):

$$\sum_{r \in R} k_{rc} y_{ijr} \geq x_{ij} z_{ic} \quad \forall\, i \in I,\ \ j \in J,\ \ c \in C \ . \tag{4.4}$$

For example, the resource in Figure 4.2 belongs to categories $c1$ and $c3$. Appointments 1, 2, and 4 in the same figure can be scheduled to the resource but appointments 3 and 5 cannot.

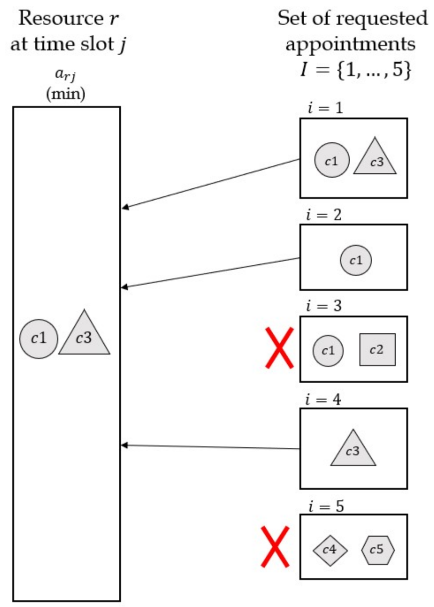

Figure 4.2: Example of a resource, its resource categories, and appointments it can serve.

In addition to satisfying the hard constraints of sufficient resource capacity and the right resource category, the model aims to serve each appointment within its service time goal. The model has a set of service types, $S = \{1, ..., n_S\}$, and each requested appointment i belongs to one service type $s_i \in S$. The service types have a service time goal $g_s > 0$ that determines within how many time slots from the request an appointment should be scheduled. Lateness of an appointment is defined as the difference between the scheduled time slot and the goal time slot, divided by the service time goal for comparability between different goals. The lateness of all appointments, $L(\mathbf{X})$, as defined in equation (4.5), is minimized in the objective function of the model:

$$L(\mathbf{X}) \;=\; \sum_{i \in I} \sum_{j=b_i+g_{s_i}+1}^{n_J} \frac{j-(b_i+g_{s_i})}{g_{s_i}} x_{ij} + \sum_{i \in I} M \; x_{i,3000} \quad , \qquad (4.5)$$

where $\mathbf{X}$ is a matrix of the decision variables x_{ij} and $M >> n_J$ is a large parameter representing lateness from booking an appointment to the additional time slot. Summing the above equation over all time slots would mean that scheduling two appointments (of the same service type) to the time slot matching their service time goal would be considered as valuable as serving one appointment three days early and one three days late. However, the performance of the service provider studied in this thesis is measured by the maximum service time rather than the average. It should be more valuable to serve both appointments in time than to serve one of them late, regardless of how early the other one would be served. In this model the lateness penalty is only affected for time slots later than the service time goal ($\{j \in J \mid j > b_i + g_{s_i}\}$). Otherwise the penalty is zero. For example, the lateness of an appointment ι requested at time slot 0 ($b_\iota = 0$) and with a service time goal of 30, the lateness is:

$$L(\iota) = \begin{cases} 0, & \text{if } 0 < j \leq 30 \\ \frac{j-30}{30}, & \text{if } 30 < j \leq n_J \\ M, & \text{if } j = 3000 \end{cases} . \tag{4.6}$$

While the capacity of resources cannot be exceeded (constraint (4.3)), the model aims to utilize as much of the available capacity as possible. Unused capacity of resource r is the difference between its availability a_{jr} and the sum of appointment durations d_i assigned to it. The model minimizes the sum of unused capacity over all resources and time slots denoted by:

$$U(\mathbf{Y}) = \sum_{r \in R \setminus \{\phi\}} \sum_{j \in J \setminus \{3000\}} (a_{jr} - \sum_{i \in I} y_{ijr} d_i) , \tag{4.7}$$

where $\mathbf{Y}$ is a matrix of the decision variables y_{ijr}. The unit of $U(\mathbf{Y})$ is minutes and each minute is as valuable as another so no scaling is necessary. Unused capacity of resource ϕ is not included in $U(\mathbf{Y})$.

4.3 Rolling horizon heuristic

The demand of the assessed healthcare provider is affected by weekends, holidays, holiday seasons, and the time of the year in general. These changes appear in a yearly cycle. Moreover, some appointments might already be booked as far as six months away. In order to take into account the yearly demand cycle and to get ahead of the partially booked resource capacities, the optimization horizon needs to cover a full year. With this time horizon and the numerous resources and appointments linked to it, it is likely that the problem will be computationally too large to be solved in one piece. In this case a rolling horizon heuristic will be utilized.

Besides the active period, each rolling horizon interval will consider an equal prediction period. While the future resource capacities cannot be adjusted based on the prediction, some prediction input can be used in planning which appointments should be scheduled to the active period. Any appointments requested in the active period but also scheduled for the additional time slot ($j = 3000$) are kept on to be scheduled in the later intervals.

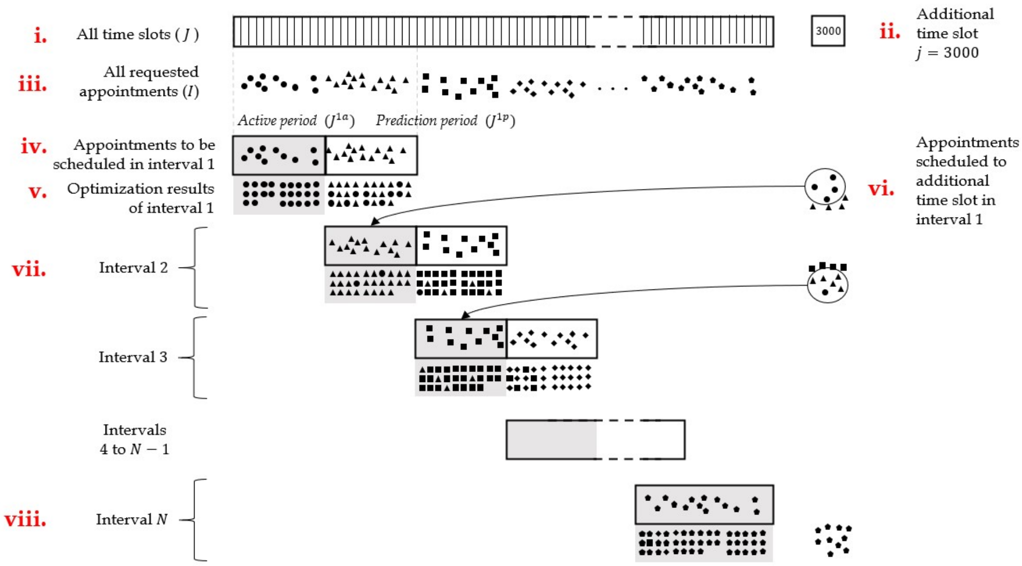

Figure 4.3: Utilized rolling horizon heuristic

Figure 4.3 visualizes this rolling horizon approach. The set of time slots, J (Figure 4.3i. and ii.), and the set of requested appointments, I (Figure 4.3iii.), are divided into N intervals. The first interval of the rolling horizon considers time slots in the active period J^{1a} and the prediction period J^{1p}, as well as the appointments requested during those time slots: $I^1 = \{i \in I \mid b_i \in J^1\}$ where $J^1 = J^{1a} \cup J^{1p}$ (Figure 4.3iv.). The additional time slot, 3000, is also included in each interval. The optimal schedule for appointments I^1 in time slots $J^1 \cup \{3000\}$ is solved and appointments scheduled for J^{1a} are locked (Figure 4.3v.). Appointments scheduled for $j = 3000$ are considered unscheduled and the ones requested during J^{1a} are added to be scheduled in the next interval (Figure 4.3vi.). Appointments requested during J^{1p} are included in the next interval no matter where they are scheduled to. In the next interval, $n+1$, time slots $J^{n+1} = J^{(n+1)a} \cup J^{(n+1)p}$ are considered, as well as the combination of the appointments requested during J^{n+1} and unscheduled appointments requested during J^{na}: $I^{n+1} = \{i \in I \mid b_i \in J^{n+1}\} \cup \{i \in I \mid b_i \in J^{na} \text{ and } x^n_{i,3000} = 1\}$ (Figure 4.3vii.). The rolling horizon intervals are continued until they cover the full scheduling horizon. The last interval, N, does not have a prediction period and all scheduled appointments are locked (4.3viii.).

4.4 Resulting capacity allocation

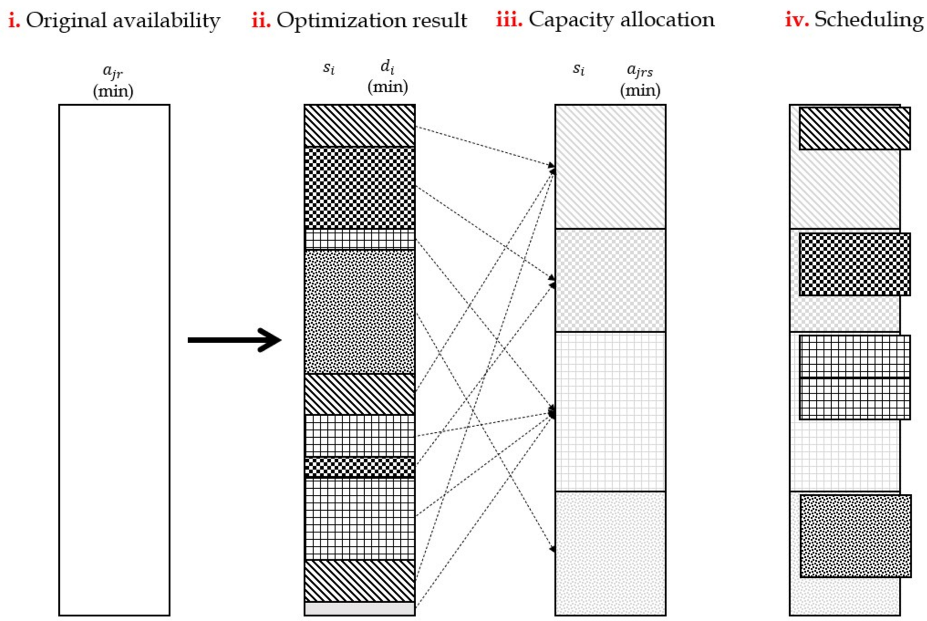

Figure 4.4: Scheduling policy

The optimized result requires perfect knowledge of future demand. While future demand can be expected to follow similar distribution as before, the exact request dates and amounts are not known for any future time. Thus, there is no need to save the optimization results variable by variable but rather as a summary of how much capacity is allocated to each service type in each time slot. At the beginning of the optimization each resource r has a capacity a_{jr} for each time slot j represented by the empty box in Figure

4.4i.. During the schedule optimization each resource capacity is filled with appointments and their durations. The capacity of the example resource in Figure 4.4 ii. has been filled with appointments of four different service types (patterns) with different durations (sizes). Based on the optimization result, the resource capacity, a_{jr}, is divided into service type specific slots, a_{jrs}, by summing the duration of appointments of the same same service type. Any unused capacity is allocated to the longest service type slot. An example of this capacity allocation is seen in Figure 4.4iii..

The service type slots, a_{jrs}, form the capacity allocation studied in this thesis. When using the capacity allocation in scheduling, appointments can only be scheduled to slots with their service type (Figure 4.4 iv.). Each service type should be using just the right amount of capacity to keep the customer wait times at a planned level.

Chapter 5

Case study

In this Chapter the developed capacity allocation method is implemented to a public dental care case. First, the specifics and data of the case are described and some minor adjustments are made in the optimization model. Next, the model complexity is addressed and the rolling horizon heuristic is validated. Finally, the solved capacity allocation is numerically tested and the simulation results are compared to the wait times achieved with the current scheduling system.

5.1 Case description and data

The case service provider's current scheduling system is based on capacity allocation. Each resource's daily capacity is divided between different service types and appointments can only be booked to slots with the matching service type. However, the capacity divisions are made based on vague goal percentages for each service type overall and not tightly monitored. Consequently, the actual capacities for each service type might be far from intended and the periodic fluctuation of demand is not taken into account.

The service provider's dental services are divided into sixteen (16) different service types and they are offered in twelve (12) different clinics. The service types can be divided into four (4) main categories: general care, urgent care, specialist care, and orthodontic care. Each clinic is equipped to offer services for service types in one or more main categories. Urgent care appointments all belong to the same service type and are served by one clinic, which only handles the urgent appointments. All the capacity of this clinic is already allocated to one service type. Keeping it in the model would only add complexity, with no added value. This leaves the case with eleven (11) clinics, fifteen (15) service types, and three (3) categories. The service provider employs three different types of professionals: dentists, dental hygienists, and dental nurses. A dentist is always paired up with a dental nurse and they have a designated operating room for the day. Dentists cannot work without these other two components so this combination will just be called "a dentist" (D). Similarly dental hygienists and independent dental nurses get a designated room for the day, and this combination of professional and room will be called "a dental hygienist" (DH) or "a dental nurse" (DN).

Professionals of the same category and in the same clinic have the capabilities to perform the same services. Therefore, it would not add value to the model to use each individual professional as a resource. Instead, this case work will use each combination of clinic and professional category (D, DH, DN) as resources and thus have a set of 33 resources. The daily capacity of a resource will depend on the number of professionals working each day. The earlier described service type categories (general, specialist, and orthodontic care) and professional categories (D, DH, DN) are all used as resource categories, giving the model a set of 6 resource categories. Each resource belongs to one professional category and one or more service type categories, depending on the clinic of the resource. Each appointment requires a specific professional category and a specific service type category.

Table 5.1: Set and parameter dimensions of the case data.

$n_I = 19\ 472$	Number of appointments, $i \in I$
$n_J = 364 + 1$	Number of time slots, $j \in J$
$n_R = 33 + 1$	Number of resources, $r \in R$
$n_C = 6$	Number of resource categories, $c \in C$
$n_S = 15$	Number of service types, $s \in S$
$5min \leq d_i \leq 480min$	Duration of appointment i
$330min \leq a_{jr} \leq 4485min$	Capacity of resource r in time slot j
$1d \leq g_s \leq 90d$	Service time goal of service type s
$(k_{rc}) \in \mathbf{R}^{34\times6}$	Resource r is of category c
$(z_{ic}) \in \mathbf{R}^{19472\times6}$	Appointment i requires a resource of category c
$(v_{jr}) \in \mathbf{R}^{366\times34}$	Resource r is active in time slot j

Set and parameter dimensions for the case are listed in Table 5.1. The matrix of the binary parameters k_{rc}, as well as example rows of z_{ic} and a_{jr} can be found from Appendix A. Each service type is associated with a service goal ranging from zero to ninety days. The service goal and service type category of each service type can be found listed in Table 5.2.

Table 5.2: Service time goal and service category of each service type.

s	service time goal (d)	service category
1	1	orthodontic
2	21	general
3	30	specialist
4	30	general
5	30	general
6	90	general
7	90	general
8	90	general
9	90	general
10	90	orthodontic
11	90	orthodontic
12	90	orthodontic
13	90	orthodontic
14	90	general
15	90	general

Appointment data collected from the service provider covers a time period from January 2017 to November 2019. The requested appointments from "2018" (October 2017 to September 2018) are used as the known demand for the optimization model and they are scheduled over the known resource capacities of "2019" (October 2018 to September 2019). The requested appointment of "2019" are used as an independent set for the numerical testing in Section 5.4. Transition from current scheduling system to the new one is taken into account by including appointments requested before October 2018 but scheduled for after it and keeping them in the slots they were given originally.

There were 180 034 requested appointments during "2018". Since the model already uses grouped resources, some grouping will be used for the appointments as well to keep the model size in check. The appointments requested on the same day, with the same service type and professional requirement, are grouped to reach up to 480 minutes of combined duration. This leaves the model with 19 472 requested appointments.

5.2 Adjustments in implementation

Since the professional(s) and the operating room is represented by a single resource, the resource categories required by each appointment need to be satisfied by one resource, not a combination of two or more. This is ensured with an additional constraint,

$$\sum_{r \in R} y_{ijr} = x_{ij} \qquad \forall\ i, j \quad , \tag{5.1}$$

stating that if appointment i is scheduled for time slot j ($x_{ij} = 1$) then the sum of resources r that the appointment i is scheduled to in this time slot j must also equal one ($\sum_{r \in R} y_{ijr} = 1$).

The service time goals for 1 and 90 days are imposed by the government, while the the other goals (21 and 30 days) are set by the service provider itself. An additional weight parameter is added to the lateness calculation (equation (4.5)) to represent a harder penalty from exceeding the 1 or 90 day goals:

$$w_{ij} = \begin{cases} 100, & \text{if } g_{s_i} = 1 \\ 50, & \text{if } j - b_i > 90 \\ 1, & \text{otherwise} \end{cases} \quad , \tag{5.2}$$

$$L(\mathbf{X}) \;=\; \sum_{i \in I} \sum_{j = b_i + g_{s_i} + 1}^{n_J} \frac{j - (b_i + g_{s_i})}{g_{s_i}} x_{ij}\, w_{ij} + \sum_{i \in I} 2000\ x_{i,3000} \quad . \tag{5.3}$$

Note that $M = 2000$ is chosen as the penalty for booking appointments to the additional time slot.

5.3 Solving the capacity allocation

With the parameter dimensions listed in Table 5.1, the full model has 248 million variables and 49 million constraints. While a section covering three months (18 million variables) can still be solved in a few hours, a section covering 6 months (70 million variables) results in a memory error. The rolling horizon heuristic is needed in order to solve the full problem.

To verify the performance of the rolling horizon method described in Section 4.4, the optimal result of a 3 month section is compared to the rolling horizon result of three one month intervals. The achieved objective value is only 5,3% higher in the case of three intervals, while the solve time is only one third. The rolling horizon heuristic can be assumed to be a fair approximation of the full horizon optimum.

The full model is solved in 12 intervals of 1 month active period and 1 month prediction period. The solving times of the intervals varied between 500 to 2 000 seconds.

Figure 5.1 shows a few months of the actual daily capacity allocation used in 2019 and the one derived from the optimization results. The actual capacity allocation clearly shows how the current standard of static goal percentages of each service type is realized for each day. While there is still a steady daily capacity of some service types in the new capacity allocation, there is is also a fair amount of periodic fluctuation. This should match the fluctuation of the incoming demand.

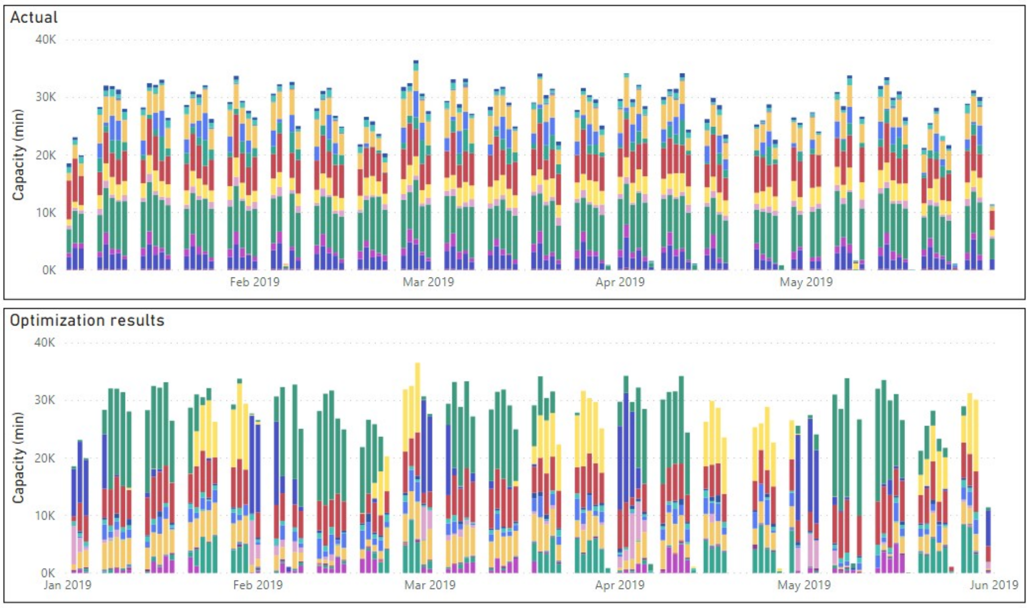

Figure 5.1: Actual capacity allocation in 2019 and the capacity allocation derived from the schedule optimization results with each color representing a different service type.

5.4 Simulation model

A deterministic discrete-event simulation model with fixed-increment advancement is used to examine wait times resulting from scheduling with the derived capacity allocation. The 2019 demand data is used as an independent random sample to evaluate the capacity allocation based on 2018 demand. Simulation results are compared to the actual 2019 wait times, lateness of appointments, and the number of scheduled appointments.

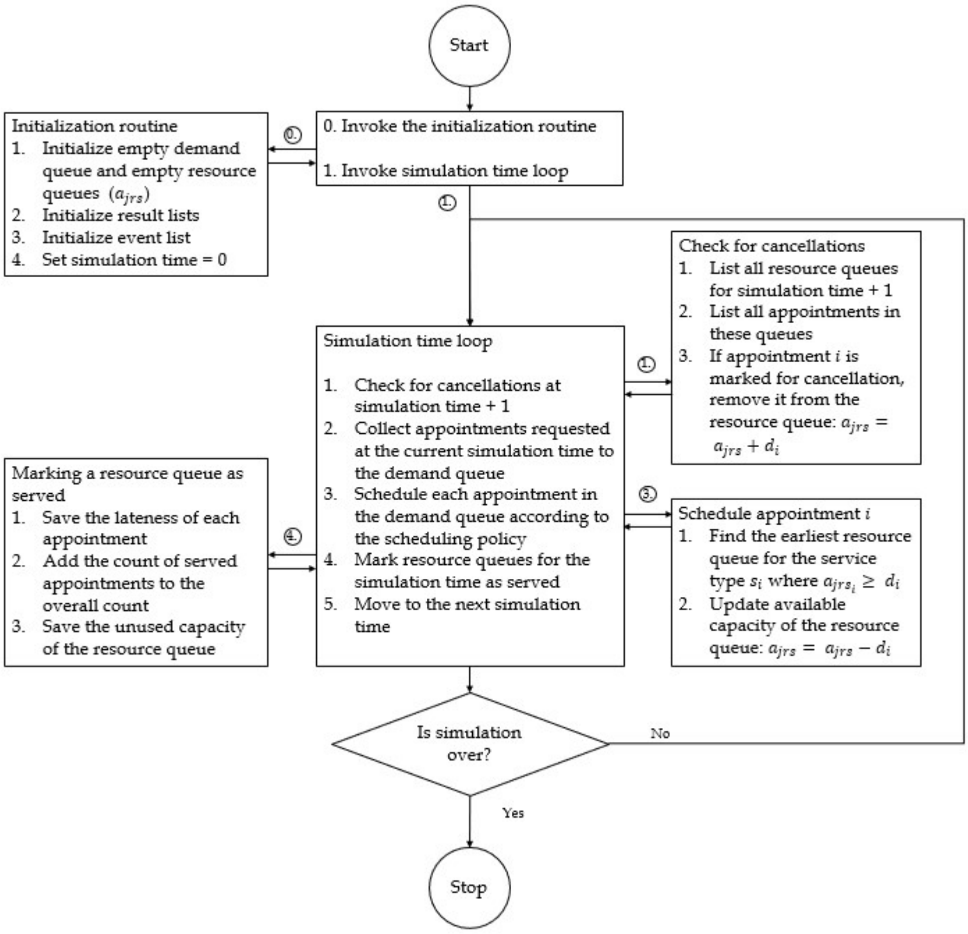

Figure 5.2: Simulation flow

Diagram of the simulation flow can be seen in Figure 5.2. First, an empty demand queue and empty scheduling queues are initialized. Each resource, $r \in R$, has a scheduling queue for each day, $j \in J$, and each service type, $s \in S$, with capacity a_{jrs} according to the capacity allocation. The additional resource, ϕ, in time slot $j = 3000$ has unlimited queue capacity and appointments that do not fit anywhere else can be scheduled there. Results lists for storing the lateness of appointments and the unused capacity of the resources are also initialized, and the appointments request data from 2019 is set as the event list. Finally, the simulation time is set to 0 (= 01/10/2018).

Appointment cancellations are handled in the first phase of the daily simulation loop. The model assumes a worst case scenario, where appointments are cancelled just before the clinic's cancellation policy of 24 hours. All the appointments scheduled 1 day later than the current simulation time are checked. If appointment i is marked to be cancelled in the request data it is removed from the scheduling queue by adding its duration back to the available capacity: $a_{jrs} = a_{jrs} + d_i$. A new appointment can be scheduled in its place in the next phases.

The second phase of the simulation time loop collects appointments requested at current simulation time to the demand queue. To imitate the advance scheduling of a public healthcare case, the simulation has only one demand queue operating in first-in-first-out (FIFO) basis. In the third phase of the loop, each appointment i in the demand queue is scheduled to the earliest scheduling queue that matches the service type s_i and has enough available capacity, $a_{jrs_i} \geq d_i$. When this queue is found, its available capacity is updated by reducing the duration of the scheduled appointment: $a_{jrs} = a_{jrs} - d_i$. The scheduler is concerned about good capacity utilization. If the new capacity of the first queue found would fall between 30 and 5 minutes ($5 < a_{jrs} < 30$), the scheduler checks if there are other matching queues during the same day whose new capacity would not fall in this interval. Additionally, if scheduled queues within three days of the simulation time have unused capacity, this capacity is released to any service types that can be served in that clinic. Excluded from this are scheduled queues for service types with service time goal of less than three days.

Once all of the appointments are scheduled to the earliest fitting slot (or to the additional time slot if nothing else is available) the loop moves to the fourth phase. Appointments scheduled for the current simulation time are marked as served. This means that the wait time (days) and lateness (as in equation (5.3)) of each appointment is calculated and the count of served appointments is added to the overall count.

Simulation time is moved one tick forward and if the time is still within the simulation horizon the loop is repeated from phase one. If the horizon is exceeded, the simulation ends.

5.5 Results

Tables 5.3 and 5.4 show a comparison of actual wait times in 2019 and the wait times achieved by using the optimized service type slots, respectively. On the simulation Table, results that are significantly better or worse are highlighted in green and red.

The overall lateness of the simulation is 27% better than in the baseline. Nearly all of the 95th percentile wait times, and even most of the maximum wait times of the simulation stay under the government appointed 90 day limit, whereas in the baseline there are many values over 100 and even 200 days. The wait time measures of service type 1 are also improved even if the service goal of 1 day is still not steadily achieved. A clear worst performer is service type 2 with the service time goal of 21 days. Its lateness and average service time are considerably higher than in the baseline. Overall, the policy goals are met and a general improvement in patient wait times is apparent.

Table 5.3: Baseline; actual booked appointments and wait times in 2019

service type s	Service time goal g (d)	Scheduled appointments	Lateness L	Average wait time (d)	95th precentile of wait time (d)	Maximum wait time (d)
1	1	1 492	679 700	5	28	83
2	21	14 405	12 526	28	78	152
3	30	3 241	55 124	48	105	229
4	30	31 076	215 802	40	91	235
5	30	2 875	22 364	38	92	232
6	90	12 069	16 463	67	103	182
7	90	17 279	20 532	55	105	249
8	90	5 204	206	24	89	162
9	90	6 594	839	35	87	161
10	90	20 652	12 414	46	101	228
11	90	98	44	48	91	120
12	90	231	9	49	97	98
13	90	920	1 488	69	118	128
14	90	4 151	752	35	75	177
15	90	2 453	151	33	70	127
Total		**122 740**	**1 038 415**			

Table 5.4: Simulation results

service type s	Service time goal g (d)	Scheduled appointments	Lateness L	Average wait time (d)	95th precentile of wait time (d)	Maximum wait time (d)
1	1	1 481	149 300	2	6	12
2	21	10 999	571 543	57	132	139
3	30	3 130	17 353	49	82	108
4	30	36 005	5 193	28	46	49
5	30	3 109	28	16	32	44
6	90	13 506	0	45	71	83
7	90	18 802	0	56	75	77
8	90	5 747	0	26	55	66
9	90	6 459	644	59	91	97
10	90	16 275	10 041	73	98	112
11	90	117	0	17	35	58
12	90	253	0	20	44	55
13	90	971	0	60	76	84
14	90	3 564	0	44	67	73
15	90	2 103	4 176	73	109	112
Total		**122 521**	**758 278**			

5.6 Sensitivity analysis

In this Section a few alternative compositions of the weight parameter w_{ij} are compared to see if and how the simulation results change. The weights used in the case study are labelled "$w^{90} = 50$":

$$w_{ij}^1 = \begin{cases} 100, & \text{if } g_{s_i} = 1 \\ 50, & \text{if } j - b_i > 90 \\ 1, & \text{otherwise} \end{cases} . \tag{5.4}$$

Their performance is compared to relaxing the penalty from exceeding 90 days, labelled "$w^{90} = 1$":

$$w_{ij}^2 = \begin{cases} 100, & \text{if } g_{s_i} = 1 \\ 1, & \text{otherwise} \end{cases} , \tag{5.5}$$

and to adding penalty to the worst performer, "$w^{21} = 50$":

$$w_{ij}^3 = \begin{cases} 100, & \text{if } g_{s_i} = 1 \\ 50, & \text{if } g_{s_i} = 21 \\ 1, & \text{otherwise} \end{cases} . \tag{5.6}$$

Figure 5.5 presents a summary of the baseline and different simulation results. The appointments are grouped by their service time goal. The lateness of the baseline, which changes with w_{ij}, is set to be 100 percent and the lateness of each simulation is stated in relation to the baseline value.

Relaxing the penalty from exceeding 90 days of wait time results in a much smaller overall lateness but there are only small differences in the other measures. Even the added weight to appointments with 21 day service goal does not better the performance on their part and brings the overall lateness to the same level as the baseline. A closer look into the case data reveals that while the overall demand for appointments with 21 day service goal stays the same in 2018 and 2019, the shape of the 2019 demand is quite different. In 2019 this specific service goal is demanded in spikes rather than evenly all the time. Therefore, even if the weighted optimization reserves more capacity for this service type it cannot be utilized efficiently in the simulation. This is a draw back for the solution, but not unexpected with the amount of weight put on the shape of the past demand cycles.

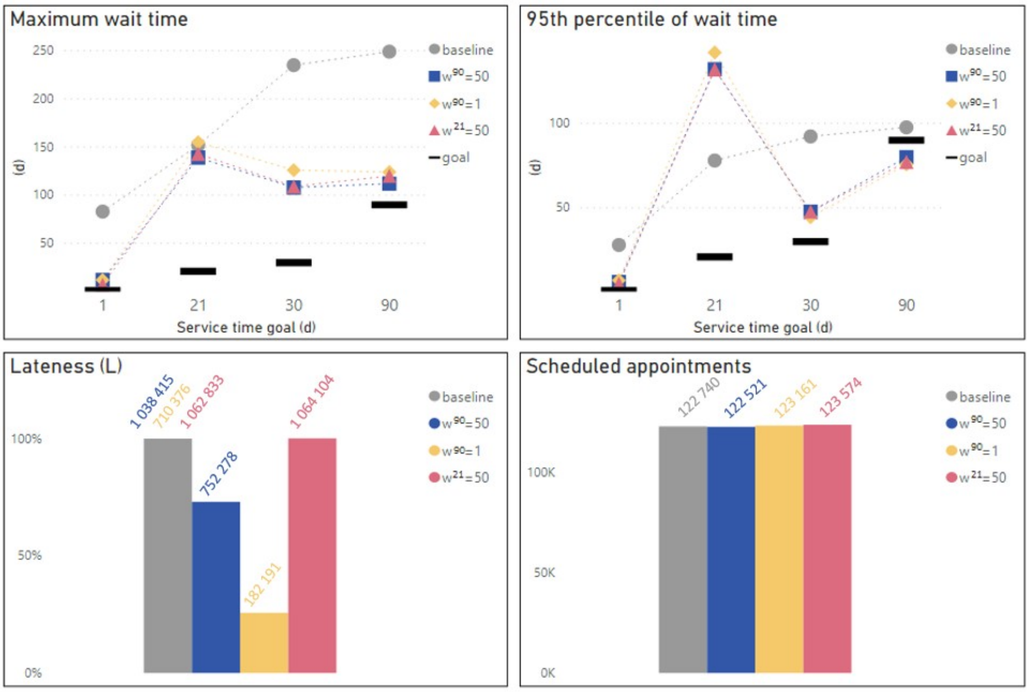

Table 5.5: Scheduled appointments, lateness, 95th percentile of wait times and maximum wait times of baseline and simulations with different w_{ij}.

Chapter 6

Conclusion

The simulated advance scheduling gives promising results about the value of the developed model. While the number of served patients is not greatly affected, the overall lateness and the maximum wait times are lower, which means a more constant customer experience is offered. Sensitivity analysis reveals a weakness when the shape of the demand differs greatly from past sample, but this type of notable changes will be troublesome with any scheduling policy.

A large part of the usefulness of the method comes from its ease of implementation. There is no need for specific parameter evaluations or calculation, or predictive demand analysis. A visual confirmation of yearly (or other cycle) demand similarities already leads to improvements in overall wait times. The model works with simple inputs of past demand and planned future resource capacities. While the solving of the optimization model is somewhat time-consuming, it needs to be solved only once a year.

In future development of the method, inclusion of predictive details in the optimization phase could benefit the performance of the scheduling. If a general increase in demand, or a planned ramp up of a certain service type is expected, they could be implemented to the optimization demand set and thus flexibility towards them would be included in the capacity allocation. Many different shapes of the objective function could also be studied depending on the preferences of the service provider. Small reward could be added to the $j < b_i + g_{s_i}$ side of the lateness function, or each service type or service goal might have their own weight in the objective function.

The study and improvement of indirect wait times was a long avoided subject in the in healthcare scheduling studies. While recent studies offer some answers to the advance scheduling problem, there are no suitable methods for a large service provider utilizing online advance scheduling. This thesis contributes to fill this void by using a simplified but large scale version of operating room capacity allocation ILP problem.

Bibliography

N. Bailey. A Study of Queues and Appointment Systems in Hospital Out-Patient Departments, with Special Reference to Waiting-Times. *Journal of the Royal Statistical Society*, 14(2):185–199, 1952. URL `https://www.jstor.org/stable/2983867`.

J. Beliën and E. Demeulemeester. Building cyclic master surgery schedules with leveled resulting bed occupancy. *European Journal of Operational Research*, 176(2):1185–1204, 2007. URL `https://doi.org/10.1016/j.ejor.2005.06.063`.

D. Bertsimas and J. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, 1st edition, 1997. ISBN 886529191.

A. Bischi, L. Taccari, E. Martelli, E. Amaldi, G. Manzolini, P. Silva, S. Campanari, and E. Macchi. A rolling-horizon optimization algorithm for the long term operational scheduling of cogeneration systems. *Energy*, 184:73–90, October 2019. URL `https://doi.org/10.1016/j.energy.2017.12.022`.

E. Castro and S. Petrovic. Combined mathematical programming and heuristics for a radiotherapy pre-treatment scheduling problem. *Journal of Scheduling*, 15(3):333–346, 2012.

D. Conforti, F. Guerriero, R. Guido, and M. Veltri. An optimal decision-making approachfor the management of radiotherapy patients. *OR Spectrum*, 33:123–148, April 2009. doi: 10.1007/s00291-009-0170-y.

J. Dai, N. Geng, and X. Xie. Dynamic advance scheduling of outpatient appointments in a moving booking window. *European Journal of Operational Research*, 2020. URL `https://doi.org/10.1016/j.ejor.2020.11.030`.

A. Erdelyi and H. Topaloglu. Computing protection level policies for dynamic capacity allocation problems by using stochastic approximation methods. *IIE Transactions*, 41(6):498–510, March 2009. doi: 10.1080/07408170802706543. URL https://doi.org/10.1080/07408170802706543.

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

J. Feldman, N. Liu, H. Topaloglu, and S. Ziya. Appointment Scheduling Under Patient Preferenceand No-Show Behavior. *Operations Research*, 62(4):794–811, 2014. URL http://dx.doi.org/10.1287/opre.2014.1286.

Y. Gocgun and M. Puterman. Dynamic scheduling with due dates and time windows: an application to chemotherapy patient appointment booking. *Health Care Manage Sci*, 17:60–76, 2014. doi: 10.1007/s10729-013-9253-z.

D. Gupta and B. Denton. Appointment scheduling in health care: Challenges and opportunities. *IIE Transactions*, 40(9):800–819, July 2008. doi: 10.1080/07408170802165880. URL https://doi.org/10.1080/07408170802165880.

M. Holte and C. Mannino. The implementor/adversary algorithm for the cyclic and robust scheduling problem in health-care. *European Journal of Operational Research*, 226(3):551–559, 2013. URL https://doi.org/10.1016/j.ejor.2012.10.029.

B. Kemper, C. Klaassen, and M. Mandjes. Optimized appointment scheduling. *European Journal of Operational Research*, 239:243–255, 2014.

A. Kuiper, B. Kemper, and M. Mandjes. A computational approach to optimized appointment scheduling. *Queueing Systems*, 79:5–36, 2015.

A. Law and W. Kelton. *Simulation modeling and analysis*. McGraw-Hill Higher Education, 3rd edition, 2000. ISBN 0070592926.

N. Liu, S. Ziya, and V. Kulkarni. Dynamic Scheduling of Outpatient Appointments UnderPatient No-Shows and Cancellations. *Manufacturing Service Operations Management*, 12(2):347–364, 2010. doi: 10.1287/msom.1090.0272.

C. Mannino, E. Nilssen, and T. Nordlander. A pattern based, robust approach to cyclic master surgery scheduling. *Journal of Scheduling*, 15:553–563, 2012.

S. Ogulata and R. Erol. A Hierarchical Multiple Criteria Mathematical Programming Approach for Scheduling General Surgery Operations in Large Hospitals. *Journal of Medical Systems*, 27(3):259–270, 2003.

I. Ozkarahan. Allocation of Surgeries to Operating Rooms by Goal Programing. *Journal of Medical Systems*, 24(6):339–378, 2000.

M. Parizi and A. Ghate. Multi-class, multi-resource advance scheduling with no-shows, cancellations and overbooking. *Computers Operations Research*, 67:90–101, March 2016. doi: 10.1016/j.cor.2015.09.004. URL https://doi.org/10.1016/j.cor.2015.09.004.

J. Patrick, M. Puterman, and M. Queyranne. Dynamic Multipriority Patient Scheduling for a Diagnostic Resource. *Operations Research*, 56(6):1507–1525, April 2008. doi: 10.1287/opre.1080.0590. URL https://doi.org/10.1287/opre.1080.0590.

A. Saure, J. Patrick, S. Tyldesley, and M. Puterman. Dynamic multi-appointment patient scheduling for radiation therapy. *European Journal of Operational Research*, 223(2):573–584, December 2012. doi: 10.1016/j.ejor.2012.06.046. URL https://doi.org/10.1016/j.ejor.2012.06.046.

H. Schuetz and R. Kolisch. Capacity allocation for demand of different customer-product-combinations with cancellations, no-shows, and overbooking when there is a sequential delivery of service. *Annals of Operations Research*, 206:401–423, 2013.

J. Silvente, G. Kopanos, E. Pistikopoulos, and A. Espuña. A rolling horizon optimization framework for the simultaneous energy supply and demand planning in microgrids. *Applied Energy*, 155:485–501, October 2015. URL https://doi.org/10.1016/j.apenergy.2015.05.090.

J. Tang and Y. Wang. An adjustable robust optimisation method for elective and emergency surgery capacity allocation with demand uncertainty. *International Journal of Production Research*, 53(24):7317–7328, 2015. URL https://doi.org/10.1080/00207543.2015.1056318.

V. Truong. Optimal Advance Scheduling. *Management Science*, 61(7):1584–1597, February 2015. doi: 10.1287/mnsc.2014.2067. URL https://doi.org/10.1287/mnsc.2014.2067.

W. Wang and D. Gupta. Adaptive appointment systems with patient preferences. *Manufacturing Service Operations Management*, 13(3):373–389, 2011.

J. Zhang, M. Dridi, and A. El Moudni. A two-level optimization model for elective surgery scheduling with downstream capacity constraints. *European Journal of Operational Research*, 276(2):602–613, 2019. URL https://doi.org/10.1016/j.ejor.2019.01.036.

Appendix A

Binary parameters in case work

Table A.1: Examples of binary parameter z_{ic}'s values.

i / c	D	DH	DN	general	specialist	orthodontic
1	0	1	0	1	0	0
2	1	0	0	1	0	0
3	1	0	0	1	0	0
4	1	0	0	1	0	0
5	1	0	0	1	0	0
6	1	0	0	1	0	0
7	0	0	1	1	0	0
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
19 470	0	1	0	1	0	0
19 471	1	0	0	0	0	1
19 472	1	0	0	0	0	1

Table A.2: Binary parameter k_{rc}'s values.

r / c	D	DH	DN	general	specialist	orthodontic
clinic 1, D	1	0	0	1	0	0
clinic 1, DH	0	1	0	1	0	0
clinic 1, DN	0	0	1	1	0	0
clinic 2, D	1	0	0	1	0	0
clinic 2, DH	0	1	0	1	0	0
clinic 2, DN	0	0	1	1	0	0
clinic 3, D	1	0	0	1	1	0
clinic 3, DH	0	1	0	1	1	0
clinic 3, DN	0	0	1	1	1	0
clinic 4, D	1	0	0	1	0	0
clinic 4, DH	0	1	0	1	0	0
clinic 4, DN	0	0	1	1	0	0
clinic 5, D	1	0	0	1	0	0
clinic 5, DH	0	1	0	1	0	0
clinic 5, DN	0	0	1	1	0	0
clinic 6, D	1	0	0	1	0	0
clinic 6, DH	0	1	0	1	0	0
clinic 6, DN	0	0	1	1	0	0
clinic 7, D	1	0	0	1	0	1
clinic 7, DH	0	1	0	1	0	1
clinic 7, DN	0	0	1	1	0	1
clinic 8, D	1	0	0	1	0	1
clinic 8, DH	0	1	0	1	0	1
clinic 8, DN	0	0	1	1	0	1
clinic 9, D	1	0	0	1	0	0
clinic 9, DH	0	1	0	1	0	0
clinic 9, DN	0	0	1	1	0	0
clinic 10, D	1	0	0	1	0	0
clinic 10, DH	0	1	0	1	0	0
clinic 10, DN	0	0	1	1	0	0
clinic 11, D	1	0	0	1	0	0
clinic 11, DH	0	1	0	1	0	0
clinic 11, DN	0	0	1	1	0	0
ϕ	1	1	1	1	1	1

Table A.3: Examples of binary parameter a_{jr}'s values.

j / r	clinic 1 D	clinic 1 DH	. . .	clinic 11 DH	clinic 11 DN	ϕ
1	60	0	. . .	0	0	0
2	315	75	. . .	30	30	0
3	195	75	. . .	210	105	0
4	180	0	. . .	180	120	0
5	135	135	. . .	105	30	0
6	0	0	. . .	0	0	0
.	.	.	. . .	.	.	.
.	.	.	. . .	.	.	.
.	.	.	. . .	.	.	.
359	1230	660	. . .	2010	1020	0
360	1635	210	. . .	1935	690	0
361	795	660	. . .	1485	960	0
362	1245	630	. . .	1800	630	0
363	0	0	. . .	0	0	0
364	0	0	. . .	0	0	0
3000	0	0	. . .	0	0	1