Data Informed Planning in Healthcare

Practical Models for Rostering and Scheduling under Uncertainty and Risk

Thomas Elliott Adams

A thesis submitted in fulfilment of the requirements for the degree of
Doctor of Philosophy in Operations Research,
The University of Auckland, 2019.

Abstract

Planning and scheduling of resources is a crucial process within healthcare organisations. Careful planning allows limited resources, such as staff and operating rooms, to be used efficiently. However this is often a difficult and time consuming process, and not all elements are known with certainty. Mathematical modelling can be applied to assist in rostering and scheduling. Further, historical data collected by the healthcare organisations enables modelling of the uncertainty in the planning processes.

This thesis considers two planning processes: rostering physicians; and scheduling surgeries; and develops mathematical models for both situations. A model for rostering physicians is proposed which incorporates the admission and discharge of patients, and uses patient pathways to connect multiple rosters. In addition an objective function for scheduling surgeries is formed that utilises a risk measure applied to the due dates of operations, and new methods for estimating the probability that surgical sessions run overtime are developed. Further, both models are applied to case studies in a practical setting. The rostering model has been used to create a roster that was implemented in a hospital department, while efforts are ongoing to promote the adoption of the surgery scheduling methods in several surgical services.

Both uncertainty and risk are investigated in the two models. In the rostering model the uncertainty considered is in the number of patients the physicians are caring for, whereas in the surgery scheduling model the uncertainty lies in the duration of the operations performed. In the rostering model risk directly relating to the uncertainty modelled, which is in the admissions and discharges of patients, is avoided. On the other hand, in the surgery scheduling model the risk of operations being performed much after they are due is avoided, rather than risk associated with the durations of operations.

This research demonstrates the incorporation of historical data and the related uncertainty into rostering and scheduling approaches in order to improve the quality of the rosters and schedules produced.

Acknowledgements

I would first like to thank my supervisors, Mike O'Sullivan and Cameron Walker, their advice and guidance made this work not only possible but a pleasure. I would also like to thank my parents Marion and Matthew, my sister Henrietta my brother Jack, and my girlfriend Nina, for their support and encouragement. Finally I must thank all those who made this research possible through prior efforts and collaboration. Thanks to Paul Muir, Jonathan Christiansen, and Ellen Sumpter at Waitemata District Health Board for providing data and expertise on the rosters at Waitakere Hospital. Thanks to Nicolas Szecket, Rupert Handy, Andrew Peterson, Tim Denison, and Art Nahill at Auckland District Health Board for presenting the difficulties they face rostering physicians to my supervisors. Thanks to Hanieh Sanei, and Amelia White for their previous work describing and modelling the rosters at Auckland City Hospital. Thanks to Precision Driven Health for providing an opportunity to work on surgical scheduling, and thanks to Kevin Ross, Kelly Atkinson, and in particular Luke Boyle for their efforts coordinating my work with Counties Manukau District Health Board. Thanks to Bill Lewis and Keming Wang at Counties Manukau District Health Board for providing data and guidance on scheduling surgeries.

Contents

List of Figures

List of Tables

List of Physician Rostering Terms

O	Set of the groups of physicians to be rostered separately
R_o	Set of the physicians in roster group o, the set of integers from 1 to $\lvert R_o \rvert$
W_o	Set of the weeks in the roster of roster group o, the set of integers from 1 to $\lvert W_o \rvert$
S_o	Set of the possible shifts for the physicians in roster group o
L	Set of the groups towards which the workloads of the physicians contribute
$V_{o,r,l}$	A binary indicator that physician r in roster group o contributes their patients to workload group l
C	Set of all days in the planning horizon
D	Set of the days of the week
P	Set of the parts of the day
K	Set of classes of patient
$A_{k,c,p}$	The number of patients admitted on day c, in part p, of class k
$U_{k,c,p,o,s}$	Proportion of admissions of class k on day c, part p, that shift s of roster group o adds to their workload
$Q_{k,c,p}$	Proportion of patients discharged on day c, in part p, of class k, after taking into account patients that were transfered
$T_{l,k,c,p}$	Proportion of patients transferred from workload group l on day c, in part p, of class k

$F_o \subseteq S_o \times W_o \times D$ Set of shift, week, day tuples used to index which shifts must be performed, and when, for roster group o

$B_o \subseteq S_o \times W_o \times D$ Set of shift, week, day tuples for roster group o, used to indicate the shifts that are banned on each day of each week

$\mathcal{L}_{o,s,w,d} \subseteq S_o \times W_o \times D$ Set of shift, week, day tuples which includes shifts that are linked to future shifts, such that if registrar r of roster group o works shift s in week w, day d and $(s, w, d) \in \mathcal{L}_o$ then they must also work shifts $s_1, \ldots, s_n$ in weeks $w_1, \ldots, w_n$ on days $d_1, \ldots, d_n$, and must not work these shifts otherwise

$G_{o,s,w,d} \subseteq S_o \times W_o \times D$ Set of shift, week, day tuples which includes shifts that are linked to future shifts, such that if registrar r of roster group o works shift s in week w, day d and $(s, w, d) \in G_o$ then they must also work shifts $s_1, \ldots, s_n$ in weeks $w_1, \ldots, w_n$ on days $d_1, \ldots, d_n$, but may work these shifts if they do not

$E_{o,s,w,d} \subseteq S_o \times W_o \times D$ Set of shift, week, day tuples which includes shifts that are linked to future shifts, such that if registrar r of roster group o works shift s in week w, day d and $(s, w, d) \in G_o$ then they must also work shifts $s_1, \ldots, s_n$ in weeks $w_1, \ldots, w_n$ on days $d_1, \ldots, d_n$, but may work these shifts if they do not

$\mathcal{R}_o \subseteq S_o \times D$ Set of shift, day pairs used to index the number of shifts that need to be performed by roster group o

$i_{o,s,d}$ The number of registrars in roster group o that must be assigned shift s on day d

$y_{o,w,d}$ The number of shifts that registrars in roster group o can be assigned on day d in week w

$f_1 : O \times R_o \times C \longrightarrow W_o$ A function that returns the week in a given roster group's cycle that a registrar is working, given the count day

$f_2 : C \longrightarrow D$ A function that returns the day of the week given the count day

$f_3 : \mathbb{Z} \times O \longrightarrow W_o$ A function that returns the week in a given roster group's cycle given an integer number of weeks from the start of the planning horizon

$j : K \times C \times P \rightarrow \mathcal{P}(K \times C \times P)$	A function that describes how a patients' class changes over time. Returns the set of class, week, part tuples that transition directly to the given class, week, part tuple
$x_{o,s,w,d}$	Whether shift s is performed on week w, day d, by roster group o (1 if it is, 0 otherwise)
$N_{l,k,c,p}$	Number of patients of class k, workload group l is attending to at the end of part p on count day c
$M^{+}_{c,p}$	An upper bound on the workload of the workload group with the largest workload in part p on count day c
$M^{-}_{c,p}$	A lower bound on the workload of the workload group with the smallest workload in part p on count day c
$\bar{M}$	The largest difference in workloads between the workload groups over the entire planning horizon
Z	Set of scenarios for admissions and discharges
β	The percentage of scenarios which are excluded from the conditional expectation when calcualting conditional value at risk, starting with the best scenarios and going to the worst
α	The β percentile
$\gamma^{(z)}$	If the largest difference in workloads in the scenario z is above α then $\gamma^{(z)}$ is equal to the amount that the largest difference in workloads is greater than α, otherwise it is 0

List of Surgery Scheduling Terms

O	Set of operations to be performed
d_o	Due date of operation o
μ_o	Mean duration of operation o
σ_o	Standard deviation of the duration of operation o
S	Set of surgical sessions
b_s	The start time of session s
l_s	The length of session s
t	The turn around time between operations
$c_{o,s}$	The cost of performing operation o in session s
λ	The exponent applied to the lateness of an operation in the cost function
M_s	The sum of the expected durations of the operations (and the turn around times between the operations) assigned to a session s
V_s	The sum of the variances of durations of the operations assigned to a session s
L_s	A random variable that represents the time taken to perform all of the operations in the session, s, given that the durations of each of the operations are themselves random variables
υ	The maximum allowed probability that each session goes overtime, in other words $\mathcal{P}(L_s \leq l_s) \leq \upsilon$
$x_{o,s}$	Whether operation o is performed in session s (1 if it is, 0 otherwise)

$f(V_s, l_s, \upsilon)$ A function that takes the assigned session variance, session length, and allowed overtime probability and return the allowed assigned session duration

β The percentage of operations which are excluded from the conditional expectation when calcualting conditional value at risk, starting with the operations with the lowest cost and going to the highest

α The β percentile

γ_o If cost of performing operation o is above α then γ_o is equal to the amount that the cost is greater than α, otherwise it is 0

$\bar{\mu}$ The mean duration of all the operations

$\bar{\sigma}$ The standard deviation of the duration of all the operations

$G_{L_s|M_s}$ The empirical conditional cumulative distribution function of realised session duration L_s, given assigned session duration M_s

$h_{l_s}(M_s)$ The probability that a session of length l_s goes overtime given that it has an assigned session duration M_s

List of Acronyms

MoH	Ministry of Health
DHB	District Health Board
ADHB	Auckland District Health Board
CMDHB	Counties Manukau District Health Board
WDHB	Waitemata District Health Board
ACH	Auckland City Hospital
MSC	Manukau Surgery Centre
MMH	Middlemore Hospital
NSH	North Shore Hospital
WTH	Waitakere Hospital
ADCU	assessment, diagnostic, and cardiology unit
ED	emergency department
GM	general medicine
GP	general practitioner
OR	operating room
MSS	master surgery schedule
LP	linear programme

MIP	mixed integer programme
RMP	restricted master problem
MP	master problem
SP	sub-problem
ACCPM	analytic centre cutting plane method
CVaR	conditional value at risk
SAA	sample average approximation
GND	grand normal distribution
SND	session normal distribution
DED	duration empirical distribution
DVED	duration variance empirical distribution
GLM	generalised linear model

THE UNIVERSITY OF AUCKLAND
Te Whare Wānanga o Tāmaki Makaurau
NEW ZEALAND

School of Graduate Studies
AskAuckland Central
Alfred Nathan House
The University of Auckland
Tel: +64 9 373 7599 ext 81321
Email: postgradinfo@auckland.ac.nz

Co-Authorship Form

This form is to accompany the submission of any PhD that contains published or unpublished co-authored work. **Please include one copy of this form for each co-authored work**. Completed forms should be included in all copies of your thesis submitted for examination and library deposit (including digital deposit), following your thesis Acknowledgements. Co-authored works may be included in a thesis if the candidate has written all or the majority of the text and had their contribution confirmed by all co-authors as not less than 65%.

Please indicate the chapter/section/pages of this thesis that are extracted from a co-authored work and give the title and publication details or details of submission of the co-authored work. Chapter 3 of thesis. Title of publication: Physician Rostering for Workload Balance Journal: Operations Research for Health Care	
Nature of contribution by PhD candidate	Development of mathematical model and solution technique, conception and form of article
Extent of contribution by PhD candidate (%)	80%

CO-AUTHORS

Name	Nature of Contribution
Michael O'Sullivan	Discussion and editing
Cameron Walker	Discussion and editing

Certification by Co-Authors

The undersigned hereby certify that:

- the above statement correctly reflects the nature and extent of the PhD candidate's contribution to this work, and the nature of the contribution of each of the co-authors; and
- that the candidate wrote all or the majority of the text.

Name	Signature	Date
Michael O'Sullivan		7/11/19
Cameron Walker		7/11/19

— Chapter 1 —

Introduction

Healthcare organisations in New Zealand face an increasing pressure to operate efficiently as the populations they serve grow and age (MacPherson 2016). Significant effort is continuously spent by the management of these organisations scheduling medical personnel (also known as rostering) and scheduling appointments and operations. Improvements to scheduling processes can result in better outcomes for both patients and staff, and reduce the workload of the management.

At the same time technological advances mean that more and more information is being gathered about each patient's interactions with these organisations. For example during a typical visit to an emergency department the times that: the patient arrives; is triaged by a nurse; is seen by a physician; moves from one room to another; and finally leaves; are all recorded.

In this thesis we investigate whether patient data can be better utilised in scheduling models in healthcare, particularly when making decisions under uncertainty. We develop two optimisation models, one that uses historical patient admission and discharge data to inform physicians' schedules (a.k.a. rosters), and one that uses empirical distributions derived from historical surgical operation data to schedule surgical sessions. Both models incorporate elements of the uncertainty present in their respective areas, and risk averse objective functions that avoid the worst outcomes. In addition we demonstrate the application of the models to case studies in hospitals in New Zealand and determine the value of using the models and incorporating patient data into these decisions.

The two models address problems that were brought to our attention through prior collaboration with healthcare facilities, and inspired further research into areas believed to be both practically useful and scientifically valuable. We were approached first by clinicians from Auckland City Hospital (ACH) and then from Waitakere Hospital (WTH) in regards to rostering physicians in their general medicine (GM) departments. Similarities in the objectives, balancing physicians workloads, and roster requirements of the two problems spurred our interest in developing a generic model for incorporating patient pathway data and multiple groups of physicians. One of the rosters created, which was shown to reduce the difference in physician workloads, has been validated and implemented by WTH. Similarly, initial contracted research for Middlemore Hospital (MMH) to identify whether their surgical service capacity was capable of reaching their targets led to the identification of further opportunities to improve the service explored in this thesis. Despite the scheduling models developed showing promise in reducing waiting times and the risk of performing overtime, clinicians at MMH are reluctant to relinquish scheduling decisions to a mathematical model and therefore the validation and implementation of these models is not possible at present. We are, however, exploring opportunities to validate and implement the models at other healthcare facilities.

Although the two models we present are from separate fields, personnel scheduling and surgery scheduling do have some common features. Some similarities are clear, such as an objective of minimising cost and that both problems can be modelled using mixed integer programmes. Not as immediately apparent, is that the two problems share a common overall process consisting of three broad phases: demand modelling; block scheduling; and individual scheduling. Although the boundaries between the phases (particularly block and individual scheduling) are not always distinct and two phases are occasionally addressed in a single model, most problems encountered can be classified into one of the three phases. These phases are usually tackled separately because combining them together often results in a problem that is too complex to solve (Ernst et al. 2004), or because the inter-phase dependencies (i.e., dependencies between the phases) cannot be modelled using the same framework that is used for the intra-phase dependencies (i.e., dependencies within individual phases) that are, for example, modelled using mixed integer programming.

For personnel scheduling the three step approach is described by both Wolfe (1979) and Ernst et al. (2004), although the latter separates block scheduling into days off scheduling, shift scheduling and line of work construction, and individual scheduling into task assignment and staff assignment. The demand modelling phase involves determining the number of staff that are required at different times over a planning period. This could be performed by analysing historical data of demand and service rates over time and calculating the number of staff required to meet the demand. Alternatively the advice of experts in the relevant field, or one of many other methods could be used.

The block scheduling phase, sometimes called shift scheduling, deals with finding feasible shifts for staff to work that meet the demand that has been determined in the previous (demand modelling) phase. Rules governing the length of shifts need to be considered in this phase.

The individual scheduling phase, referred to as shift allocation, assigns personnel to the shifts, subject to labour laws. This phase is often combined with the block scheduling phase in personnel scheduling as the assignment of personnel to shifts is highly dependent on the shifts considered, and there may be interaction between the rules that govern feasible shifts and those that govern which shifts personnel can work (Ernst et al. 2004).

For surgery scheduling the three phases are detailed by Zhu et al. (2018). The demand modelling phase is usually called case-mix planning and determines the amount of operating room (OR) time, or number of blocks, that each surgical specialty, or surgeon, requires to complete their operations. This allocation may be based on forecasts of the number and duration of the surgeries for each specialty or surgeon.

Given the required number of OR blocks for each specialty, the block scheduling phase, referred to as master surgery scheduling, assigns blocks of time in ORs to each specialty and penalises under supply. Many other considerations are often taken into account such as the capacities of services before or after the ORs or availability of staff (Zhu et al. 2018). The master surgery schedule (MSS) is often cyclic and rotates on a weekly or monthly basis (Zhu et al. 2018).

The individual scheduling phase for surgery scheduling is usually split into two problems called advance scheduling and allocation scheduling. Advance scheduling assigns a date for each surgery, whereas allocation scheduling determines the exact start time of the surgery. In either case the problems deal with assigning individual surgeries to the blocks in the master surgery schedule. In contrast to personnel scheduling, master surgery scheduling is not often combined with advance or allocation scheduling (Cardoen, Demeulemeester, and Beliën 2010).

The concept of a planning horizon also exists in both personnel and surgery scheduling. The planning horizon is the time period into the future that is being planned for. The planning horizon may be different in each of the three phases, and typically it would decrease as planning moves from demand modelling to block scheduling to individual scheduling. In personnel scheduling the planning horizon may be on the scale of years for demand modelling, years or months for block scheduling and months or even weeks for individual scheduling. For surgery scheduling the planning horizons are typically shorter with demand modelling on the scale of years or months, months or weeks for block scheduling and weeks or days for individual scheduling.

This thesis considers both the block and individual scheduling phases for personnel scheduling, and the individual scheduling phase for surgery scheduling, as well as incorporating uncertainty and risk aversion into the respective models. Section 1.2 outlines the structure of the thesis after section 1.1 presents background information about the New Zealand health system.

1.1 New Zealand Health System

Before proceeding with the rest of the thesis we first present a brief overview of the New Zealand Health system. The aim of this outline is to provide the context for the models we develop and the case studies we apply them to.

The Ministry of Health (MoH) is a public service department that is part of the Government of New Zealand and oversees healthcare in New Zealand. The MoH plans and funds public health services, monitors the performance of the health system, and provides advice to the Minister of Health on improving health outcomes.

The MoH manages its public funds, which in 2019/20 amount to almost $20 billion, through a programme called Vote Health (The New Zealand Ministry of Health 2018a). Vote Health forms the majority of the funding for New Zealand's health system, however there are other important sources of funding such as: the Accident Compensation Corporation, local government, and private insurance.

About 80% of the funding provided by Vote Health is allocated to District Health Boards (DHBs). DHBs are local public organisations that are responsible for providing health services in their region, and are governed by a board of up to 11 members of which seven are publicly elected (The New Zealand Ministry of Health 2018b). There are 20 DHBs throughout New Zealand and their main objective is to improve, promote, and protect the health of the people in their district. In return for the funding from the MoH the DHBs negotiate accountability arrangements with the MoH, in which they agree to provide health services to their district's population (The New Zealand Ministry of Health 2018c). DHBs are particularly important to this work because they own and operate public hospitals, which are where the solutions generated by the models developed in this thesis would be implemented. The DHBs that operate hospitals would benefit from the use of these models for personnel and surgery scheduling in their hospitals.

1.2 Thesis Structure

The remainder of this thesis is structured as follows. First, in chapter 2, we review both the personnel and surgery scheduling literature. For personnel scheduling we focus mainly on physician rostering problems as they are most closely related to our work. Of particular interest in the surgery scheduling literature are the methods for dealing with both the due dates of operations and the uncertain nature of their duration.

The next two chapters focus on a physician rostering problem. In chapter 3 a mixed integer programming model is presented that generates a cyclic roster for physicians with the objective of balancing the number of patients for which each physician is caring. The model combines historical data on the patients admitted to and discharged from the hospital, and the pathways that the patients follow during their stay to determine the patient workloads of the physicians.

The model developed in chapter 3 is then applied to two case studies from New Zealand hospitals in chapter 4. These case studies serve both as an illustration of how the model can be applied to real problems and as an opportunity to assess the value of using the model.

The focus then shifts to surgery scheduling and in chapter 5 another mixed integer programming model is presented that schedules a list of operations into surgical sessions. The two main features of the model are, first, an objective function that focuses on the due dates of the operations, including a risk averse objective that focuses on the latest operations. Second, several methods for modelling the uncertainty of operation durations' are considered, including approaches that estimate the probability that a session runs overtime based on empirical distributions from historical operation and session data. The due dates of operations and their uncertain durations are combined in this model, although they occur on different time scales (the intermediate and short term respectively). Due dates and uncertain durations are combined because the alternative is to solve two separate problems: first selecting which operations should be performed within a shorter time scale (one or two weeks) based on their due dates; second assign the selected operations to surgical sessions while taking into account their uncertain durations. The first step in this alternative process limits the possibilities in the second step, and so a combined problem is used to avoid this limitation.

Chapter 6 applies the surgery scheduling model presented in chapter 5 to a case study from another hospital in New Zealand. The effects of several model parameters on the surgical schedule are explored through computational experiments using data from 2016.

Finally, chapter 7 summarises the main findings from the previous chapters and compares and contrasts the approaches used in the optimisation models, particularly with regards to uncertainty and risk.

— Chapter 2 —

Literature Review

Personnel and surgery scheduling have both been studied extensively beginning around 1970. Comprehensive reviews of the literature are available for both areas: Ernst et al. (2004) and Van den Bergh et al. (2013) review personnel scheduling; Cardoen, Demeulemeester, and Beliën (2010), and Zhu et al. (2018) review surgery scheduling. Therefore we present here a review of only some of the relevant literature in both fields. In addition we briefly review the literature on Benders' decomposition, since its use forms part of this thesis. A complete review of Benders' decomposition literature is provided in Rahmaniani et al. (2017).

2.1 Personnel Scheduling

We first review personnel scheduling in general before narrowing the focus to healthcare applications, and in particular those dealing with the continuity of care of patients. We began by searching for review publications and found Ernst et al. (2004) and Van den Bergh et al. (2013) and used the publications reviewed as the basis of our review. Additionally we used Scopus to search for more recent publications from the latest review paper in 2013 to the time of the search at the beginning of 2016. We combined the terms healthcare, physician, and nurse with either rostering or personnel scheduling, and limited the search to the following journals: Annals Of Operations Research, Computers And Industrial Engineering, Computers And Operations Research, Decision Sciences, European Journal Of Operational Research, Flexible Services And Manufacturing Journal, Health Care Management Science, Human Resources For Health, International Journal

Of Production Economics, Journal Of Scheduling, Operations Research For Health Care, and Operations Research. This search returned 67 results, and after reviewing the abstracts for relevance to patient pathways, continuity of care, and uncertain admissions, 11 of the 67 publications were reviewed in full.

As mentioned in chapter 1 personnel scheduling is separated into three phases: demand modelling, block scheduling, and individual scheduling. The personnel scheduling model developed in chapter 3 includes elements of both block and individual scheduling (also referred to as shift scheduling and shift allocation), as it both determines the order of the shifts and the staff that they are assigned to. We therefore focus on these two phases in the literature review.

An important concept in shift scheduling and allocation is the distinction between cyclic and acyclic rosters. In a cyclic roster a single sequence of shifts is made for all of the employees. Each employee starts at a different point in the sequence and, when an employee works the last shift in the sequence, they go back to the beginning and work the first shift again (see table 3.1 for an example). Conversely, in an acyclic roster each employee has a sequence of shifts that they work, which can be different from the sequence that other employees work. Tables 2.1 and 2.2 show examples of cyclic and acyclic rosters respectively, where A, P, O, and X represent different types of shifts that the employees can work.

Table 2.1: Example Cyclic Roster

Employee				Shifts						
Week 1	Week 2	Week 3	Week 4	M	T	W	T	F	S	S
1	4	3	2	A	P	O	O	A	P	O
2	1	4	3	O	O	A	P	O	X	X
3	2	1	4	O	A	P	O	O	A	A
4	3	2	1	P	O	O	A	P	X	X

Table 2.2: Example Acyclic Roster

Employee	Shifts–Week 1							Shifts–Week 2							Shifts–Week 3							Shifts–Week 4						
	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S
1	O	O	A	P	A	P	O	O	A	P	A	P	X	X	O	A	P	O	O	A	A	P	O	O	O	O	X	X
2	A	P	O	A	P	X	X	A	P	O	O	O	A	A	P	O	O	O	O	X	X	O	O	A	P	A	P	O
3	O	A	P	O	O	A	A	P	O	A	P	O	X	X	O	O	A	P	A	P	O	O	A	P	O	O	O	O
4	P	O	O	O	O	X	X	O	O	O	O	A	P	O	A	P	O	A	P	X	X	A	P	O	A	P	A	A

Bailey (1985) presents an approach for integrated days off and shift scheduling, the combination of which is referred to as the tour construction problem. The objective of the mixed integer programme (MIP) that they formulate is to minimise the cost of labour and under staffing. As the integrated problem is a combination of two set covering problems a set of feasible patterns is required for each day.

Balakrishnan and Wong (1990) solves the ‘rotating workforce scheduling problem’, which is equivalent to a cyclic scheduling problem, using a network flow model. The beginning of the first day in the planning horizon is the source node, and the end of the last day is the sink node. Arcs represent work or rest periods, which are consecutive days on which the same shift is performed (work period) or consecutive days off (rest period). By defining the costs on the arcs as the cost of the corresponding work period, the shortest path from the source to the sink gives the lowest cost cyclic roster.

Naudin et al. (2012) considers three mathematical models to solve a staff scheduling problem. The problem consists of a set of tasks that need to be performed on each day of the planning horizon, and a set of employees, each of which can perform a subset of the tasks. The first model uses binary assignment variables for determining which personnel perform tasks. The second and third models are both Dantzig-Wolfe decompositions of the first model, the difference being that the second uses valid daily schedules as the columns whereas the third uses weekly schedules. In both instances column generation is used within a branch and bound framework to find valid schedules. Valid inequalities are also derived that are shown to improve the performance of the models.

Rocha, Oliveira, and Carravilla (2013) proposes an integer programme for the staff scheduling problem, with a novel formulation of sequence constraints. Two case studies are used to demonstrate the flexibility of the model. It is also applied to a set of benchmark instances; however the MIP formulation does not outperform a tabu search heuristic.

2.1.1 Healthcare

Within the personnel scheduling literature a large part has focused on the healthcare setting. The interest in the healthcare sector is because both the size of the workforce and the 24 hour nature of the services provided make the resulting personnel scheduling problems challenging. In addition labour costs represent a large part of most healthcare organisations total costs, providing an incentive to improve their rostering methods.

The rostering of nurses in particular has received a lot of attention for several reasons. First because the nursing workforce often has non-homogeneous skills, and can therefore perform different tasks. Second, since nurses are often paid hourly, as opposed to via a salary, nurses working overtime is a major cost for healthcare organisations. Third labour laws and restrictions imposed by collective agreements can make creating rosters for nurses more complex. These factors combine to make nurse rostering both a time consuming and costly problem for healthcare organisations and an interesting challenge for the academic community.

Nurse rostering problems are categorised in De Causmaecker and Vanden Berghe (2011) by introducing an $\alpha|\beta|\gamma$ notation similar to that used in machine scheduling. The α category represents the personnel environment such as the availability and skills of staff, or constraints on individual workload. The β category represents the services delivered and the time structure of the problem, for example the number and type of shifts that need to be performed. The γ category represents the objective function, which could be soft versions of constraints in the α and β categories, or any other desired characteristic. A 'soft' constraint is when the amount that a constraint is violated by is included in the objective function.

Millar and Kiragu (1998) rosters nurses on 12 hour cyclic shifts. All possible 'stints' (sequences of shifts and days off) are first defined. Using 12 hour shifts means that there are fewer possible stints as there are only two shifts each day (the day shift and the night shift). A graph is then formed using the stints as nodes, which can be used to find both cyclic and acyclic rosters.

Burke and Curtois (2014) introduces new approaches to nurse rostering benchmarks using branch and price. A method based on regular expressions is developed to model many of the rules defining patterns of shifts. Dual stabilisation and bounding methods are used to improve the pricing problem, which is a resource constrained shortest path problem that is solved using dynamic programming. Constraint branching is also employed within the branch and bound tree.

Another set of benchmark instances has been created by the International Nurse Rostering Competition, to facilitate the comparison of various solution methods. Santos et al. (2014) present a compact MIP that models all of the constraints in the competition instances. Clique and odd-hole cuts are shown to be effective in improving the dual bound, in particular the inclusion of all cuts found, not just the most violated one, is suggested. A MIP based heuristic is also presented that uses a variable neighbourhood descent approach.

Physicians, in contrast to nurses, have received less attention in the personnel scheduling literature. Nevertheless since the rostering model presented here is designed for a physician rostering problem a larger part of this section is dedicated to physician rostering.

Many of the physician rostering problems in the literature consider rostering physicians in the emergency department (ED). For example, Beaulieu et al. (2000) use a set partitioning model to schedule physicians in the ED. Their objective is to minimise the weighted sum of deviations from certain goal constraints. Since including all of the constraints renders the problem infeasible, an initial solution is found by removing conflicting constraints. An iterative process is then performed, where violated constraints are re-introduced and the problem re-solved. This process terminates when a feasible solution cannot be found.

Another approach to rostering physicians in the ED is presented in Ferrand et al. (2011). Binary variables are used to assign individual physicians to shifts and three categories of constraints are considered: regulatory, work requirements, and personal preferences. The objective function is to minimise the amount that the constraints are violated.

Ganguly, Lawrence, and Prather (2014) aims to balance staffing costs with patient service levels. Both the demand modelling and block scheduling phases are examined in a combined problem. The model determines the number of staff to allocate to predetermined shift schedules, which account for labour laws and other work rules. A chance constraint is used to ensure a minimum level of service in each time period, taking into account the different skill levels of the staff.

El-Rifai et al. (2015) attempts to find start times and durations for physicians' shifts to minimise the time patients spend waiting in an emergency department. A queuing network model of the department is used to form balance equations that track each patient's progress through the department. These equations are incorporated into an integer programme, along with a patient arrival scenario generated from historical data. Stochastic optimisation is then used to find the shifts that minimise the average patient waiting time. Personnel are not assigned to the shifts, and therefore the assignment of patients to personnel is not modelled.

Uncertain demand is also addressed in Kim and Mehrotra (2015) where a two stage stochastic program is developed to minimise the cost of over or understaffing in future demand scenarios. Feasible staff schedules, and future adjustments to these schedules, are pregenerated based on staffing and scheduling rules.

Kortbeek et al. (2015) also considers uncertainty in the workloads of nurses and uses a bed census prediction model to find the most cost effective numbers of nurses to be allocated to various care units and to a flexible pool.

Outside of EDs, Bard, Shu, and Leykum (2014) presents an approach for scheduling physicians in primary care clinics. The problem considered aims to maximise the number of trainees assigned to clinic duty each month. The constraints are split into soft constraints, which represent other goals such as ensuring each trainee is assigned one clinic session per week, and hard constraints, such as a limit on the number of trainees from each team that are on clinic duty each day. A complexity analysis is performed on the problem, which shows that a simplified version of the problem can be modelled as a minimum cost flow problem; however the full problem is more complex.

Bruni and Detti (2014) proposes a flexible MIP formulation for a physician scheduling problem. Shifts in the planning horizon are organised into regular working days, regular nights, and Sundays. The physicians are partitioned into groups which can work with different capabilities in different departments, each department requires a certain number of doctors of a given competency at any point in the day. Workload is balanced among the physicians by imposing a maximum either in the total number of shifts, or a weighted sum of the shifts based on the difficulty or inconvenience of the shift.

Brunner and Edenharter (2011) also incorporates different experience levels of physicians in a MIP formulation of a long term staffing problem. The objective is to minimise the total number of staff, by deciding the start time and length of shifts, while meeting a time-varying required number of physicians. Within the planning horizon each week is independent, and can be solved separately. The weekly problems are further decomposed in a column generation approach, with the demand constraints kept in the master problem and the sub-problems (SPs) used to generate feasible schedules.

Fairness, in terms of both total working hours and the distribution of specific undesirable shifts, is integrated into a physician rostering model in Stolletz and Brunner (2012). A tour scheduling problem is presented, which assigns physicians shifts and days off over a two week planning horizon, with the aim of minimising the overtime hours required to meet the demand for physicians. A shift matrix is generated in a preprocessing step instead of incorporating the shift constraints into the problem.

Previous work on improving patients' continuity of care through physician scheduling has included several approaches. For example, Kazemian et al. (2014) attempts to minimise the number of patient handoffs, where a handoff is defined as when a patient is handed from one physician to another, typically at the end of their shift. Using historical data the number of handoffs incurred by a shift change at any time of the day is estimated. The starting and ending times of physicians' shifts that minimise the number of handoffs can then be found. Additionally, Smalley, Keskinocak, and Vats (2015) develops a handoff metric based on how recently each physician has worked. A roster can then be formed which minimises this metric, while meeting demand for physicians and complying with regulatory constraints.

Continuity of care has also been a focus outside of hospitals, for example in home care. The amount of overtime nurses perform while providing home care is minimised in Lanzarone and Matta (2014) while ensuring that continuity of care is maintained.

From this review we have identified the following areas that we believe are gaps in the current physician rostering research:

- workloads of individual staff members being linked to patient admission and discharge data within a MIP;
- taking into account uncertainty in admissions and discharges when assigning physicians to shifts, as opposed to finding shift timings as in Ganguly, Lawrence, and Prather (2014) and El-Rifai et al. (2015);
- using risk measures to create rosters that perform well in the worst case scenarios; and
- considering multiple cyclic rosters with independent groups of physicians and rules in the same framework.

2.2 Surgery Scheduling

The management of operating rooms (ORs) has been studied thoroughly and several reviews of the literature are available such as Cardoen, Demeulemeester, and Beliën (2010), and Zhu et al. (2018). Aside from publications found through these reviews we once again used Scopus to search for more recent publications from 2018 to the time of the search at the beginning of 2019. We combined the terms surgery and operating rooms with scheduling, and limited the search to the same set of journals which is: Annals Of Operations Research, Computers And Industrial Engineering, Computers And Operations Research, Decision Sciences, European Journal Of Operational Research, Flexible Services And Manufacturing Journal, Health Care Management Science, Human Resources For Health, International Journal Of Production Economics, Journal Of Scheduling, Operations Research For Health Care, and Operations Research. This search returned 48 results, and after reviewing the abstracts for relevance to due date based objectives, and approaches to chance constraints, 8 of the 48 publications were reviewed in full.

We consider the three phases of the OR scheduling process to be: case mix planning (cf. demand modelling), master surgery scheduling (cf. block scheduling), and patient scheduling (cf. individual scheduling). These three phases are not rigorously defined and there is some disagreement about the boundaries between the phases, in particular Cardoen, Demeulemeester, and Beliën (2010) propose a more detailed ‘decision delineation’ both because of a lack of clarity between the three phases, and to provide a structured approach to allow researchers to detect contributions in their area of interest. We keep to the three phase model as the disagreement between the phases is minor and we believe it better reflects the process of managing ORs in practice.

This review focuses on patient scheduling, which addresses the problems of assigning an OR, day and time to each patient that requires an operation. These problems are combined in various ways, for example ‘advance scheduling’ refers to fixing a date for each patient’s operation, whereas ‘allocation scheduling’ refers to determining an OR and sequence for the patients’ operations.

We define a surgical session as a period with a specified start and end time, in a particular OR, on a particular day. Using this definition master surgery scheduling specifies the start and end times of the sessions, and possibly assigns them to specialties. What we refer to as sessions are frequently called ‘blocks’ in the literature.

Many different objectives have been studied for patient scheduling problems, however by far the most common are those which relate to the utilisation and (over-)usage of the ORs. Marques, Captivo, and Vaz Pato (2012) proposes an integer programme to create a weekly schedule for elective operations from a waiting list, with the goal of maximising the utilisation of the ORs. Pulido et al. (2014) aims to minimise the sum of three costs: the underutilisation of the ORs;

the overtime of the ORs; and the time that surgeons spend waiting. A MIP is developed that schedules operations in ORs and assigns surgeons to each operation, and a simulation model is developed to evaluate solutions. Van Huele and Vanhoucke (2014) tackles an integrated physician and surgery scheduling problem using the most commonly encountered constraints from each problem, and the objective is to minimise the overtime costs of the ORs.

The time that patients spend waiting for their operations, or the deviation of the time the operation is started from when it is due (earliness/lateness) are also used as objectives, although not as often as OR related costs. Addis et al. (2016) present a scheduling and re-scheduling approach where the day and OR block is assigned to operations on a waiting list for three weeks in advance. A rolling horizon approach is used where the first week of the solution is fixed, then after a realisation of that week, where some operations are cancelled because of random operation durations, the problem is stepped forward and solved again. The objective is to minimise a combination of the time that patients spend waiting for their operation, and the lateness of the operations. Ballestín, Pérez, and Quintanilla (2018) builds a schedule of elective operations using two phases. First an initial schedule is determined for the next two weeks, and four objectives are compared in this phase. Second a final schedule is made for the next week, taking into account cancellations and unavailability of surgeons and equipment. The four objectives that are compared all relate to the due dates of the operations and aim to reduce the percentage of patients that breach their due date.

Occasionally, objectives are combined in a multi-objective approach, for example Cardoen, Demeulemeester, and Beliën (2009a), where the priority of each operation, time taken to travel to the hospital, time spent recovering, and the peak number of beds used are all considered as objectives. Rachuba and Werners (2014) considers three objectives: the total waiting time of the patients; the total OR overtime; and the number of patients deferred to the next planning period. A MIP is then used to assign patients to sessions. Marques, Captivo, and Vaz Pato (2015) maximises both the utilisation of the ORs and the number of operations performed, and notes that because of the idle time between operations there is a trade off between the two objectives. When maximising utilisation longer operations are preferred since there are fewer gaps, however shorter operations are preferred when maximising the number of operations performed. They propose a two phase heuristic to solve the bi-criteria problem.

One method of differentiating surgical scheduling processes is whether ‘block’ or ‘open’ scheduling is used. In ‘block’ scheduling, blocks, which are equivalent to what we defined earlier as sessions, are assigned to specialties or surgeons. Surgeries can only be booked in a session that has been assigned to their specialty or surgeon. For example, Shylo, Prokopyev, and Schaefer (2013), Rachuba and Werners (2014), and Landa et al. (2016) use block scheduling. In contrast ‘open’ scheduling allows any operation to be booked in any session regardless of specialty or surgeon such as in Fei, Chu, and Meskens (2009), and Y. Wang, Tang, and Fung (2014).

Uncertainty, from sources such as the duration of operations or the arrival of emergency cases, has been addressed frequently in the literature, using a variety of techniques. The duration of operations is most commonly addressed. For example, Denton, Viapiano, and Vogl (2007) assigns the start time of operations within a session and represents the operation durations using scenarios, in addition to proposing several heuristics to solve the start time assignment problem. Using scenarios to represent uncertainty in operation durations is common as in Jebali and Diabat (2015), where a two-stage stochastic programme is presented that minimises patient-related costs and expected utilisation costs. The stochastic programme is solved using sample average approximation (SAA). Landa et al. (2016) also uses scenarios to represent the uncertainty in operation durations.

The arrival of emergency cases is another common source of uncertainty addressed in the literature. Lamiri, Grimaud, and Xie (2009) formulates a stochastic planning problem to determine how elective cases should be assigned to sessions, and the OR capacity that should be reserved for emergency operations in order to minimise the sum of patient assignment and expected overtime costs. Surgery duration and arrival uncertainty are also tackled simultaneously such as in Min and Yih (2010) where a stochastic dynamic programme is developed to determine the number of patients selected from a waiting list, while taking into account the priorities of the patients. Kroer et al. (2018) also tackles uncertain operation durations and emergency arrivals and aims to reduce the amount of overtime worked. To take the uncertain durations and arrivals into account a stochastic model is used, and two heuristics are proposed to solve it.

Another approach to the uncertainty of operation durations is taken by Hans et al. (2008) which aims to free OR capacity by maximising the number of empty ORs, while minimising the risk of overtime. To achieve this they allocate a fixed amount of each session as ‘planned slack’, which is proportional to the square root of the sum of the variances of the operations planned in that session. Constructive heuristics and local search methods are used to find solutions. Shylo, Prokopyev, and Schaefer (2013) presents a similar approach where an approximation of the sum of the durations of operations assigned to a session, and the sum's standard deviation, is included in an integer programme under the assumption that the sum of the durations is normally distributed. A piecewise linear approximation is used for the standard deviation of the sum and an iterative procedure is proposed where the discretisation used for the approximation is updated at each step.

A further approach to the uncertainty in operation scheduling is used by Addis et al. (2016) and Marques and Captivo (2017). Both employ a 'level of robustness' approach that assumes a subset of the operations assigned to a session each take their worst (longest) possible value, while the remaining operations all take their expected duration. The solutions are made more robust by increasing the size of this subset. Others such as Denton, Miller, et al. (2010), Sagnol et al. (2018), and Vali-Siar, Gholami, and Ramezanian (2018) use a similar concept termed a 'budget of uncertainty', which is founded on the idea that operations durations must lie with an uncertainty set which gives lower and upper bounds on how long an operation can take. The budget of uncertainty is an upper bound on the number of operations that will take their longest possible duration on a particular day. J. Wang et al. (2018) integrates this approach with a scenario based approach.

Of particular interest are the constraints that ensure that sessions do not run overtime. In the simplest case the sum of the durations of operations assigned to the session must be less that the duration of the session. If the durations of the operations are uncertain then this constraint will mean that the session will not run overtime on average, but there may be realisations in which it does. Most approaches enforce a soft constraint on the session duration, where the overtime (or expected overtime) is penalised in the objective function, for example Lamiri, Dreo, and Xie (2007), Cardoen, Demeulemeester, and Beliën (2009b), Fei, Chu, and Meskens (2009), and Van Huele and Vanhoucke (2014). Others such as Oostrum et al. (2008), Shylo, Prokopyev, and Schaefer (2013), Y. Wang, Tang, and Fung (2014), Landa et al. (2016), and Kamran, Karimi, and Dellaert (2018) use a 'chance constraint' that ensures that each session will not run overtime with some given probability.

Incorporating due dates for operations is often neglected as a factor when scheduling operations. Whether this is because of the focus on utilisation measures as objectives or because due dates are not well defined in many health systems is unclear. Occasionally, a deadline is enforced for each operation, for example in Fei, Chu, Meskens, and Artiba (2008), where operations are assigned to ORs to minimise the total operating cost under the condition they are completed before their deadline. They present a MIP, which they solve using a branch and price approach, including node selection and branching strategies. Fei, Chu, and Meskens (2009), Marques, Captivo, and Vaz Pato (2012), Marques, Captivo, and Vaz Pato (2015), and Molina-Pariente, Hans, and Framinan (2018) also use deadlines to enforce due dates for operations.

One drawback of using a fixed deadline for operations is that it is possible that the model is infeasible if there are too many operations that must be performed before a given time. This is common in practice when a surgical system is at or over capacity and there are long waiting lists for elective operations. Instead the earliness or lateness of each operation can be captured in the objective function. As previously described, this approach is used in Addis et al. (2016) and Ballestín, Pérez, and Quintanilla (2018). Others such as Marques and Captivo (2017) and Barrera et al. (2018) also penalise the lateness of operations using various objective functions.

The gaps identified in the surgery scheduling literature are:

- using a chance constraint for session duration that is based on an empirical distribution conditional on the operations assigned to the session; and
- using a risk measure, applied to the early and lateness of operations, as an objective.

2.3 Benders' Decomposition

Benders' decomposition (Benders 1962) is a technique for solving MIPs by fixing the integer variables and reducing the problem to a linear programme (LP). Although this method can be applied to any MIP it is most useful when special structures exist in the constraints of the MIP that mean the resulting LP is particularly easy to solve, for example if it can be modelled as a network problem, or if the problem decomposes into many smaller problems.

The algorithm works by solving two types of problem: the restricted master problem (RMP), and the sub-problem (SP). The RMP is a relaxed version of the original problem with only the integer variables and any constraints that only contain the integer variables. This problem is solved to obtain feasible values for the integer variables and a lower bound for the problem. These values of the integer variables are then used in the SP along with the continuous variables and all the constraints from the original problem that are not in the RMP to generate a cut which is added to the RMP, and provide an upper bound for the problem. The algorithm alternates between solving the RMP and SP until the bounds converge.

Many improvements to Benders' decomposition have been developed since its inception and Rahmaniani et al. (2017) provide a comprehensive review of the literature. Enhancements are categorised as one of: solution procedure, solution generation, cut generation, or decomposition strategy.

One of the first improvements, Magnanti and Wong (1981), considers the problem of choosing an optimal solution to the dual SP from many optimal solutions. The concept of Pareto-optimal cuts is introduced along with an LP to find a Pareto-optimal cut, given a core point – a point in the relative interior of the convex hull of the master problem (MP). Although finding a Pareto-optimal cut requires an additional LP to be solved at each iteration, the quality of the cuts generated results in a reduction of overall computation for the problems examined.

Birge and Louveaux (1988) proposes a multi-cut L-shaped algorithm for solving two-stage stochastic programmes with recourse. They show that the multi-cut method performs better in the worst case, which for Benders' decomposition suggests that if the SP is decomposable and disaggregated cuts are able to be formed they are preferable to an aggregated cut.

The method for finding Pareto-optimal cuts introduced by Magnanti and Wong (1981) was further improved by Papadakos (2008), which showed that only one LP needs to be solved each iteration as long as a different core point is used at each iteration. It is also shown that a new core point can be obtained by a convex combination of the previous core point and the LP solution. However, obtaining an initial core point is still difficult in some situations.

Fortz and Poss (2009) incorporates Benders' cuts into a branch and cut framework with Benders' cuts being generated when a new incumbent solution is found in the branch and bound tree. They also discuss initialising the branch and bound tree with a pool of cuts generated by solving the LP relaxation of the root node using a cutting plane algorithm.

Fischetti, Salvagnin, and Zanette (2010) formulates an extended primal SP which is infeasible if and only if a violated cut (either feasibility or optimality) can be generated. An LP is then proposed that detects a 'minimal source of infeasibility', and generates a most-violated cut. This also allows feasibility and optimality cuts to be generated in the same framework.

Naoum-Sawaya and Elhedhli (2013) also discusses Benders' cuts implemented in a branch and cut framework. They use the analytic centre cutting plane method (ACCPM) to solve the master problem at each node and generate multiple cuts at each node by repeatedly solving the linear relaxation using a cutting plane algorithm. The cuts generated using ACCPM are shown to be Pareto-optimal and a method for updating the analytic centre after adding cuts is provided.

Although the implementation of Benders' decomposition we use is not novel, nor the effectiveness of its use surprising, the review of the literature was useful in determining techniques that were likely to be effective for our problem.

Conclusion

We have identified several gaps in both the physician rostering and surgery scheduling literature. The gaps identified in the physician rostering research are:

- workloads of individual staff members being linked to patient admission and discharge data within a MIP;
- taking into account uncertainty in admissions and discharges when assigning physicians to shifts;
- using risk measures to create rosters that perform well in the worst case scenarios; and
- considering multiple cyclic rosters with independent groups of physicians and rules in the same framework.

The rostering model presented in chapter 3 addresses these by:

- incorporating patient pathways into a MIP to enable the calculation of physician workloads, which allows workloads to be balanced and continuity of care improved;

- optimising a roster over many scenarios of admissions and discharges to create rosters that are robust to this uncertainty;
- using risk a risk averse objective function to create rosters that perform well in the worst scenarios; and
- linking rosters in different areas through patient pathways to prevent the workloads of one group being improved at the expense of another.

The gaps identified in the surgery scheduling literature are:

- using a chance constraint for session duration that is based on an empirical distribution conditional on the operations assigned to the session; and
- using a risk measure, applied to the early and lateness of operations, as an objective.

In chapter 5 we present a surgery scheduling model that addresses these gaps by:

- using historical surgery and session data to define conditional empirical distributions of the probability that a session runs overtime given the operations assigned to the session, and using these empirical distributions in chance constraints to improve the efficiency of scheduling operations; and
- introducing a risk averse objective function in terms of the early and lateness of operations, as compared to an exponent applied to the lateness, in order to prevent operations with long durations having to wait longer before being performed.

— Chapter 3 —

General Medicine Rostering Model

In most hospitals the general (or internal) medicine department is responsible for diagnosing and treating patients that do not require surgery. Typically it is staffed 24/7 and accepts admissions from both the emergency department (ED), and referrals from external services such as general practitioners. When a patient arrives at general medicine (GM) they are assigned to a ward team made up of senior, junior, and trainee physicians. Generally they are assigned to the team of the physician that admitted them. The patient is diagnosed and the admitting physician determines whether they should remain in general medicine. If the patient does remain in general medicine then the ward team will continue to check up on the patient until they are discharged.

Unlike other departments that operate 24/7, such as the ED, patients can stay in GM for many days, or even weeks. Having the same physician treat a patient throughout their stay in hospital is termed continuous care. Continuity of care, however, can be disrupted if a ward team is responsible for too many patients, resulting in the transfer of some patients to another ward. When a patient's care is fragmented like this it hinders the formation of patient-doctor relationships, which is undesirable as Hjortdahl and Laerum (1992) shows these relationships improve patient satisfaction. Findings from Epstein et al. (2010) also suggest that increased fragmentation may lead to increased length of stay for patients with pneumonia or heart failure. In addition Epstein et al. (2010) suggests changes to personnel scheduling as a possible means of reducing fragmentation.

The negative impacts of fragmenting a patient's care mean that the GM department is concerned with balancing the workloads of the teams to improve the continuity of care of the patients. The continuity of care that patients experience can be improved by changing the physicians' rosters. The rosters dictate who is working each day and therefore who is admitting new patients and which ward teams they are assigned to. If the roster is unfavourable to one team they may end up admitting and, hence, caring for many more patients than the other teams. This imbalance in patient workload can be corrected by transferring patients from one team to another, however this fragments the patient's care. Fewer transfers need to be made if the workloads of the teams are kept balanced. Since the roster determines how many patients each team is assigned it can be used to balance the workloads of the teams.

Cyclic rosters are perceived as 'fair' to staff as all staff work the same shifts over a complete cycle. In the long term this should mean that, on average, staff experience the same workloads. A possible exception to this is if there are longitudinal effects on the workload that align with the cycles of the roster, for example if the first weekend of the month often has a high number of admissions and the cycle length of the roster is also one month. Seasonal fluctuations in demand also cannot be accommodated by cyclic rosters, whereas an acyclic roster could plan for an increase in demand in winter. In practice, however, rosters are often replanned every 3 or 6 months as junior physicians being trained rotate through departments in the hospital. This offers opportunities to create separate cyclic rosters for the summer and winter seasons that can service different levels of demand.

Although cyclic rosters ensure that workloads are fair in the long term, at any point in time a cyclic roster does not inherently balance the current workload of the ward teams, since even short sequences of shifts can have a large short term impact on the number of patients a ward team is caring for. Therefore to determine rosters that balance ward team workloads day-to-day, and avoid long term effects that align with the roster's cycles, the relationship between individual shifts and team workload must be incorporated into the roster building process. Cyclic rosters do, however, ensure a fair distribution of the different types of shifts in the long term, and are used in our model for this reason.

There are many possible metrics that could be minimised in order to balance the workloads of the teams. For example the sum of the differences in workloads or the mean absolute deviation both measure the imbalance of workloads. In the model presented, the single largest difference in workloads between two teams at any point in time is minimised. This measure is extremely sensitive to outliers, however, it is also useful for improving continuity of care.

There are no hard and fast rules for when a patient will be transferred to another team to balance the workloads, nevertheless the larger the difference in workloads the higher the likelihood that a patient is transferred. Therefore minimising the largest difference in workloads will reduce the probability that patients are transferred between teams, and improve continuity of care. Even so, there are other objectives that would reduce the probability that patients are transferred between teams such as the number of times the difference in workloads exceeds a given threshold, however for the purposes of this model the objective is to minimise the largest difference in workloads.

The rest of this chapter presents the physician rostering problem of interest, and develops a mixed integer programme (MIP) model to solve it, and is based in part on Adams, O'Sullivan, and Walker (2019). Further it is shown that when Benders' decomposition is applied to the model the sub-problems (SPs) can be reduced from linear programmes (LPs) to simple calculations, both for the traditional version and a version proposed by Fischetti, Salvagnin, and Zanette (2010).

3.1 Problem Description

The rostering of physicians is a particularly complex problem because of the many regulations to which the rosters must adhere. For example, there are often rules around the handling of night shifts or a night 'run' (where several night shifts are worked consecutively), and how frequently physicians must have the weekend off. In New Zealand the major restrictions from a national collective agreement are: that physicians must have at least every second weekend off, after night shifts are worked physicians must have three days off to recover, and any shift that ends after 4 p.m. gets paid overtime.

In the GM departments we have investigated, physicians usually work in teams of at least two, which include a senior consultant, a registrar, and possibly additional trainees (e.g., a house officer). Registrars are physicians that are training to become specialists and perform placements in different departments. When they are in GM one of their responsibilities is admitting new patients. Since registrars admit new patients their roster determines the workloads of the teams. Therefore we focus on the problem of rostering registrars to balance workload. Registrars also have the most restrictions on their roster which makes creating rosters for them difficult to do by hand. The remaining physicians' rosters can be generated reasonably easily from the registrars' rosters.

For the purposes of the model we assume that the following two statements are true for every physician: 1) they are part of a group of physicians who share a cyclic roster; and 2) they are part of a team which aggregates the patients they admit together and care for them collectively. The team may be just themselves, or themselves and a senior consultant, or themselves, a senior consultant, and several other physicians.

Every day, each physician must be assigned to a shift or given a day off. Typically there are several types of shifts either with different roles or varying durations. At least one of these shifts includes the task of admitting new patients. The patient pathway, that is the way in which patients move through the department from first being seen by a physician to being discharged, is then taken into account to determine the number of patients that each of the teams cares for. It is also possible for a physician to be assigned more than one shift on a day. This only occurs when the physician is working night shifts and their regular duties are covered by another physician. In this case the physician can be assigned the night shift and one other shift, which the covering physician performs.

Not all shifts that the physicians have to perform are relevant to the rostering process. Those shifts that are relevant are either; related to admissions or the patients' pathways, or have rules associated with them that can influence those shifts. The shifts related to the admissions and pathways are relevant as they directly affect the workloads of the teams. A shift that has a rule associated with it, for example 'must be performed on the day after shift X' or 'must be performed by X physicians each day' can influence the workloads of the teams by restricting the possible assignments of the shifts related to admissions.

We assume that a cyclic roster is being used and one physician is working each week in the roster at any point in time. This means that assigning the shifts to physicians is equivalent to assigning the shifts to days and weeks in the cyclic roster – see section 3.2 for an example. Therefore, the problem description is:

> For each group of physicians that share a cyclic roster, we wish to assign the shifts required by that roster (and days off) to the days and weeks in the roster in such a way that minimises the largest difference in the workloads of the teams at any point over the planning horizon, subject to any restrictions on the rosters.

3.2 Mathematical Formulation

In this section we present a mathematical formulation of the problem described in section 3.1. The objective of the model is to balance patient workload across the teams. The workloads are balanced by minimising the largest difference in workloads among all of the teams throughout the entire planning period. The model is characterised by a set of parameters (tables 3.3 to 3.5), a set of functions (tables 3.6 and 3.7), and a set of decision variables (table 3.8). The parameters are split into three categories: general; those that depend on historical data; and those that depend on the particular rules of a hospital. The functions are also split into general functions and functions that depend on the hospital's rules.

The concepts of a roster group and a workload group are introduced. A *roster group* is a collection of personnel (in our case physicians) that are identical in terms of the types of shifts they can work and can therefore all be rostered on a cyclic roster. In hospitals these groups can be used to distinguish physicians from different departments, or physicians from nurses. *Workload groups* are groups towards which the personnel contribute their individual workload. In some situations the workload groups may be individual personnel. Alternatively, personnel's workloads may be aggregated together and shared between the personnel, such as when a group of physicians send all the patients they admit to the same ward and are responsible for all of the patients in that ward as a group. The problem then is to minimise the maximum difference in workloads between the workload groups.

For example, in the hospital considered in section 4.1 two distinct groups of physicians are rostered at the same time; those in the inpatient wards, and those in the assessment, diagnostic, and cardiology unit (ADCU). Each of these groups work different shifts in different locations and have separate rules to which their rosters must adhere, and therefore have separate cyclic rosters. The rosters are linked because of how admissions are split between the departments. In this example each physician is only responsible for the patients that they admit and there is no aggregation of workload. This means that each physician has their own workload group, making this feature of the model largely irrelevant for this case study. In the second case study, however, several physicians admit new patients to the same ward and collectively take care of the patients in that ward as a group.

An example of cyclic rosters for two roster groups, the physicians in the roster groups and the workload groups, is given in tables 3.1 and 3.2. In the first week physician 1 will work the shifts listed in row 1 of table 3.1. In the second week they will work the shifts listed in row 2 and so on. It should be noted that roster group 1 will work a 6 week cyclic roster whereas roster group 2 will work a 4 week cyclic roster, but workload group ‘Green’ refers to the same workload group in both roster group 1 and 2, i.e., registrars in both roster groups contribute to the Green workload. In reference to the shifts; ‘A’ stands for admission, ‘P’ for post-acute (following up with patients the day after they are admitted), ‘X’ for a day off, and ‘O’ for other. These are explained in more detail in section 4.1.

Table 3.1: Details for Roster Group 1

Registrar	Workload	Shifts						
	Group	Mon	Tue	Wed	Thu	Fri	Sat	Sun
1	Green	O	O	A	P	O	O	O
2	Blue	O	O	A	P	O	X	X
3	Green	O	A	P	O	O	A	A
4	Blue	P	O	O	O	O	X	X
5	Green	A	P	O	O	O	A	A
6	Blue	P	O	O	O	O	X	X

Table 3.2: Details for Roster Group 2

Registrar	Workload	Shifts						
	Group	Mon	Tue	Wed	Thu	Fri	Sat	Sun
7	Green	O	O	A	P	O	O	O
8	Blue	O	O	A	P	O	X	X
9	Green	O	A	P	O	O	A	A
10	Blue	P	O	O	O	O	X	X

3.2.1 Parameters and Sets

We start by introducing a number of parameters and sets for the model. Although our definition of roster and workload groups is applicable to any type of personnel, for the remainder of this work we use these groups for physician rosters only. The set O represents the separate roster groups, each roster group has: its own cyclic roster; a set of physicians as members; and a set of shifts. The set of physicians in roster group o is denoted by R_o (originally from R(egistrars)), and W_o is the set of weeks in a cycle of roster group o. Both R_o, and W_o are assumed to be sets of ordered integers from 1 to $|R_o|$, and $|W_o|$ respectively. This assumption is made to simplify the arithmetic required when dealing with cyclic rosters. The set of possible shifts for roster group o is S_o. Here a shift refers to a set of duties and responsibilities to be performed between a start and an end time. L is the set of workload groups, and $V_{o,r,l}$ is a binary parameter that describes which physicians contribute to each workload group and is 1 if physician r in roster group o contributes their patients to workload group l.

The set C is a set of integers corresponding to the days in the planning horizon, where each integer is the number of days since an arbitrary starting point, and D is the set of the days of the week (Mon–Sun). Each count day is divided into parts, defined by P, independently of the shifts. For example, the parts could be: 12 a.m. to 8 a.m.; 8 a.m. to 4 p.m.; and 4 p.m. to 12 p.m. The parts can correspond to operational periods such as shift durations or simply be one hour blocks. Patients are divided into classes in the set K based on their diagnosis and pathway through the hospital. Patients can transition between classes at the beginning of each part of the day. A summary of the general sets and parameters of the model is given in table 3.3.

Table 3.3: General Sets and Parameters

Symbol	Usage
O	The groups of physicians to be rostered separately
R_o	The physicians in roster group o, the set of integers from 1 to $\|R_o\| \equiv \{1, 2, \ldots, \|R_o\|\}$
W_o	Cycle weeks in the roster of roster group o, the set of integers from 1 to $\|W_o\| \equiv \{1, 2, \ldots, \|W_o\|\}$
S_o	Possible shifts for the physicians in the roster group o
L	The groups towards which the workloads of the physicians contribute
$V_{o,r,l}$	Binary indicator that physician r in roster group o contributes their patients to workload group l
C	Count of days in the planning horizon
D	Days of the week
P	Parts of the day
K	Classes of patient

Four parameters are derived from the historical data: the actual number of patients $A_{k,c,p}$ of each class k that are admitted in each part p of each day c; the proportion of admissions $U_{k,c,p,o,s}$ from each class k that each roster group o's shift s is responsible for on a given part p of a given day c; the proportion of patients $T_{l,k,c,p}$ from each class k that is transferred from each workload group l

out of the general medicine department in part p of day c; and the proportion of patients $Q_{k,c,p}$ of each class k that are discharged in each part p of each day c. Proportions are used for discharges instead of counts. If the assignment of patients to workload groups is different in the model than it was when the data was recorded (because the roster is different), then some workload groups may be caring for fewer patients than they were historically. If the historical number of discharges was subtracted from this new, lower, workload, then a team could end up caring for a negative number of patients. Using proportions for discharges removes that possibility.

The proportion of admissions assigned to each shift, $U_{k,c,p,o,s}$, also depends on the particular hospital's rules regarding patient pathways. For example a particular shift might be assigned certain classes of patient, or have a limit to the number of patients it is assigned. As long as the rules for assigning patients to the shifts do not depend on how many patients the physician working the shift is currently caring for, $U_{k,c,p,o,s}$ can be computed using knowledge of the hospital's patient pathways and the admissions and discharges.

Both $Q_{k,c,p}$ and $T_{l,k,c,p}$ quantify the proportion of patients that are removed from the physicians' workloads in each part of each day. Discharges represent patients that no longer need to be in hospital, whereas transfers represent patients that still require care but are moved out of the general medicine department either for a medical reason or because of hospital rules on the number of patients teams can care for. Transfers are assumed to occur at the beginning of each part of the day, just after discharges from the previous part are applied. A summary of the parameters dependent on the historical data is given in table 3.4, in addition methods for calculating these parameters from historical data are given in algorithms 4.1 to 4.3.

The particular rules for a roster define the parameters F_o, B_o, $\mathcal{L}_{o,s,w,d}$, $G_{o,s,w,d}$, $E_{o,s,w,d}$, $\mathcal{R}_o$, and $\mathcal{R}_l$ which represent fixed, banned, linked, following, exclusive and required shifts respectively.

Fixed shifts are shifts that must be worked on a particular day of a particular week. *Banned* shifts are shifts that cannot be worked on a particular day of a particular week. *Linked* shifts are shifts that must be worked if the prior shift they are linked to has been worked, and must not be worked otherwise. For example if a physician works the admission shift they must work the post-acute shift the next day, but they cannot work the post-acute shift otherwise. *Following* shifts are shifts that must be worked if the prior shift they are associated with

Table 3.4: Parameters Dependent on Historical Data

Symbol	Usage
$A_{k,c,p}$	Number of patients admitted on count day c, in part p, of class k
$U_{k,c,p,o,s}$	Proportion of admissions of class k on day c, part p, that shift s of roster group o adds to their workload
$Q_{k,c,p}$	Proportion of patients discharged on count day c, in part p, of class k, after taking into account patients that were transferred.
$T_{l,k,c,p}$	Proportion of patients transferred from workload group l on count day c, in part p, of class k.

has been worked, but can be worked otherwise. For example if a physician works on a particular weekend they must have the next weekend off, but they can have the second weekend off even if they did not work the first one.

Exclusive shifts are shifts cannot be worked if another shift is worked, for instance in the second case study in chapter 4 two 'Long' shifts cannot be worked on consecutive days. In other words the shifts are mutually exclusive. $F_o, B_o, \mathcal{L}_{o,s,w,d}$, and $G_{o,s,w,d}$ are all subsets of the Cartesian product $S_o \times W_o \times D$. In contrast $E_{o,s,w,d}$ is a subset of the power set of this Cartesian product. This is because a shift may be mutually exclusive with a single other shift, a group of other shifts, or both.

Required shifts are shifts that must be performed by an exact number of physicians on a particular day, either by physicians in the same roster group, $\mathcal{R}_o$, or physicians in the same workload group, $\mathcal{R}_l$. $\mathcal{R}_o$ and $\mathcal{R}_l$ are sets of shift, day tuples for which a required number of physicians, $i_{o,s,d}$ and $i_{l,s,d}$, are defined. To be able to define $\mathcal{R}_l$, the physicians in the workload group, l, must either all come from the same roster group or the sets of shifts of roster groups that they come from must have at least one shift in common.

The number of shifts that physicians can be assigned, $y_{o,w,d}$, is usually 1, with the exception being when the physician is working night shifts. When a physician works night shifts their regular duties are sometimes covered by another physician, so they can be assigned a second shift. Note that this method of using a day and week dependent parameter only works if the conditions when physicians can work more than one shift only occurs at a fixed point in the cyclic roster, in the case of a physician covering night shifts this means that the night shifts must be fixed in the cyclic roster. In the problems we study in chapter 4 there is only ever one night tour, which can be fixed in a given place in the cyclic roster without loss of generality. However a more flexible approach would be to define sets of shifts that could or could not be assigned on the same day, regardless of their location in the cyclic roster. These sets could then be used to define exclusive shifts. A summary of the sets and parameters dependent on the rules of the hospital is given in table 3.5.

Note that *Fixed*, *Banned*, *Linked*, and *Following* shifts impose restrictions on a single physician's sequence of shifts, whereas *Required* shifts constrain the shifts that are performed by the roster group as a whole. It is possible to extend the definition of *Fixed*, *Banned*, *Linked*, and *Following* shifts to allow interactions between physicians' sequences of shifts. For example if physician 1 worked shift A on Monday it could mean that physician 2 must work shift B on Tuesday. We have not encountered any rules of this type and therefore do not include any in the current formulation of the model. Also note that within this framework there is not always a unique way of representing a set of rules, however generally a rule lends itself to a particular type of restriction that we have defined here.

Table 3.5: Sets Dependent on Hospital Rules

Symbol	Usage
$F_o \subseteq S_o \times W_o \times D$	A set of shift, week, day tuples used to index which shifts must be performed, and when, for roster group o
$B_o \subseteq S_o \times W_o \times D$	A set of shift, week, day tuples for roster group o, used to indicate the shifts that are banned on each day of each week
$\mathcal{L}_{o,s,w,d} \subseteq S_o \times W_o \times D$	A set of shift, week, day tuples which includes shifts that are linked to future shifts, such that if physician r of roster group o works shift s in week w, day d and $\mathcal{L}_{o,s,w,d} \neq \emptyset$ then they must also work the shift, week, day tuples linked to that shift, $(\hat{s}, \hat{w}, \hat{d}) \in \mathcal{L}_{o,s,w,d}$
$G_{o,s,w,d} \subseteq S_o \times W_o \times D$	A set of shift, week, day tuples which includes shifts that are linked to future shifts, such that if physician r of roster group o works shift s in week w, day d and $G_{o,s,w,d} \neq \varnothing$ then they must also work the shift, week, day tuples following that shift, $(\hat{s}, \hat{w}, \hat{d}) \in G_{o,s,w,d}$, but may work these shifts even if they do not work shift s
$E_{o,s,w,d} \subseteq$ $\mathcal{P}(S_o \times W_o \times D)$	A set of sets of shift, week, day tuples which includes shifts that are linked to other shifts, such that if physician r of roster group o works shift s in week w, day d then for each set $\mathcal{E} \in E_{o,s,w,d}$ they cannot work any of the shift, week, day tuples in the set, $(\hat{s}, \hat{w}, \hat{d}) \in \mathcal{E}$, but may work these shifts if they do not work shift s
$\mathcal{R}_o \subseteq S_o \times D$	A set of shift, day pairs used to index the number of shifts that need to be performed by roster group o
$i_{o,s,d}$	The number of physicians in roster group o that must be assigned shift s on day d
$\mathcal{R}_l \subseteq S_l \times D$	A set of shift, day pairs used to index the number of shifts that need to be performed by workload group l, where S_l is the intersection of the sets of shifts of the roster groups of the physicians in workload group l
$i_{l,s,d}$	The number of physicians in workload group l that must be assigned shift s on day d
$y_{o,w,d}$	The number of shifts that physicians in roster group o can be assigned on day d in week w

3.2.2 Functions

Three functions are defined which are used to map between the planning horizon and the cycles of the roster groups. The first function, $f_1 : O \times R_o \times C \longrightarrow W_o$, returns the week in a given roster group's cycle that a physician is working, given the count day. The second, $f_2 : C \longrightarrow D$, returns the day of the week given the count day. The third, $f_3 : \mathbb{Z} \times O \longrightarrow W_o$, returns the week in a given roster group's cycle, given an integer number of weeks from the start of the planning horizon. A summary of the general functions is given in table 3.6.

Table 3.6: General Functions

Symbol	Usage
$f_1 : O \times R_o \times C \longrightarrow W_o$	A function that returns the week in a given roster group's cycle that a physician is working, given the count day
$f_2 : C \longrightarrow D$	A function that returns the day of the week given the count day
$f_3 : \mathbb{Z} \times O \longrightarrow W_o$	A function that returns the week in a given roster group's cycle given an integer number of weeks from the start of the planning horizon

The function, $j : K \times C \times P \longrightarrow \mathcal{P}(K \times C \times P)$, returns the set of class, day, part tuples $(\hat{k}, \hat{c}, \hat{p})$ that will transition into the given class in the given day and part (k, c, p), or in other words the predecessor class, day, part tuples of a given class, day, part tuple. Note that only tuples $(\hat{k}, \hat{c}, \hat{p})$ where $(\hat{c}, \hat{p})$ is the direct chronological predecessor of (c, p) can transition to (k, c, p). This ensures that patients cannot skip any parts of any days. In addition each $(\hat{k}, \hat{c}, \hat{p})$ is the predecessor of exactly one (k, c, p), in other words each $(\hat{k}, \hat{c}, \hat{p})$ has exactly one successor. This means that all patients of a particular class in a given part of a given day transition to the same class in the subsequent part, if they are not discharged or transferred. A summary of the functions dependent on the hospital's rules are given in table 3.7.

Table 3.7: Functions Dependent on Hospital Rules

Symbol	Usage
$j : K \times C \times P \longrightarrow \mathcal{P}(K \times C \times P)$	A function that describes how a patients' class changes over time. Returns the set of class, week, part tuples that transition directly to the given class, week, part tuple.

3.2.3 Decision Variables

Binary decision variables, $x_{o,s,w,d}$, are used to determine whether the roster for roster group o includes a particular shift on a given day of a given week. Given an assignment of shifts and patient admission data the variable $N_{l,k,c,p}$ measures the number of patients of each class each workload group is caring for in each part of each count day. The variables $M^{+}_{c,p}$, $M^{-}_{c,p}$ and $\bar{M}$ are then used to determine the largest difference in workloads between the workload groups at any point in the planing horizon. A summary of the decision variables used in the model is given in table 3.8.

Table 3.8: Decision Variables

Symbol	Usage
$x_{o,s,w,d}$	Whether shift s is performed on week w, day d, by roster group o (1 if it is, 0 otherwise), $\forall o \in O, s \in S_o, w \in W_o, d \in D$
$N_{l,k,c,p}$	Number of patients of class k, workload group l is attending to at the end of part p on count day c, $\forall\, l \in L, k \in K, c \in C, p \in P$
$M^{+}_{c,p}$	An upper bound on the workload of the workload group with the largest workload in part p on count day c, $\forall c \in C, p \in P$
$M^{-}_{c,p}$	A lower bound on the workload of the workload group with the smallest workload in part p on count day c, $\forall c \in C, p \in P$
$\bar{M}$	The largest difference in workloads between the workload groups over the entire planning horizon

3.2.4 Objective Function

The objective function minimises the largest difference in workloads between the workload groups over the planning horizon, and is shown below in (3.1):

$$\min \ \bar{M}. \tag{3.1}$$

The workloads are not normalised by the number of physicians in the workload groups, because in the instances we have encountered all of the workload groups have had the same number of physicians making normalisation unnecessary. The model could be generalised by using normalised workloads by changing the definition of $N_{l,k,c,p}$ to be the number of patients per physician, and dividing the corresponding constraints by the number of physicians in the workload group.

3.2.5 Constraints

Constraints described in this section ensure that the roster obeys all of the rules. The earlier definition of the appropriate sets and functions allows the constraints to be expressed concisely. The constraints defined in (3.2) mean all fixed shifts are worked and those in (3.3) prevent the banned shifts from being worked. The constraints in (3.4) make sure shifts that are linked to the shift s on day d, week w are worked if and only if that shift is worked. The constraints in (3.5) make sure shifts that must follow the shift s on day d, week w are worked if that shift is worked, but can also be worked if it is not. The constraints in (3.6) make sure the sets of shifts that are mutually exclusive with the shift s on day d, week w are not worked if that shift is worked. The constraints in (3.7) and (3.8) make sure that the correct number of the required shifts are worked for each roster and workload group. Finally the constraints in (3.9) ensure that the number of shifts assigned is less than or equal to the number allowed. Constraints (3.2)–(3.9) are shown below:

$$
\begin{aligned}
x_{o,s,w,d} &= 1 && \forall o \in O, (s,w,d) \in F_o, && (3.2)\\
x_{o,s,w,d} &= 0 && \forall o \in O, (s,w,d) \in B_o, && (3.3)\\
x_{o,s,w,d} &= x_{o,\hat{s},\hat{w},\hat{d}} && \forall o \in O, (\hat{s},\hat{w},\hat{d}) \in \mathcal{L}_{o,s,w,d}, && (3.4)\\
x_{o,s,w,d} &\le x_{o,\hat{s},\hat{w},\hat{d}} && \forall o \in O, (\hat{s},\hat{w},\hat{d}) \in G_{o,s,w,d,} && (3.5)\\
x_{o,s,w,d} + \sum_{(\hat{s},\hat{w},\hat{d})\in\mathcal{E}} x_{o,\hat{s},\hat{w},\hat{d}} &\le 1 && \forall o \in O, \mathcal{E} \in E_{o,s,w,d,} && (3.6)\\
\sum_{w\in W_o} x_{o,s,w,d} &= i_{o,s,d} && \forall o \in O, (s,d) \in \mathcal{R}_{o,} && (3.7)\\
\sum_{o\in O, w\in W_o} V_{o,w,l} x_{o,s,w,d} &= i_{l,s,d} && \forall l \in L, (s,d) \in \mathcal{R}_{l,} && (3.8)\\
\sum_{s\in S_o} x_{o,s,w,d} &\le y_{o,w,d} && \forall o \in O, w \in W, d \in D. && (3.9)
\end{aligned}
$$

Constraint (3.10) is the patient balance equation, which ensures every patient is assigned to a workload group. The number of patients each workload group is caring for is made up of two parts. The first part is the sum of the patients they were previously caring for multiplied by one minus the proportion transferred. The function $j(k,c,p)$ returns the classes, count days and parts of patients that will become a patient of class k in part p on count day c. This enables the model to track patients through the parts of the day and from one day to the next.

$$N_{l,k,c,p} = \left(1 - Q_{k,c,p}\right) \Bigg(\underbrace{\sum_{(\hat{k},\hat{c},\hat{p}) \in j(k,c,p)} \left(\left(1 - T_{l,\hat{k},\hat{c},\hat{p}}\right) N_{l,\hat{k},\hat{c},\hat{p}} \right)}_{\text{Previous patients retained}} \tag{3.10}$$

$$+ \underbrace{\sum_{o \in O, s \in S_o, r \in R_o} \left(A_{k,c,p} U_{k,c,p,o,s} x_{o,s,f_1(o,r,c),f_2(c)} V_{o,r,l} \right) \Bigg)}_{\text{New patients admitted}}$$

$$\forall l \in L, k \in K, c \in C, p \in P.$$

The second part is the sum of patients that are admitted by physicians who contribute to that workload group. The week index for the shift variable is calculated by using f_1, and the day is found using f_2. Note that these functions assume a fixed ordering of the physicians within each roster group.

The two parts are then multiplied by one minus the proportion of patients that are discharged, to get the number of patients that remain. Note that $T_{l,k,c,p}$ is only applied to patients that are carried over from previous parts, whereas $Q_{k,c,p}$ is also applied to patients that are received in that part.

Constraints (3.11) and (3.12) define an upper bound for the maximum and a lower bound for the minimum workload for each part of each day respectively. The largest difference in workloads is then defined by the constraints in (3.13). Note that the minimisation objective will ensure that the constraints are binding for the part where the largest difference in workloads occurs.

$$M^{+}_{c,p} \geq \sum_{k \in K} N_{l,k,c,p} \qquad \forall c \in C, p \in P, l \in L, \tag{3.11}$$

$$M^{-}_{c,p} \leq \sum_{k \in K} N_{l,k,c,p} \qquad \forall c \in C, p \in P, l \in L, \tag{3.12}$$

$$\bar{M} \geq M^{+}_{c,p} - M^{-}_{c,p} \qquad \forall c \in C, p \in P. \tag{3.13}$$

3.2.6 Uncertain Admissions

Until this point it has been assumed that the number of patients admitted in each part is certain and determined by $A_{k,c,p}$. Likewise the other parameters determined by the data $U_{k,c,p,o,s}$, $Q_{k,c,p}$, and $T_{l,\hat{k},\hat{c},\hat{p}}$ have also been assumed to be certain.

In order to incorporate the uncertainty inherent in the system the number of patients admitted in each part can be estimated by modelling it using a parametric distribution, such as the Poisson distribution, or using the empirical distribution. The length of stays of each of the patients can also be estimated in a similar manner. However, translating these processes into the number of patients that each physician is caring for in each part of each day would require representing the patients' pathways as a type of queue system. This may be possible for simple pathways, however in general this is not always possible.

We can instead assume a situation where there are a number of possible scenarios for each of $A_{k,c,p}$, $U_{k,c,p,o,s}$, $Q_{k,c,p}$, and $T_{l,\hat{k},\hat{c},\hat{p}}$. The scenarios could either be proposed by the healthcare organisation, or generated from available historical data, as we have done in section 4.1.4. The aim in this new situation is to find a roster that takes into account all of the possible scenarios, rather than a single realisation of admissions and discharges. Two methods for aggregating the objective function across scenarios are presented later in this section. First, however, the implications on the decision variables and constraints are discussed.

Suppose then that there is a set of scenarios Z, and for each $z \in Z$ the values of $A^{(z)}_{k,c,p}$, $U^{(z)}_{k,c,p,o,s}$, $Q^{(z)}_{k,c,p}$, and $T^{(z)}_{l,\hat{k},\hat{c},\hat{p}}$ are known. The decision variables that relate to the number of patients being attended to by each workload group, and the variables that measure the differences between groups, also need to be extended to account for the scenarios, becoming $N^{(z)}_{l,\hat{k},\hat{c},\hat{p}}$, $M^{+(z)}_{c,p}$, $M^{-(z)}_{c,p}$, and $\bar{M}^{(z)}_{c,p}$.

The constraints (3.10)–(3.13) are replaced with the following. Note that the parameters and decision variables are now indexed by scenario, except for $x_{o,s,f_1(o,r,c),f_2(c)}$, as we want one roster for all the scenarios, and $V_{o,r,l}$, since the physicians contribute to the same workload groups in all the scenarios.

$$\begin{aligned} N^{(z)}_{l,k,c,p} = \left(1 - Q^{(z)}_{k,c,p}\right) \Bigg(& \sum_{(\hat{k},\hat{c},\hat{p}) \in j(k,c,p)} \left(\left(1 - T^{(z)}_{l,\hat{k},\hat{c},\hat{p}}\right) N^{(z)}_{l,\hat{k},\hat{c},\hat{p}}\right) \\ & + \sum_{o \in O, s \in S_o, r \in R_o} \left(A^{(z)}_{k,c,p} U^{(z)}_{k,c,p,o,s} x_{o,s,f_1(o,r,c),f_2(c)} V_{o,r,l}\right)\Bigg) \\ & \forall l \in L, k \in K, c \in C, p \in P, z \in Z, \end{aligned} \tag{3.10}$$

$$M^{+(z)}_{c,p} \geq \sum_{k \in K} N^{(z)}_{l,k,c,p} \qquad \forall c \in C, p \in P, l \in L, z \in Z, \tag{3.11}$$

$$M^{-(z)}_{c,p} \leq \sum_{k \in K} N^{(z)}_{l,k,c,p} \qquad \forall c \in C, p \in P, l \in L, z \in Z, \tag{3.12}$$

$$\bar{M}^{(z)} \geq M^{+(z)}_{c,p} - M^{-(z)}_{c,p} \qquad \forall c \in C, p \in P, z \in Z. \tag{3.13}$$

In the case of a single scenario the objective function was to minimise the largest difference in workloads in any part of the planning horizon. We consider two ways of extending this objective to multiple scenarios. The first method, referred to as the 'Mean' objective, is to minimise the mean of the largest difference in workloads across all the scenarios, shown below:

$$\min \frac{1}{|Z|} \sum_{z \in Z} \bar{M}^{(z)}. \tag{3.14}$$

If, instead, we assume that the healthcare organisation is risk averse and is more concerned with the outcomes in the worst scenarios, then we could minimise the mean of the largest difference in workloads in the worst 5% of scenarios. This particular type of risk aversion is called conditional value at risk (CVaR) or expected shortfall. Rockafellar and Uryasev 2000 show how to express this within a LP by introducing additional decision variables α, and $\gamma^{(z)}$. If we assume a minimisation objective which we want to minimise over the worst (largest) $(1-\beta)$% of scenarios, then α represents the β percentile of the objective function over the scenarios. The other additional variables, $\gamma^{(z)}$, represent the amount that the largest difference in workloads in scenario z, is above α. If the largest difference in workloads in scenario z is less than α then $\gamma^{(z)}$ is 0. The objective of minimising the largest difference in workloads in the worst $(1-\beta)$% of scenarios, referred to as the CVaR @ β% objective, can then be formulated as:

$$\min \ \alpha + \frac{1}{|Z|(1-\beta)} \sum_{z \in Z} \gamma^{(z)}. \tag{3.15}$$

With the following additional constraints to ensure the definitions of α, and $\gamma^{(z)}$:

$$\gamma^{(z)} \geq \bar{M}^{(z)} - \alpha \qquad \forall z \in Z. \tag{3.16}$$

The new problem, where the uncertainty in admissions and discharges is represented in scenarios is similar to the sample average approximation (SAA) problem in stochastic programming, where the samples are replaced by scenarios.

The SAA problem approximates a stochastic programme, where one or more parameters are assumed to come from a known probability distribution(s) and the objective is to optimise the expectation of a function of the decision variables and the random parameters, by taking a sample from the distribution(s) and optimising the average of the objective function applied to this sample. If a sampling procedure is used to generate the scenarios then algorithms for SAA problems such as in Kleywegt, Shapiro, and Homem-de-Mello (2002), which involves successively solving the problem with more and larger samples until a convergence criteria is met, can be used.

For clarity the entire model is given in full below:

$$
\begin{array}{rrll}
\min & \frac{1}{|Z|}\sum_{z\in Z}\bar{M}^{(z)} & & (3.14)\\
\text{s.t.} & x_{o,s,w,d} = 1 & \forall o \in O, (s,w,d) \in F_o, & (3.2)\\
 & x_{o,s,w,d} = 0 & \forall o \in O, (s,w,d) \in B_o, & (3.3)\\
 & x_{o,s,w,d} = x_{o,\hat{s},\hat{w},\hat{d}} & \forall o \in O, (\hat{s},\hat{w},\hat{d}) \in \mathcal{L}_{o,s,w,d}, & (3.4)\\
 & x_{o,s,w,d} \leq x_{o,\hat{s},\hat{w},\hat{d}} & \forall o \in O, (\hat{s},\hat{w},\hat{d}) \in G_{o,s,w,d}, & (3.5)\\
 & x_{o,s,w,d} + \sum_{(\hat{s},\hat{w},\hat{d})\in\mathcal{E}} x_{o,\hat{s},\hat{w},\hat{d}} \leq 1 & \forall o \in O, \mathcal{E} \in E_{o,s,w,d}, & (3.6)\\
 & \sum_{w\in W_o} x_{o,s,w,d} = i_{o,s,d} & \forall o \in O, (s,d) \in \mathcal{R}_o, & (3.7)\\
 & \sum_{o\in O, w\in W_o} V_{o,w,l}x_{o,s,w,d} = i_{l,s,d} & \forall l \in L, (s,d) \in \mathcal{R}_l, & (3.8)\\
 & \sum_{s\in S_o} x_{o,s,w,d} = 1 & \forall o \in O, w \in W, d \in D, & (3.9)
\end{array}
$$

$$
\begin{aligned}
N^{(z)}_{l,k,c,p} = \left(1 - Q^{(z)}_{k,c,p}\right)\Bigg(& \sum_{(\hat{k},\hat{c},\hat{p})\in j(k,c,p)} \left(\left(1 - T^{(z)}_{l,\hat{k},\hat{c},\hat{p}}\right) N^{(z)}_{l,\hat{k},\hat{c},\hat{p}}\right) \\
& + \sum_{o\in O, s\in S_o, r\in R_o} \left(A^{(z)}_{k,c,p} U^{(z)}_{k,c,p,o,s} x_{o,s,f_1(o,r,c),f_2(c)} V_{o,r,l}\right)\Bigg) \\
& \forall l \in L, k \in K, c \in C, p \in P, z \in Z,
\end{aligned}
\qquad (3.10)
$$

$$M^{+(z)}_{c,p} \geq \sum_{k \in K} N^{(z)}_{l,k,c,p} \qquad \forall c \in C, p \in P, l \in L, z \in Z, \tag{3.11}$$

$$M^{-(z)}_{c,p} \leq \sum_{k \in K} N^{(z)}_{l,k,c,p} \qquad \forall c \in C, p \in P, l \in L, z \in Z, \tag{3.12}$$

$$\bar{M}^{(z)} \geq M^{+(z)}_{c,p} - M^{-(z)}_{c,p} \qquad \forall c \in C, p \in P, z \in Z, \tag{3.13}$$

$$N^{(z)}, M^{+(z)}, M^{-(z)}, \bar{M} \geq 0, \tag{3.17}$$

$$x \in \{0, 1\}. \tag{3.18}$$

3.3 Benders' Decomposition

With the formulation of the optimisation problem defined in section 3.2 we apply Benders' decomposition with the aims of showing that the sub-problems (SPs) can be solved by a simple calculation, and the solution to this calculation gives cuts that are equivalent to those found by a LP proposed in Fischetti, Salvagnin, and Zanette (2010). Benders' decomposition (Benders 1962) breaks a problem into a restricted master problem (RMP) and one or more SPs. The SPs are represented in the RMP by a set of constraints. This set of constraints usually starts small, sometimes as the empty set, and more constraints are added as the result of solving the SPs.

To simplify the notation in this section we re-write constraints (3.2)–(3.9), which ensure that the roster conforms to all of the rules, as $\mathcal{A}x = \mathcal{b}$. In addition the patient balance equations in (3.10) are re-written as $N^{(z)}_{l,k,c,p} = \mathcal{B}^{(z)}_{l,k,c,p}x$, where $\mathcal{B}^{(z)}_{l,k,c,p}$ is a vector of length $|O||S||W||D|$ that incorporates the values of the parameters $A^{(z)}_{k,c,p}$, $U^{(z)}_{k,c,p,o,s}$, and $V_{o,r,l}$ directly, and $Q^{(z)}_{k,c,p}$, and $T^{(z)}_{l,\hat{k},\hat{c},\hat{p}}$ indirectly through a substitution of $N^{(z)}_{l,\hat{k},\hat{c},\hat{p}}$. It is also important to note here that it is assumed that $\mathcal{B}^{(z)}_{l,k,c,p}x$ is always positive as it models a patient workload, and therefore the variables $N, M^{+}, M^{-}, \bar{M}$ are all constrained to be non-negative.

The set $X = \{x | \mathcal{A}x = \mathcal{b}, x \in \{0, 1\}\}$ represents the set of feasible rosters and $\bar{x}$ is a given value of the binary variables belonging to the set X, in other words a given feasible roster. The problem in the previous section, defined by (3.1)–(3.13), can then be re-expressed as:

$$\min_{\bar{x} \in X} \left\{ \min_{N^{(z)}_{l,k,c,p}, M^{+(z)}_{c,p}, M^{-(z)}_{c,p}, \bar{M}^{(z)}} \{\mathcal{Q}(N^{(z)}_{l,k,c,p}, M^{+(z)}_{c,p}, M^{-(z)}_{c,p}, \bar{M}^{(z)}, \bar{x})\} \right\}, \tag{3.19}$$

where $\mathcal{Q}(N^{(z)}_{l,k,c,p}, M^{+(z)}_{c,p}, M^{-(z)}_{c,p}, \bar{M}^{(z)}, \bar{x}\})$ is the optimal value of the problem:

$$\min \quad \sum_{z \in Z} (\bar{M}^{(z)}), \tag{3.20}$$

$$\text{s.t.} \quad N^{(z)}_{l,k,c,p} = \mathcal{B}^{(z)}_{l,k,c,p}\bar{x}, \qquad \forall l \in L, k \in K, c \in C, p \in P, z \in Z, \tag{3.10$'$}$$

$$M^{+(z)}_{c,p} - \sum_{k \in K} (N^{(z)}_{l,k,c,p}) \geq 0, \qquad \forall l \in L, c \in C, p \in P, z \in Z, \tag{3.11}$$

$$\sum_{k \in K} (N^{(z)}_{l,k,c,p}) - M^{-(z)}_{c,p} \geq 0, \qquad \forall l \in L, c \in C, p \in P, z \in Z, \tag{3.12}$$

$$\bar{M}^{(z)} - (M^{+(z)}_{c,p} - M^{-(z)}_{c,p}) \geq 0, \qquad \forall c \in C, p \in P, z \in Z, \tag{3.13}$$

$$N, M^{+}, M^{-}, \bar{M} \geq 0. \tag{3.17}$$

Given dual variables $\pi^1, \pi^2, \pi^3, \pi^4$ that correspond to constraints (3.10′)–(3.13), the dual of this problem is (a description of the dual transformation is given in appendix A):

$$\max \quad \sum_{z \in Z} \left((\pi^{(1,z)})^{\intercal} (\mathcal{B}^{(z)} \bar{x}) \right), \tag{3.21}$$

$$\text{s.t.} \quad \sum_{k \in K} (\pi^{(1,z)}_{l,k,c,p}) - \pi^{(2,z)}_{l,c,p} + \pi^{(3,z)}_{l,c,p} \leq 0, \quad \forall l \in L, c \in C, p \in P, z \in Z, \tag{3.22}$$

$$\sum_{l \in L} (\pi^{(2,z)}_{l,c,p}) - \pi^{(4,z)}_{c,p} \leq 0, \quad \forall c \in C, p \in P, z \in Z, \tag{3.23}$$

$$\pi^{(4,z)}_{c,p} - \sum_{l \in L} (\pi^{(3,z)}_{l,c,p}) \leq 0, \quad \forall c \in C, p \in P, z \in Z, \tag{3.24}$$

$$\sum_{c \in C, p \in P} (\pi^{(4,z)}_{c,p}) \leq 1, \quad \forall z \in Z, \tag{3.25}$$

$$\pi^1 \text{ free}, \pi^2, \pi^3, \pi^4 \geq 0. \tag{3.26}$$

The dual problem is always feasible and bounded (see appendix A). Therefore given the set of vertices of the feasible polytope $\mathcal{E}$, for the dual problem, there is an optimal solution at one of these vertices π_e, $e \in \mathcal{E}$ for each integer solution $\bar{x}$. The full problem in (3.19) can therefore be restated as:

$$\min \quad \eta, \tag{3.27}$$

$$\text{s.t.} \quad \mathcal{A}x = b, \tag{3.28}$$

$$\eta \geq (\pi^1_e)^{\intercal}(\mathcal{B}x), \quad \forall e \in \mathcal{E}, \tag{3.29}$$

$$x \in \{0, 1\}. \tag{3.30}$$

It is usually not practical to include all of the constraints (3.29). Instead two main approaches are used to solve the problem. The first, introduced by Benders (1962), solves a relaxation of the problem that only includes a subset of the constraints (3.29), the RMP. Then the solution to the RMP is passed to the SPs and cuts are generated and added to the RMP. This process is repeated until an optimal solution is found.

The second approach also begins with an RMP. However instead of solving the RMP to optimality, solutions found during the branch and bound process (both integer and fractional) are used to generate cuts which are added to the RMP within a single branch and bound tree.

Both methods require optimality cuts of the form $\eta \geq (\pi^{(1)})^\intercal(\mathcal{B}x)$, for which we need the solution to the dual SP. This leads to the following proposition:

Proposition 3.1. *Given $\mathcal{B}$ and a solution to the RMP, $\bar{x}$, the optimality cut required for Benders' decomposition can be expressed as:*

$$\eta \geq \sum_{z\in Z}\left(\sum_{k\in K}(\mathcal{B}^{(z)}_{l^{+(z)},k,\bar{c}^{(z)},\bar{p}^{(z)}}x) - \sum_{k\in K}(\mathcal{B}^{(z)}_{l^{-(z)},k,\bar{c}^{(z)},\bar{p}^{(z)}}x)\right), \tag{3.31}$$

where

$$\bar{c}^{(z)},\bar{p}^{(z)} = \arg\max_{c\in C,p\in P}\left(\max_{l\in L}\left(\sum_{k\in K}(\mathcal{B}^{(z)}_{l,k,c,p}\bar{x})\right) - \min_{l\in L}\left(\sum_{k\in K}(\mathcal{B}^{(z)}_{l,k,c,p}\bar{x})\right)\right), \tag{3.32}$$

$$l^{+(z)} = \arg\max_{l\in L}\left(\sum_{k\in K}(\mathcal{B}^{(z)}_{l,k,\bar{c}^{(z)},\bar{p}^{(z)}}\bar{x})\right), \tag{3.33}$$

$$l^{-(z)} = \arg\min_{l\in L}\left(\sum_{k\in K}(\mathcal{B}^{(z)}_{l,k,\bar{c}^{(z)},\bar{p}^{(z)}}\bar{x})\right). \tag{3.34}$$

Proof. We will prove this proposition by finding feasible solutions to both the primal and the dual and showing that the objective function values are equal, and therefore, by the Strong Duality Theorem, that the solutions are optimal.

We begin by noting that the primal SP can be further decomposed because there is no interaction between any of the scenarios, giving the following problem for a single scenario, with the (z) superscript dropped and rearranged slightly from earlier:

$$\begin{aligned}
\min \quad & \bar{M} && && (3.35)\\
\text{s.t.} \quad N_{l,k,c,p} &= \mathcal{B}_{l,k,c,p}\bar{x}, && \forall l \in L, k \in K, c \in C, p \in P, && (3.10'')\\
M^{+}_{c,p} &\geq \sum_{k \in K}(N_{l,k,c,p}), && \forall l \in L, c \in C, p \in P, && (3.11')\\
\sum_{k \in K}(N_{l,k,c,p}) &\geq M^{-}_{c,p}, && \forall l \in L, c \in C, p \in P, && (3.12')\\
\bar{M} &\geq M^{+}_{c,p} - M^{-}_{c,p}, && \forall c \in C, p \in P, && (3.13')\\
N, M^{+}, M^{-}, \bar{M} &\geq 0. && && (3.17)
\end{aligned}$$

The $N_{l,k,c,p}$ variables are specified exactly by $N_{l,k,c,p} = \mathcal{B}_{l,k,c,p}\bar{x}$, which can be substituted into the remaining constraints:

$$\begin{aligned}
\min \quad & \bar{M} && && (3.35)\\
\text{s.t.} \quad M^{+}_{c,p} &\geq \sum_{k \in K}(\mathcal{B}_{l,k,c,p}\bar{x}), && \forall l \in L, c \in C, p \in P, && (3.11'')\\
\sum_{k \in K}(\mathcal{B}_{l,k,c,p}\bar{x}) &\geq M^{-}_{c,p}, && \forall l \in L, c \in C, p \in P, && (3.12'')\\
\bar{M} &\geq M^{+}_{c,p} - M^{-}_{c,p}, && \forall c \in C, p \in P, && (3.13'')\\
M^{+}, M^{-}, \bar{M} &\geq 0. && && (3.17)
\end{aligned}$$

The decomposed and substituted problem has a feasible solution of:

$$\bar{M} = \max_{c \in C, p \in P}\left(\max_{l \in L}\left(\sum_{k \in K}(\mathcal{B}^{(z)}_{l,k,c,p}\bar{x})\right) - \min_{l \in L}\left(\sum_{k \in K}(\mathcal{B}^{(z)}_{l,k,c,p}\bar{x})\right)\right),$$

$$M^{+}_{c,p} = \max_{l \in L}\left(\sum_{k \in K}(\mathcal{B}^{(z)}_{l,k,\bar{c}^{(z)},\bar{p}^{(z)}}\bar{x})\right) \ \forall c \in C, p \in P,$$

$$M^{-}_{c,p} = \min_{l \in L}\left(\sum_{k \in K}(\mathcal{B}^{(z)}_{l,k,\bar{c}^{(z)},\bar{p}^{(z)}}\bar{x})\right) \ \forall c \in C, p \in P,$$

with an objective function value of

$$\max_{c \in C, p \in P}\left(\max_{l \in L}\left(\sum_{k \in K}(\mathcal{B}^{(z)}_{l,k,c,p}\bar{x})\right) - \min_{l \in L}\left(\sum_{k \in K}(\mathcal{B}^{(z)}_{l,k,c,p}\bar{x})\right)\right).$$

The solution is feasible as $M^{+}_{c,p}$ and $M^{-}_{c,p}$ are equal to the maximum and minimum values of the right hand sides of constraints (3.11″) and (3.12″) respectively. $\bar{M}$ is then equal to the maximum value of the right hand side of constraint (3.13″).

This solution effectively computes the maximum difference in workloads for the given roster by finding the part of the day on which the difference between the maximum and minimum workloads is largest.

We now consider the dual problem, beginning with the scenario-decomposed version of the original dual:

$$\begin{aligned}
&\max && (\pi^{(1)})^\intercal(\mathcal{B}\bar{x}), && \\
&\text{s.t.} && \sum_{k\in K}(\pi^{(1)}_{l,k,c,p}) - \pi^{(2)}_{l,c,p} + \pi^{(3)}_{l,c,p} \leq 0, && \forall l \in L, c \in C, p \in P, \\
& && \sum_{l\in L}(\pi^{(2)}_{l,c,p}) - \pi^{(4)}_{c,p} \leq 0, && \forall c \in C, p \in P, \\
& && \pi^{(4)}_{c,p} - \sum_{l\in L}(\pi^{(3)}_{l,c,p}) \leq 0, && \forall c \in C, p \in P, \\
& && \sum_{c\in C, p\in P}(\pi^{(4)}_{c,p}) \leq 1, && \\
& && \pi^1 \text{ free}, \pi^2, \pi^3, \pi^4 \geq 0. &&
\end{aligned}$$

Because of the maximisation objective, and the assumption that $\mathcal{B}\bar{x}$ is always positive, the first constraint can be considered an equality constraint and the following substitution can be made

$$\sum_{k\in K}(\pi^{(1)}_{l,k,c,p}) = \pi^{(2)}_{l,c,p} - \pi^{(3)}_{l,c,p}.$$

Using this substitution the dual of the decomposed SP is given below, note that because of the substitution the objective function now requires a sum over K, and the $\bullet$ subscript is used as a place holder to represent a vector when only one of the subscripts is specified:

$$\max \quad (\pi^2 - \pi^3)^\intercal \left(\sum_{k\in K}(\mathcal{B}_{\bullet k \bullet\bullet}\bar{x})\right) \tag{3.21$'$}$$

$$\text{s.t.} \quad \pi^4_{c,p} \geq \sum_{l\in L}(\pi^2_{l,c,p}), \qquad \forall c \in C, p \in P, \tag{3.23$'$}$$

$$\pi^4_{c,p} \leq \sum_{l\in L}(\pi^3_{l,c,p}), \qquad \forall c \in C, p \in P, \tag{3.24$'$}$$

$$\sum_{c\in C, p\in P}(\pi^4_{c,p}) \leq 1, \tag{3.25$'$}$$

$$\pi^2, \pi^3, \pi^4 \geq 0. \tag{3.26$'$}$$

The decomposed and substituted dual problem has a feasible solution

$$\pi^2_{l^+,\bar{c},\bar{p}} = 1, \pi^3_{l^-,\bar{c},\bar{p}} = 1, \pi^4_{\bar{c},\bar{p}} = 1,$$

and all other variables are 0, and $\bar{c}$, $\bar{p}$, l^+, and l^- are defined as in (3.32)–(3.34). This solution also has an objective function value of

$$\max_{c\in C, p\in P}\left(\max_{l\in L}\left(\sum_{k\in K}(\mathcal{B}^{(z)}_{l,k,c,p}\bar{x})\right) - \min_{l\in L}\left(\sum_{k\in K}(\mathcal{B}^{(z)}_{l,k,c,p}\bar{x})\right)\right).$$

Therefore we have feasible solutions to both the primal and the dual problems with the same objective function value and by the Strong Duality Theorem the two solutions solve the primal and dual SPs respectively. These solutions to the decomposed dual SPs are multiplied by their objective coefficients and aggregated to obtain the optimality cut for Benders' decomposition as follows:

$$\eta \geq \sum_{z\in Z}\left(\sum_{k\in K}(\mathcal{B}^{(z)}_{l^{+(z)},k,\bar{c}^{(z)},\bar{p}^{(z)}}x) - \sum_{k\in K}(\mathcal{B}^{(z)}_{l^{-(z)},k,\bar{c}^{(z)},\bar{p}^{(z)}}x)\right). \tag{3.31}$$

□

Proposition 3.1 means that, instead of solving an LP for the dual SP, we can compute $\bar{c}$, $\bar{p}$, l^+, and l^- for each scenario, z, to get the indices of the non-zero dual variables. The optimality cut can then be formed.

The cuts in (3.31) are known as aggregated cuts because of the sum of coefficients over all the scenarios. With a small reformulation of the RMP, given below, one cut can be added for each scenario. These are known as disaggregated cuts, and in situations where they can be applied they have been shown to be an improvement over aggregated cuts. Given that the set of vertices of the feasible polytope for each scenario's sub-problem are represented by $\mathcal{E}^{(z)}$. Each vertex of $\mathcal{E}^{(z)}$ is represented by $e^{(z)}$, and the dual solution at each vertex is $\pi^{(z)}_{e^{(z)}}$. Using the decomposed and substituted version of the dual SP the master problem (MP) in (3.27)–(3.30) can be restated as:

$$\begin{aligned}
\min \quad & \bar{\eta} && (3.36)\\
\text{s.t.} \quad & \mathscr{A}x = \mathscr{b}, && (3.37)\\
& \bar{\eta} \geq \sum_{z\in Z}\eta^{(z)}, && (3.38)\\
& \eta^{(z)} \geq (\pi^{(2,z)}_{e^{(z)}})^\intercal(\mathcal{B}^{(z)}_{\bullet k\bullet\bullet}x) - (\pi^{(3,z)}_{e^{(z)}})^\intercal(\mathcal{B}^{(z)}_{\bullet k\bullet\bullet}x), \quad \forall e^{(z)} \in \mathcal{E}^{(z)}, z \in Z, && (3.39)\\
& x \in \{0,1\}. && (3.40)
\end{aligned}$$

To solve this problem constraint (3.39) is initially ignored and a value of x is found that satisfies $\mathscr{A}x = \mathscr{b}$. This x is used to add cuts, to the MP, of the form:

$$\eta^{(z)} \geq \sum_{k \in K} (\mathcal{B}^{(z)}_{l^{+(z)},k,\bar{c}^{(z)},\bar{p}^{(z)}} x) - \sum_{k \in K} (\mathcal{B}^{(z)}_{l^{-(z)},k,\bar{c}^{(z)},\bar{p}^{(z)}} x), \tag{3.41}$$

where $\bar{c}$, $\bar{p}$, l^{+}, and l^{-} are are determined as before, using (3.32)–(3.33). The RMP is then solved again with the new cuts added, and another x is found. This is again used to add more cuts to the RMP. This process is repeated until a solution of the RMP is found where the objective of the RMP is equal to (or within a required tolerance of) the sum of the right hand sides of the cuts that are generated from that solution.

In addition, Fischetti, Salvagnin, and Zanette (2010) proposes an LP (given below) to find cuts for Benders' decomposition. The LP selects a cut which corresponds to a minimal infeasible subsystem of an extended primal SP, given a solution $\bar{x}, \eta$ to the RMP.

$$\max \quad (\pi^2 - \pi^3)^{\intercal} \left(\sum_{k \in K} (\mathcal{B}_{\bullet k \bullet \bullet} \bar{x}) \right) - \pi^0 \eta \tag{3.42}$$

$$\text{s.t.} \quad \pi^4_{c,p} \geq \sum_{l \in L} (\pi^2_{l,c,p}), \qquad \forall c \in C, p \in P, \tag{3.43}$$

$$\pi^4_{c,p} \leq \sum_{l \in L} (\pi^3_{l,c,p}), \qquad \forall c \in C, p \in P, \tag{3.44}$$

$$\sum_{c \in C, p \in P} (\pi^4_{c,p}) \leq \pi^0, \tag{3.45}$$

$$\sum_{l \in L, c \in C, p \in P} \left(\pi^2_{l,c,p} + \pi^3_{l,c,p} \right) + \omega^0 \pi^0 \leq 1, \tag{3.46}$$

$$\pi^0, \pi^2, \pi^3, \pi^4 \geq 0. \tag{3.47}$$

Proposition 3.2. *If the LP proposed by Fischetti, Salvagnin, and Zanette (2010) produces a cut, it is the same cut as that identified in proposition 3.1.*

Proof. Suppose we start with the feasible solution $\pi = 0$. In order to improve the objective function from this point we need to find d, p, l^1, and l^2 such that $\pi^2_{l^1,d,p} \left(\sum_{k \in K} (\mathcal{B}_{l^1,k,d,p} \bar{x}) \right) - \pi^3_{l^2,d,p} \left(\sum_{k \in K} (\mathcal{B}_{l^2,k,d,p} \bar{x}) \right) - \pi^0 \eta > 0$.

If no such d, p, l^1, and l^2 exist the optimal solution is $\pi = 0$, or if η is negative then $\pi^0 = 1/\omega^0$, and no cut can be formed from the sub-problem.

If such d, p, l^1, and l^2 exist then they are chosen such that $\left(\sum_{k\in K}(\mathcal{B}_{l^+,k,\bar{c},\bar{p}}\bar{x})\right) - \left(\sum_{k\in K}(\mathcal{B}_{l^-,k,\bar{c},\bar{p}}\bar{x})\right)$ is maximised, in the same manner as in (3.32) – (3.34). If we then increase both $\pi^2_{l^+,\bar{c},\bar{p}}$ and $\pi^3_{l^-,\bar{c},\bar{p}}$ by ϵ, then $\pi^4_{\bar{c},\bar{p}}$ increases by ϵ due to constraint (3.43), and π^0 therefore increases by ϵ as well because of constraint (3.45). This process has increased $\pi^2_{l^+,\bar{c},\bar{p}}$, $\pi^3_{l^-,\bar{c},\bar{p}}$ and π^0 by ϵ, therefore the left hand side of (3.46) has increased by $(2 + \omega^0)\epsilon$.

Since $\bar{c}$, $\bar{p}$, l^+, and l^- are chosen to maximise

$$\left(\sum_{k\in K}(\mathcal{B}_{l^+,k,\bar{c},\bar{p}}\bar{x})\right) - \left(\sum_{k\in K}(\mathcal{B}_{l^-,k,\bar{c},\bar{p}}\bar{x})\right)$$

increasing any π^2 or π^3 other than $\pi^2_{l^+,\bar{c},\bar{p}}$ and $\pi^3_{l^-,\bar{c},\bar{p}}$ will not result in as much improvement of the objective function, in addition no other π^3 should be increased without increasing a corresponding π^2 as $\mathcal{B}\bar{x}$ is always positive and would therefore not improve the objective function. Therefore $\pi^2_{l^+,\bar{c},\bar{p}}$ and $\pi^3_{l^-,\bar{c},\bar{p}}$ should be increased until (3.46) is binding, from where it is impossible to increase any $\pi^2_{l^+,\bar{c},\bar{p}}$ or $\pi^3_{l^-,\bar{c},\bar{p}}$ further, and therefore the solution is optimal. This occurs when $\epsilon = \frac{1}{(2+\omega^0)}$, and gives the following optimal solution $\pi^0 = \pi^2_{l^+,\bar{c},\bar{p}} = \pi^3_{l^-,\bar{c},\bar{p}} = \frac{1}{(2+\omega^0)}$, and corresponding cut:

$$\frac{1}{(2+\omega^0)}\left(\sum_{k\in K}(\mathcal{B}_{l^+,k,\bar{c},\bar{p}}x)\right) - \frac{1}{(2+\omega^0)}\left(\sum_{k\in K}(\mathcal{B}_{l^-,k,\bar{c},\bar{p}}x)\right) - \frac{1}{(2+\omega^0)}\eta \leq 0. \tag{3.48}$$

The cut in (3.48) is equivalent to the earlier cut in (3.41). □

We have now shown that the optimisation problem formulated in section 3.2 can be solved using Benders' decomposition. The result that the cuts can be generated by direct computation is valuable as it makes the SPs easier to solve than if an LP had to be solved. Further the equivalence of these cuts with those proposed by Fischetti, Salvagnin, and Zanette (2010) is also of note as the cuts they propose are the most-violated cuts. We also note that although proposition 3.2 relies on the assumption that $\mathcal{B}\bar{x}$ is positive, it is possible to relax this assumption. This would be achieved by first removing the non-negativity constraints on N, M^+, and M^-, and following through the consequences of this change, most notably that the constraints (3.22)–(3.24) in the dual SP become equality constraints.

Conclusion

In this chapter we presented a rostering model for physicians in a GM department. The objective of the model is to balance the workloads of the physicians, in terms of the number of patients they are caring for, by minimising the largest difference in workloads. Historical patient admission and discharge data is utilised in the model by including a model of the patient pathways through the GM department and linking this model to the rosters of the physicians.

The rostering model is extended to take into account uncertainty in the patient admissions and discharges. Scenarios are used to represent the uncertainty in these parameters, instead of parametric or empirical distributions, as they are more easily translated into the patient workloads of the physicians. A risk averse approach is also considered, where only the worst $(1 - \beta)\%$ of scenarios are included in the objective function.

Finally, a Benders' decomposition of the problem was introduced, in which it is shown that instead of solving an LP for each SP, the solutions can be computed from closed form expressions. In addition these closed form expressions for cuts are shown to be the same as the cuts produced when using another approach to find cuts for Benders' decomposition proposed by Fischetti, Salvagnin, and Zanette (2010).

In the next chapter we apply the model presented here to a case studies at Waitemata District Health Board (WDHB) and Auckland District Health Board (ADHB) to both determine the value of the approach and demonstrate how the model can be applied in different situations.

— Chapter 4 —

General Medicine Rostering Case Studies

In chapter 3 we formulated an optimisation model that creates cyclic rosters for physicians taking into account the number of patients that the physicians are caring for. In order to define an instance of this optimisation model we require knowledge of: the rules that govern the physicians' rosters, the pathways that patients follow, and historical patient data.

In this chapter we present case studies of two hospitals in Auckland, New Zealand, that have departments where the rostering model has been applied. The first involves the expansion of the general medicine (GM) department at Waitakere Hospital (WTH), a large hospital in West Auckland. Section 4.1 describes the pertinent details of this department and demonstrates how the inputs for the optimisation model are derived from this description. In addition historical patient data is used to compare a roster proposed by the hospital to ones created by the optimisation model.

The second case study is the GM department at Auckland City Hospital (ACH). This is presented in section 4.2 and serves only as an example of how the model can be applied to different hospitals, rather than investigating the value that using the model can provide.

4.1 Waitakere Hospital

The Waitemata District Health Board (WDHB) provides health services to the North Shore, Rodney and West Auckland areas of the Auckland region in New

Zealand. It serves the largest population of any District Health Board in New Zealand, with a population of more than 600,000 people, and is expected to grow by approximately 110,000 people (20%) in the next 10 years. WDHB operates two hospitals, Waitakere Hospital (WTH) and North Shore Hospital (NSH), and 14 other clinics.

In each of the hospitals there is a GM department. These departments deal with the diagnosis and treatment of patients that do not require surgery. The staff in GM are split into teams. Each team consists of a consultant (a specialist general physician), a registrar (a doctor training to be a specialist) and a house officer (a more junior doctor also in training).

4.1.1 Background

At WTH the patients assigned to GM are cared for in two areas: the inpatient wards; and the assessment, diagnostic, and cardiology unit (ADCU). The physicians who work in each area are divided into teams and the members of the teams that need to be considered for rostering purposes are called registrars. The registrars are unique because their shifts determine who cares for new admissions to GM. The other physicians in the teams do not need to be considered because their rosters can be determined from the rosters of the registrars.

In 2014 the GM department at WTH was undergoing an expansion up to a total of 12 GM teams to be split between the inpatient wards and the ADCU. This expansion was seen by the management team at WDHB as an opportunity to improve the continuity of care of the patients in GM.

Two possibilities were considered for splitting the teams between the inpatient wards and the ADCU. In the first, 10 teams would work in the inpatient wards and 2 in the ADCU, and in the second 8 teams would work in the inpatient wards and 4 in the ADCU. For the first split (10 inpatient, 2 ADCU), only one configuration of the ADCU registrars was considered, whereas for the second split (8 inpatient, 4 ADCU), two different configurations of the ADCU registrars were considered.

Patients are admitted to GM either from the emergency department (ED) or via referral from an external source, for example their general practitioner (GP). Patients who arrive at the hospital requiring immediate care are first seen by a physician in ED. The ED physician determines whether the patient has a GM issue; in other words if they are sick and do not need surgery. If the patient has a GM issue then the ED physician calls the GM registrar that is on call. The patient is then assessed by the GM registrar to determine how long they will remain in hospital. If the patient is expected to stay in hospital for less than 48 hours they are classified as short stay, otherwise they are classified as long stay.

Some patients are referred to the hospital directly from their GPs. These patients first go to the ADCU, where again it is determined whether they have a GM issue and whether they are short or long stay.

All patients that are considered to be long stay are split evenly between the inpatient ward teams of the registrars that are working the admission shift that day. Short stay patients are taken care of by the teams in the ADCU, up to a limit of 6 new patients each day per ADCU registrar working the admission shift. Any short stay patients above this limit are transferred to an inpatient ward before the post-acute round the following day, and are split evenly between the admitting inpatient ward registrars just like long stay patients. In addition if a patient that was originally diagnosed as short stay ends up staying in the ADCU for more than 24 hours they are transferred to an inpatient ward. Figure 4.1 gives a flow diagram depicting an overview of the patient flow.

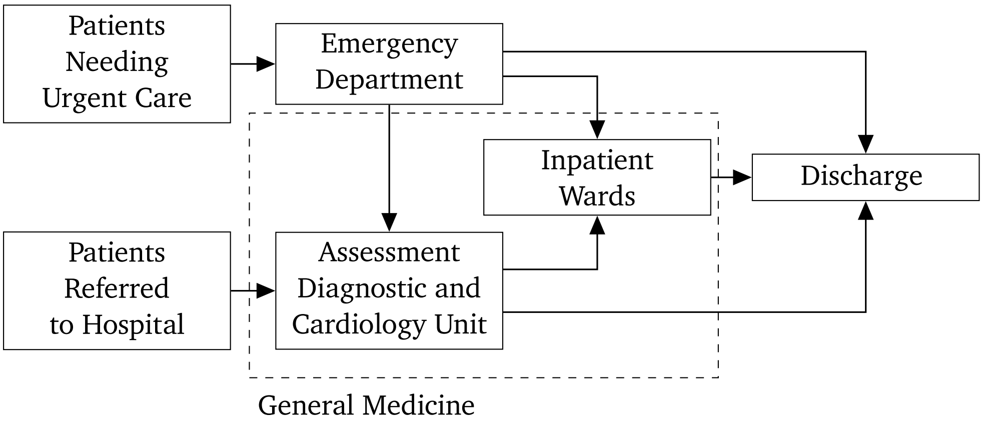

Patients Needing Urgent Care
Emergency Department
Inpatient Wards
Discharge
Patients Referred to Hospital
Assessment Diagnostic and Cardiology Unit
General Medicine

Figure 4.1: Patient Flow in General Medicine

There is also a limit of 15 on the total number of new patients that an inpatient ward team can accept in a day. However instead of the excess patients being taken care of elsewhere in Waitakere Hospital (WTH), they remain in the inpatient ward until the following morning when they are transferred to another hospital.

The registrars in GM can be assigned many different types of shifts, however only some of them are relevant to the rostering problem. The relevant shifts are:

- Admission shift (A). The registrar is on call during the day and new patients are shared between their team and any other teams with registrars working this shift;
- Post-acute shift (P). This is performed the day after the A shift, to follow up on patients;
- Night shift (N). The registrar is on call at night, however admissions are still split between the teams of the registrars who were working the A shift during the day. The admissions are given to the A shift teams because the registrar working this shift will not be there to perform a post-acute shift the following day, as they will be working the night shift again. In addition if the registrar is scheduled to perform any admission shifts during their night run then the admissions are given to their own workload group, as a relief registrar will be covering their regular shifts;
- Sleep shift (Z). After working a week of night shifts registrars are given time to catch up on sleep;
- No shift (X). Registrars are given some days off;
- Other (O). All other shifts are grouped together as they do not affect admissions or have any rules or requirements associated with them.

The rosters for the registrars who work in the inpatient wards must obey the following rules:

1. At least one registrar must perform the admission shift each day;
2. The admission shift on Saturday is followed by an admission shift on Sunday. The admission shift on all other days is followed by a post-acute shift on the next day;
3. A night 'tour' consists of 7 days of night shifts, starting on a Friday, followed by three rest days, or 'sleep' shifts;
4. Monday to Sunday of the night tour is covered by a relief registrar. During this time the registrar on night shifts can be assigned a second shift which will be performed by the relief registrar. The first three days of night shifts (Friday–Sunday) are not covered by a relief registrar;
5. Registrars must have every second weekend off (no shift).

6. If a registrar is not assigned one of the above shifts they are assigned an other shift ‘O’, which represents all of the other possible shifts they can be assigned which do not influence admissions or the rules of the described shifts;

7. Registrars cannot be assigned the night shift outside of the night tour, cannot be given an other (O) shift on weekends, and cannot be given a day off during the week.

A 10 week cyclic roster for the inpatient ward teams for the 10 inpatient, 2 ADCU split had already been proposed by the management at WDHB. This proposed roster is shown in table 4.1.

Three configurations were considered for the registrars working in the ADCU. In all configurations rule 2 for the inpatient ward registrars also applies to the ADCU registrars, however none of the other rules do. Instead they are replaced by the following: in configuration 1, two registrars alternate performing admissions from Monday to Friday; in configuration 2, four registrars share performing admissions, with two registrars admitting each day, from Monday to Friday; in configuration 3, four registrars share performing admissions (in no fixed order), with one registrar admitting each day, from Monday to Sunday and all having every second weekend off.

Configuration 1 represents the maximum possible coverage when there are only two ADCU registrars, since the post-acute shift must be worked the day after the

Table 4.1: Proposed Roster for 10 Inpatient Ward Teams

Week	Mon	Tue	Wed	Thu	Fri	Sat	Sun
1	A	P	O	O	N	N	N
2	N	N, A	N, P	N	Z	Z, A	Z, A
3	P	O	O	A	P	X	X
4	O	O	O	O	A	P	X
5	O	O	O	A	P	X	X
6	O	O	A	P	O	X	X
7	O	O	O	O	A	P	X
8	O	A	P	O	O	X	X
9	A	P	O	O	O	A	A
10	P	O	A	P	O	X	X

Table 4.2: Roster for ADCU Configuration 1

Week	Mon	Tue	Wed	Thu	Fri	Sat	Sun
1	A	P	A	P	A	P	X
2	X	A	P	A	P	X	X

Table 4.3: Roster for ADCU Configuration 2

Week	Mon	Tue	Wed	Thu	Fri	Sat	Sun
1	A	P	A	P	A	P	X
2	O	A	P	A	P	X	X
3	A	P	A	P	A	P	X
4	O	A	P	A	P	X	X

Table 4.4: Roster for ADCU Configuration 3

Week	Mon	Tue	Wed	Thu	Fri	Sat	Sun
1	A	P	O	O	O	A	A
2	P	O	O	O	O	X	X
3	O	O	A	P	A	P	X
4	O	A	P	A	P	X	X

admission shift, which prevents a registrar from admitting two days in a row. Admission shifts on the weekends cannot be covered with only two registrars. This is because it takes two separate teams to cover the weekend, one to post-acute on Saturday (after admitting on Friday) and the other to admit on Saturday and Sunday. Combined with the fact that registrars must have every second weekend off this means four registrars are needed to fully cover the weekend.

Configurations 2 and 3 represent different deployments of four registrars. Configuration two is simply doubling configuration 1 by having two registrars admit each weekday. Configuration 3 instead uses the extra registrars to perform admission shifts on the weekends, however only one registrar is admitting each day. Tables 4.2 to 4.4 show possible rosters for the three ADCU configurations.

4.1.2 Parameters

The values of the parameters that define the model for Waitakere Hospital (WTH) are given in this section, except for those that depend on the historical data and patient flow ($A_{k,c,p}$, $Q_{k,c,p}$, $U_{k,c,p,o,s}$, $T_{l,k,c,p}$), which are described separately in the following section. Only the parameters for when there are 10 teams in the inpatient wards, and the ADCU is staffed using configuration 3 are presented here, as the changes to the parameters for a different number of inpatient ward teams or a different ADCU scenario are minor. The general parameters are described first, before moving on to the parameters that depend on the rostering rules.

There are two groups of physicians that need to be rostered: the registrars in the inpatient wards, and the registrars in the ADCU. Therefore the set of roster groups is defined as $O = \{\text{Inpat}, \text{ADCU}\}$. The sets of physicians in each roster group are simply defined as the set of integers from 1 to the number of physicians in that roster group, so $R_{\text{Inpat}} = \{1, 2, 3, \ldots, 10\}$ and $R_{\text{ADCU}} = \{1, 2, 3, 4\}$. The sets of weeks in each roster groups' cyclic roster are equal to the sets of physicians so $W_{\text{Inpat}} = R_{\text{Inpat}}$ and $W_{\text{ADCU}} = R_{\text{ADCU}}$.

The shifts considered are: acute, post-acute, night, sleep, day-off, and other. Therefore the set of shifts for the inpatient teams is defined as $S_{\text{Inpat}} = \{$A, P, N, Z, X, O$\}$. The registrars in the ADCU do not have to work any night shifts (and therefore no sleep shifts), therefore their set of shifts is $S_{\text{ADCU}} = \{$A, P, X, O$\}$.

Because, in both the inpatient wards and the ADCU, each team only takes care of their own patients, and there is one registrar per team, one workload group is defined for each registrar. The set of workload groups is then defined as $L = \{\text{Inpat1}, \ldots, \text{Inpat10}, \text{ADCU1}, \ldots, \text{ADCU4}\}$. The indicator parameter that maps physicians to workload groups is $V_{o,r,l} = 1$ if l is the concatenation of o and r, 0 otherwise.

Defining one workload group for each physician being rostered makes the distinction of workload groups redundant for this problem, as tracking workload by group is equivalent to tracking workload by physician. In general, however, there is a need to distinguish between physicians and workload groups, as is demonstrated by the second case study in section 4.2.

One year, or 52 weeks, is used for the planning horizon. Therefore the set that counts all of the days in the planning horizon is $C = \{1, 2, 3, \ldots, 364\}$. The set of days of the week is $D = \{$Mon, Tue, Wed, Thu, Fri, Sat, Sun$\}$. Each day is divided into three parts: day (8 a.m.–4 p.m.), evening (4 p.m.–10 p.m.), and night (10 p.m.–8 a.m.). These periods are chosen because they correspond to the times that the shifts start (admission starts at 8 a.m., night at 4 p.m.). The set of the parts of the day is then $P = \{$D(ay), E(vening), N(ight)$\}$.

The final general parameter is the set of patient classes. The general medicine department distinguishes between patients that are expected to stay for less than

Table 4.5: Waitakere Hospital General Sets and Parameters

Symbol	Example
O	{Inpat, ADCU}
R_{Inpat}	$\{1, 2, 3, \ldots, 10\}$
R_{ADCU}	$\{1, 2, 3, 4\}$
W_{Inpat}	$\{1, 2, 3, \ldots, 10\}$
W_{ADCU}	$\{1, 2, 3, 4\}$
S_{Inpat}	{A, P, O, X, N, Z}
S_{ADCU}	{A, P, O, X, Z}
L	{Inpat1, …, Inpat10, ADCU1, …, ADCU4}
$V_{o,r,l}$	1 if l is the concatenation of o and r, 0 otherwise. For example $V_{\text{Inpat,1,Inpat1}} = 1$
C	$\{1, 2, 3, \ldots, n_C\}$
D	{Mon, Tue, Wed, Thu, Fri, Sat, Sun}
P	{D, E, N}
K	{S/S New, S/L New, L/S New, L/L New, S/S Old, S/L Old, L/S Old, L/L Old}, where 'X/X' refers first to the patient's actual duration (either short stay (S) or long stay (L)) and second to the presented stay. In addition 'New' refers to patients that were admitted that day, and 'Old' to patients that were admitted on a previous day.

48 hours, short stay, or more than 48 hours, long stay. In addition, the duration that the patient actually stays for influences their path, as patients originally thought to be short stay are moved to the inpatient wards if they actually stay for more than 24 hours. Further, whether a patient was admitted on that day, or on an earlier day is important, as there are limits on the number of new patients teams can admit in a day. The set of patients classes can therefore be written as K = {S/S New, S/L New, L/S New, L/L New, S/S Old, S/L Old, L/S Old, L/L Old}, where 'X/X' refers first to the patient's actual length of stay (either short stay (S) or long stay (L)) and second to the expected stay. In addition 'New' refers to patients that were admitted that day, and 'Old' to patients that were admitted on a previous day. The general parameters for the WTH problem are summarised in table 4.5.

The rules described in the previous section define the parameters $F_o, B_o, \mathcal{L}_o$, $G_o, \mathcal{R}_o$, and i_o, which represent the fixed, banned, linked, following, and required shifts.

Note that here we begin to use 'set-builder notation', which is used to describe sets and their elements. In particular we make use of the logical 'and' operator $\wedge$.

For the inpatient roster group the night 'tour' is fixed to start on the Friday night in the first week, and finish on Thursday night in the second week. The night tour can be fixed here without eliminating any possible solutions because any solution with the night tour in a different place can be 'rotated' around the cyclic roster so that the night tour once again starts in the first week.

Therefore F_{Inpat} must contain all of the shift, week, day tuples that correspond to that night tour, so $F_{\text{Inpat}} = \{$(N, 1, Fri), (N, 1, Sat), (N, 1, Sun), (N, 2, Mon), (N, 2, Tue), (N, 2, Wed), (N, 2, Thu), (Z, 2, Fri), (Z, 2, Sat), (Z, 2, Sun)$\}$. There are no fixed shifts for the registrars in the ADCU so $F_{\text{ADCU}} = \emptyset$.

Since the night tour starts on the Friday of the first week the days preceding this must not be assigned a night shift, a sleep shift or a day off. Therefore the set $\{(s, 1, d) : s \in \{\text{X, N, Z}\} \wedge d \in \{\text{Mon, Tue, Wed, Thu}\}\}$ must be a subset of the set of banned shifts. As there is no relief registrar for the first three days of the night tour, or the final three days of sleep shifts, no other shifts can be assigned during this time and both $\{(s, 1, d) : s \in \{\text{A, P, Z, X, O}\} \wedge d \in \{\text{Fri, Sat, Sun}\}\}$ and $\{(s, 2, d) : s \in \{\text{A, P, N, X, O}\} \wedge d \in \{\text{Fri, Sat, Sun}\}\}$ should be added to the set of banned shifts. In all of the other weeks, days off, night shifts, and sleep shifts should not be assigned during the week so $\{(s, w, d) : s \in \{\text{O, N, Z}\} \wedge w \in W_{\text{Inpat}} \setminus \{1, 2\} \wedge d \in \{\text{Sat, Sun}\}\}$ should be added to the set of banned shifts. Also, on all of the other weekends, other shifts, night shifts, and sleep shifts should not be assigned so $\{(s, w, d) : s \in \{\text{O, N, Z}\} \wedge w \in W_{\text{Inpat}} \setminus \{1, 2\} \wedge d \in \{\text{Sat, Sun}\}\}$ should be added to the set of banned shifts. There are no banned shifts for the registrars in the ADCU so $B_{\text{ADCU}} = \emptyset$.

Linked shifts are used to make sure that when a registrar works the admission shift on one day, they work the post-acute shift the following day, except for on Saturdays where if they work the admission shift they must also work the admission shift on Sunday and then the post-acute shift on Monday. Further these post-acute and admission shifts can only ever be worked if the prior shifts are also worked. This applies to both the registrars in the inpatient wards and those in the ADCU. Therefore for each roster group o, for each week w in that roster groups' set of weeks W_o, for each day of the week d other than Saturday and Sunday, the value of $\mathcal{L}_{o,\text{A},w,d}$ is $\{(\text{P}, w, \hat{d})\}$, where $\hat{d}$ is the next day of the week after d. In addition the value of $\mathcal{L}_{o,\text{A},w,\text{Sat}}$ is $\{(\text{A}, w, \text{Sun})\}$, and $\mathcal{L}_{o,\text{A},w,\text{Sun}}$ is $\{(\text{P}, f_3(w + 1, o), \text{Mon})\}$, i.e., the post-acute shift on Monday of the following week in the cyclic roster.

Following shifts are defined so that if a registrar works in a weekend, then they must have the following weekend off. They can have the following weekend off even if they did not work the previous weekend, however. Note that for the inpatient ward teams the night shift on a Friday also counts as working in a weekend. For both roster groups the following shifts for the admission and post-acute shifts, on Saturdays and Sundays in each week are given as $G_{o,s,w,d} = \{(\mathrm{X}, f_3(w+1,o), \mathrm{Sat}), (\mathrm{X}, f_3(w+1,o), \mathrm{Sun})\}, \forall o \in O, w \in W_o, s \in \{\mathrm{A}, \mathrm{P}\}, d \in \{\mathrm{Fri, Sat, Sun}\}$. For just the inpatient wards roster group the following shifts for the night shift on Fridays, Saturdays, and Sundays are given as $G_{\mathrm{Inpat},\mathrm{N},w,d} = \{(\mathrm{X}, f_3(w+1,\mathrm{Inpat}), \mathrm{Sat}), (\mathrm{X}, f_3(w+1,\mathrm{Inpat}), \mathrm{Sun})\}, \forall w \in W_{\mathrm{Inpat}}, d \in \{\mathrm{Fri, Sat, Sun}\}$.

The required shifts for the inpatient ward teams are the admission shift and the night shift. For the ADCU teams the only required shift is the admission shift. For the inpatient teams two admission shifts and one night shift must be performed each day, and for the ADCU teams one admission shift must be performed each day. Therefore the set $\mathcal{R}_O$, which defines the shifts and days of the week on which have to be assigned to a required number of physicians, for the inpatient ward teams is $\mathcal{R}_{\mathrm{Inpat}} = \{(s,d) : s \in \{\mathrm{A}, \mathrm{N}\} \wedge d \in D\}$, and for the ADCU teams is $\mathcal{R}_{\mathrm{ADCU}} = \{(\mathrm{A}, d) : d \in D\}$. For each of the shift, day tuples in $\mathcal{R}_o$, the number of shifts of that type that must be performed on that day is given by $i_{o,s,d}$. For the inpatient ward teams $i_{o,\mathrm{A},d} = 2 \ \forall d \in D$ and $i_{o,\mathrm{N},d} = 1 \ \forall d \in D$. For the ADCU teams $i_{o,\mathrm{A},d} = 1 \ \forall d \in D$.

Finally the number of shifts that can be assigned to physicians in each roster group on each day of each week in that roster group's cyclic roster is given by $y_{o,w,d}$. For the ADCU teams this is always 1, as they can only ever be assigned one shift each day. The inpatient ward teams can be assigned two shifts on the days of the night tour that are covered by another registrar, on all other days they can only be assigned one shift. Therefore $y_{\mathrm{ADCU},w,d} = 1$ for all days in all weeks, and $y_{\mathrm{Inpat},w,d} = 2$ for week 2, and is equal to 1 otherwise. No exclusive shifts or required shifts for the workload groups are needed to model the rules for either the inpatient ward or ADCU teams. These parameters are summarised in table 4.6.

The final parameter for the optimisation model is the function, $j : K \times C \times P \longrightarrow \mathcal{P}(K \times C \times P)$, which describes how the patients' classes change throughout their stay in the GM department. Recall that a patient's class is made up of two parts: the first part is a combination of their actual length of stay (either S or L) and their expected length of stay (S or L), for example 'L/S' if a patient is actually

Table 4.6: Waitakere Hospital Sets and Parameters Dependent on Hospital Rules

Parameter	Value
F_{Inpat}	{(N, 1, Fri), (N, 1, Sat), (N, 1, Sun), (N, 2, Mon), (N, 2, Tue), (N, 2, Wed), (N, 2, Thu), (Z, 2, Fri), (Z, 2, Sat), (Z, 2, Sun)}
F_{ADCU}	$\emptyset$
B_{Inpat}	$\{(s,1,d) : s \in \{\text{X, N, Z}\} \wedge d \in \{\text{Mon, Tue, Wed, Thu}\}\cup$ $\{(s,1,\text{Fri}) : s \in \{\text{X, Z}\}\}\cup$ $\{(s,1,d) : s \in \{\text{O, Z}\} \wedge d \in \{\text{Sat, Sun}\}\}\cup$ $\{(s,2,d) : s \in \{\text{X, Z}\} \wedge d \in \{\text{Mon, Tue, Wed, Thu}\}\}\cup$ $\{(s,2,\text{Fri}) : s \in \{\text{X, N}\}\}\cup$ $\{(s,2,d) : s \in \{\text{O, N}\} \wedge d \in \{\text{Sat, Sun}\}\}\cup$ $\{(s,w,d) : s \in \{\text{X, N, Z}\} \wedge w \in W_{\text{Inpat}} \setminus \{1,2\} \wedge d \in \{\text{Mon, Tue, Wed, Thu, Fri}\}\}\cup$ $\{(s,w,d) : s \in \{\text{O, N, Z}\} \wedge w \in W_{\text{Inpat}} \setminus \{1,2\} \wedge d \in \{\text{Sat, Sun}\}\}$
B_{ADCU}	$\emptyset$
$\mathcal{L}_{o,\text{A},w,\text{Mon}}$	$\{(\text{P}, w, \text{Tue})\}, \forall o \in O, w \in W_o$
$\mathcal{L}_{o,\text{A},w,\text{Tue}}$	$\{(\text{P}, w, \text{Wed})\}, \forall o \in O, w \in W_o$
$\mathcal{L}_{o,\text{A},w,\text{Wed}}$	$\{(\text{P}, w, \text{Thu})\}, \forall o \in O, w \in W_o$
$\mathcal{L}_{o,\text{A},w,\text{Thu}}$	$\{(\text{P}, w, \text{Fri})\}, \forall o \in O, w \in W_o$
$\mathcal{L}_{o,\text{A},w,\text{Fri}}$	$\{(\text{P}, w, \text{Sat})\}, \forall o \in O, w \in W_o$
$\mathcal{L}_{o,\text{A},w,\text{Sat}}$	$\{(\text{A}, w, \text{Sun})\}, \forall o \in O, w \in W_o$
$\mathcal{L}_{o,\text{A},w,\text{Sun}}$	$\{(\text{P}, f_3(w+1, o), \text{Mon})\}, \forall o \in O, w \in W_o$
$G_{\text{Inpat},\text{N},w,d}$	$\{(\text{X}, f_3(w+1, \text{Inpat}), \text{Sat}), (\text{X}, f_3(w+1, \text{Inpat}), \text{Sun})\}, \forall w \in W_{\text{Inpat}}, d \in \{\text{Fri, Sat, Sun}\}$
$G_{o,s,w,d}$	$\{(\text{X}, f_3(w+1, o), \text{Sat}), (\text{X}, f_3(w+1, o), \text{Sun})\}, \forall o \in O, w \in W_o, s \in \{\text{A, P}\}, d \in \{\text{Fri, Sat, Sun}\}$
$\mathcal{R}_{\text{Inpat}}$	$\{(s,d) : s \in \{\text{A}, \text{N}\} \wedge d \in D\}$
$i_{\text{Inpat},\text{A},d}$	$2\ \forall d \in D$
$i_{\text{Inpat},\text{N},d}$	$1\ \forall d \in D$
$\mathcal{R}_{\text{ADCU}}$	$\{(\text{A}, d) : d \in D\}$
$i_{\text{ADCU},\text{A},d}$	$1\ \forall d \in D$
$y_{\text{Inpat},w,d}$	$2\ \forall d \in D, w \in \{2\}$ 1 otherwise
$y_{\text{ADCU},w,d}$	$1\ \forall d \in D, w \in W_{\text{ADCU}}$

Table 4.7: WTH Functions Dependent on Hospital Rules

Parameter	Value
$j(\hat{k}+\text{Old},c,\text{D})$	$\{(\hat{k}+\text{Old},c-1,\text{N}),(\hat{k}+\text{New},c-1,\text{N})\}$ $\forall\hat{k}\in\hat{K},c\in C$
$j(\hat{k}+\text{Old},c,\text{E})$	$\{(\hat{k}+\text{Old},c,\text{D})\}\ \forall\hat{k}\in\hat{K},c\in C$
$j(\hat{k}+\text{Old},c,\text{N})$	$\{(\hat{k}+\text{Old},c,\text{E})\}\ \forall\hat{k}\in\hat{K},c\in C$
$j(\hat{k}+\text{New},c,\text{E})$	$\{(\hat{k}+\text{New},c,\text{D})\}\ \forall\hat{k}\in\hat{K},c\in C$
$j(\hat{k}+\text{New},c,\text{N})$	$\{(\hat{k}+\text{New},c,\text{E})\}\ \forall\hat{k}\in\hat{K},c\in C$

a long stay, but was expected to be a short stay; the second part depends on whether they were admitted that day ('New') or earlier ('Old'). Therefore the only time that a patient's class changes is when they move from the 'Night' part on the day they are admitted to the 'Day' part on the following day. In all other transitions the patient's class remains the same. The predecessor class, count day, part tuples of a given class, count day, part tuple is therefore simply the same class on the same day in the preceding part except for the 'Old' classes in the 'Day' part. The predecessors of these classes are both the 'New' and ''Old' classes on the preceding count day in the 'Night' part. If we first define the set $\hat{K}$ = {S/S, S/L, L/S, L/L} as the possible combinations of the first part of the patient classes, and abuse the notation of + slightly to include the concatenation of strings, then the function j is defined as in table 4.7.

4.1.3 Data and Processing

We were provided with data from the GM department at Waitakere Hospital (WTH), although significant processing was required to transform it from the state in which it was provided to the inputs required for the model. This processing includes: aggregating the arrivals (and discharges) over the parts; calculating the parameters $A_{k,c,p}$, $Q_{k,c,p}$, $U_{k,c,p,o,s}$, and $T_{l,k,c,p}$; and calculating the $\mathcal{B}$ matrix.

WTH provided data on the patients that were taken care of in the GM department in 2014. During that year 9000 patients were admitted to and discharged from GM. For each one of those patients the following data was available: the date and time that they arrived at and were discharged from the hospital; the date and time that they arrived at and were discharged from GM; the nature of their discharge; and the location to which they were discharged. The type of discharge is used to distinguish patients that went home from those that are sent to another health care facility for clinical purposes. These clinical transfers

Table 4.8: Example Waitakere Hospital Patient Data

Hospital Arrival	Hospital Departure	GM Arrival	GM Departure	Discharge Type	Discharge Location
01/01/14 9:10	01/01/14 13:49	01/01/14 9:39	01/01/14 13:49	Routine	Home
05/01/14 22:13	06/01/14 2:31	05/01/14 23:25	06/01/14 2:31	Other HC	NSH
07/01/14 12:43	07/01/14 17:21	07/01/14 12:43	07/01/14 17:21	Routine	Home
10/01/14 16:27	11/01/14 11:15	10/01/14 16:27	11/01/14 11:15	Routine	Home

are different than patients transferred because a team has admitted too many patients. The patients transferred for clinical purposes are due to circumstances such as improved medical care for their condition being available at another facility. Patients that are transferred to for clinical reasons are not included in the model since they make up a very small fraction of the total number of patients, and when they do occur they only stay for a short amount of time before being transferred.

An example of the type of data provided is given in table 4.8. Note that the actual data cannot be replicated here due to its sensitive nature, and instead representative fictional data is presented.

Unfortunately there is no record of whether each patient was believed to be short or long stay, only the length of time the patient actually stayed. Instead we used estimates for the probability that a patient would be accurately classified as either short or long stay depending on their actual length of stay. Patients that actually stayed less than 24 hours are accurately classified as short stay 90% of the time. Patients that stayed more than 24 hours but less than 48 hours are accurately classified as short stay 75% of the time. Finally patients that stayed more than 48 hours are accurately classified as long stay 66% of the time.

These percentages are used alongside the data provided to determine $Q_{k,c,p}$, and the value of $A_{k,c,p}$ for all 'New' patient classes. The 'New' patient classes represent patients new to GM that day, which should be the case for all admissions. The exception is when patients are transferred from ADCU teams to inpatient teams at the beginning of the day; either because the patient has been in the ADCU for more than 24 hours, or the number of short stay patients admitted to the ADCU that day exceeded the limit of 6 new patients per ADCU team. These patients were in GM on the previous day and are therefore not 'New'. Instead these transfers from the ADCU to the inpatient teams are counted as the admission of 'Old' patients, because once they are transferred to an inpatient team they have the same pathway as conventional 'Old' patients. If the behaviour of the transferred patients was different from that of the other ‘Old’ patients then new patient classes would have been created for the transferred patients.

These transfers, and therefore the values of both $T_{l,k,c,p}$, and $A_{k,c,p}$ for ‘Old’ patient classes, depend on the number of teams working in the ADCU each day. Likewise the proportion of patients that each shift admits, $U_{k,c,p,o,s}$, also depends on the number of both inpatient teams, and ADCU teams admitting. These in turn depend on the total number of inpatient teams and the ADCU configuration. Therefore these parameters cannot be determined directly from the data; knowledge of the structure of the teams is also needed.

The value of $A_{k,c,p}$ for ‘New’ patient classes is calculated by taking the date and time each patient was admitted and converting it into the corresponding count day, part tuple, (c, p). The patients’ classes are one of ‘X/Y New’ where X and Y are either S for short stay or L for long stay. Each patient’s class is determined by examining their actual length of stay in GM, which determines X. X is S if their actual length of stay is less than 48 hours, and L if it is more. Y is then determined by generating a random number between 0 and 1; if this number is less than the corresponding probability that the patient’s length of stay is accurately classified, given their actual length of stay, then Y is equal to X, otherwise Y is the option (of S or L) different to X. This process is summarised in algorithm 4.1.

Algorithm 4.1: Admission Numbers Calculation Algorithm

```
Step 1. Set $A_{k,c,p} = 0 \ \forall k \in K, c \in C, p \in P$.
Step 2. For each patient for which there is data, as in table 4.8, perform:
  Step 2.1. Convert the date and time that the patient arrived into the
    corresponding count day and part tuple, $(c, p)$.
  Step 2.2. Calculate the patient’s actual length of stay. If it less than 48
    hours then the first part of their patient class is ‘S’, otherwise it is ‘L’.
  Step 2.3. Generate a random number between 0 and 1; if this is less
    than the probability that the patient’s length of stay is correctly classi-
    fied, then the second part of their patient class is the same as the first,
    otherwise it is the other option.
  Step 2.4. Combine these two parts with ‘New’ to get the full patient
    class $k$.
  Step 2.5. Add 1 to the corresponding value of $A_{k,c,p}$; $A_{k,c,p} = A_{k,c,p} + 1$.
```

Algorithm 4.1 is also used to calculate the number of patients discharged, $\mathcal{D}_{k,c,p}$, in each part, in each day. Although the number of patients discharged is not required in the model, only the proportion $Q_{k,c,p}$, both the number of patients discharged and the total number of patients of each class, $\mathcal{N}_{k,c,p}$, in each part, in each day is required to calculate $Q_{k,c,p}$. The number of patients discharged is calculated from the data by taking the date and time each patient was discharged and converting it into the corresponding day, part tuple, in the same way as for the number of admissions. The patients' classes are determined by their length of stay and their original class. If a patient is discharged on the same day that they are admitted then they will still be a 'New' patient, otherwise they will have become an 'Old' patient. The remaining part of the class, 'S/L' for example, is always the same as when they were admitted.

To calculate the total number of patients of each class a running total of the patients in GM is maintained from the beginning of the planning horizon. Starting from the first part of the first day of the planning horizon the number of admissions for each class is counted. The discharge proportions for that part are calculated by dividing the number of discharges in that part by the counts for each patient class. Then for every successive part the discharges from the previous part are subtracted from the patient counts, then the new admissions for that part are added to the patients that remain from the previous part, taking into account that patients transition between classes, for example from a 'New' to an 'Old' class at the end of the day. The discharge proportions for each patient class are once again calculated by dividing the number of discharges in the part by the total number of patients currently in GM. The calculation of $Q_{k,c,p}$ is summarised in algorithm 4.2, where the number of patients discharged and the running total of patients in each part of each day are represented respectively by $\mathcal{D}_{k,c,p}$, and $\mathcal{N}_{k,c,p}$.

The next parameter that is calculated is $U_{k,c,p,o,s}$, which is the proportion of admissions of class k, in part p, on day c that roster group o adds to their workload when they work shift s. For this case study these proportions depend on the number of inpatient and ADCU teams.

The ADCU admits all new patients that are initially classified as short stay, if there are ADCU teams working the 'A' shift, and these admissions are split equally between all the ADCU teams that are working the 'A' shift. The number of ADCU teams that work the 'A' shift each day depends on the ADCU configuration. In configurations 1 and 3 there is one ADCU team working the 'A' shift each day. In configuration 1 the ADCU teams can only cover Monday to Thursday, whereas in configuration 3 the ADCU teams cover every day of the week. Therefore for configuration 1, $U_{k,c,p,\text{ADCU},\text{A}} = 1 \ \forall k \in \{\text{S/S New, L/S New}\}, c \in \{\delta \in$

Algorithm 4.2: Discharge Proportion Calculation Algorithm

```
Step 1. Set $\mathcal{N}_{k,c,p} = Q_{k,c,p} = 0 \; \forall k \in K, c \in C, p \in P$.
Step 2. For each count day and part, in chronological order, and patient class perform:
    Step 2.1. Add the number of admissions in the class to the total number of patients at the end of the previous part in the predecessor classes,
        $$\mathcal{N}_{k,c,p} = A_{k,c,p} + \sum_{(\hat{k},\hat{c},\hat{p}) \in j(k,c,p)} \left( \mathcal{N}_{\hat{k},\hat{c},\hat{p}} \right).$$
    Step 2.2. Calculate the proportion of patients discharged as the number discharged divided by the total number
        $$Q_{k,c,p} = \frac{\mathcal{D}_{k,c,p}}{\mathcal{N}_{k,c,p}}.$$
    Step 2.3. Subtract the number of patients discharged from the total to get the number of patients left at the end of the part
        $$\mathcal{N}_{k,c,p} = \mathcal{N}_{k,c,p} - \mathcal{D}_{k,c,p}.$$
```

$C : f_2(\delta) \in \{\text{Mon}, \text{Tue}, \text{Wed}, \text{Thu}\}\}, p \in P$. For configuration 3, $U_{k,c,p,\text{ADCU},\text{A}} = 1 \; \forall k \in \{\text{S/S New}, \text{L/S New}\}, c \in C, p \in P$.

In configuration 2 there are two ADCU teams admitting from Monday to Thursday so $U_{k,c,p,\text{ADCU},\text{A}} = 0.5 \; \forall k \in \{\text{S/S New}, \text{L/S New}\}, c \in \{\delta \in C | f_2(\delta) \in \{\text{Mon}, \text{Tue}, \text{Wed}, \text{Thu}\}\}, p \in P$. For all other shifts and on Friday to Sunday for configurations 1 and 2, $U_{k,c,p,\text{ADCU},s} = 0$.

The inpatient teams admit all patients that are initially classified as long stay, and also admit short stay patients if there are no ADCU teams working the 'A' shift. Once again the admissions are split evenly between the teams that are working the 'A' shift.

Finally the proportion of patients transferred, $T_{l,k,c,p}$ is calculated. In this case study transfers only occur at the end of the 'Night' part for one of two reasons. Either the patient has been in the ADCU for at least one whole day, and is transferred to the inpatient wards, or the limit on the number of new patients that a team can receive in a day is reached. If the limit of new patients is reached for an ADCU team the additional patients are transferred to the inpatient ward teams working the admission shift the next day, if the limit of new patients is reached for an inpatient ward team then the patients are transferred to another healthcare facility. The transfers from the ADCU to the inpatient teams are included in the model in the $A_{k,c,p}$ parameter for 'Old' patient classes.

For a patient to have been in the ADCU for at least 24 hours, they must be an 'Old' patient class, as they will always transition to this at the end of their first day. Therefore at the end of each day any patients that are of an 'Old' patient class that are still in the ADCU should be transferred to the inpatient wards the following morning. This means that $T_{l,k,c,p} = 1$ for all ADCU workload groups and 'Old' patient classes, in the 'N' parts on all days.

To calculate the proportion of patients transferred because the number of new patients that a team can receive has been reached, a similar procedure is used as that for calculating the proportion of patients discharged. On each day the total number of new patients over the three parts of the classes that each roster group is responsible for is calculated. If this is more than the number of new patients that can be admitted by that roster groups' admitting teams the number of patients above the limit must be transferred. A strict ordering of the patient classes is used to determine which patients are transferred first. If there are not enough patients of the highest priority class then the next class down is used, until enough patients have been selected. The proportion of patients transferred is then calculated by dividing the number of patients in each class that are transferred by the total number of patients that are in that class. The total number of patients is calculated by keeping a running total of admissions minus discharges, as in the calculation of the discharge proportions.

Algorithm 4.3 describes the calculation of the transfer proportions in detail. To make the description of the algorithm easier we first define some new notation. Since the workload groups contain either only inpatient ward registrars or only ADCU registrars we define two subsets of the workload groups, the inpatient groups, $L_{\text{Inpat}} = \{\text{Inpat1}, \ldots, \text{Inpat10}\}$, and the ADCU groups, $L_{\text{ADCU}} = \{\text{ADCU1}, \ldots, \text{ADCU4}\}$. We call these two subsets 'admission groups' and they are indexed by the roster groups, as they correspond to the two roster groups, Inpat and ADCU. They are used in the algorithm because we need to calculate the total number of patients admitted by these two types of workload groups to determine whether any patients need to be transferred.

We also split the patient classes into two subsets, the 'Old' classes, $K^{(o)} = \{\text{S/S Old, S/L Old, L/S Old, L/L Old}\}$, and the 'New' classes, $K^{(n)} =$

{L/S New, L/L New, S/L New, S/S New}. This is done because only 'New' patients count towards the workload groups limits before patients need to be transferred. The ordering in $K^{(n)}$ is also the order in which the patient classes are chosen to be transferred from, when transfers are required.

Finally, within the algorithm $\mathcal{T}_{o,k,c,p}$ is used to represent the number of patients transferred (not the proportion) by an admission group, $\mathcal{N}_{o,k,c,p}$ represents the total number of patients an admission group is caring for, $\mathcal{N}_{o,c}^{(n)}$ is the number of 'New' patients an admission group is caring for at the end of a day.

Once the number of admissions, and the proportion of discharges, admissions, and transfers, $A_{k,c,p}$, $Q_{k,c,p}$, $U_{k,c,p,o,s}$, $T_{l,k,c,p}$, have been calculated for a particular configuration, they can be used to determine $\mathcal{B}$. The matrix $\mathcal{B}$ maps the roster for a configuration to the workloads of each workload group in each part of each day, given the admissions data and patient flow structure. It is used to replace the patient balance equations (3.10) in the sub-problems (SPs) of Benders' decomposition. The patient balance equations are:

$$\begin{aligned} N_{l,k,c,p} = \left(1 - Q_{k,c,p}\right) & \left(\sum_{(\hat{k},\hat{c},\hat{p}) \in j(k,c,p)} \left(\left(1 - T_{l,\hat{k},\hat{c},\hat{p}}\right) N_{l,\hat{k},\hat{c},\hat{p}} \right) \right. \\ & + \left. \sum_{o \in O, s \in S_o, r \in R_o} \left(A_{k,c,p} U_{k,c,p,o,s} x_{o,s,f_1(o,r,c),f_2(c)} V_{o,r,l} \right) \right) \\ & \forall l \in L, k \in K, c \in C, p \in P \end{aligned} \tag{3.10}$$

We now have values for $Q_{k,c,p}$, $T_{l,k,c,p}$, $A_{k,c,p}$, $U_{k,c,p,o,s}$, and $V_{o,r,l}$. In addition we know the set $\{(\hat{k},\hat{c},\hat{p}) \in j(k,c,p)\}$, and that every $(\hat{c},\hat{p})$ in the set is the direct predecessor of (c,p). Therefore we can rewrite these equations as $N = \mathcal{B}x$, where the values of $\mathcal{B}$ are calculated starting in the first part of the first day and substituting the values of the parameters in the right hand side. Then for each subsequent part the values of $\mathcal{B}$ already determined are used for $N_{l,\hat{k},\hat{c},\hat{p}}$ on the right hand side.

4.1.4 Scenario generation

The steps to calculate $A_{k,c,p}$, $Q_{k,c,p}$, $U_{k,c,p,o,s}$, and $T_{l,k,c,p}$ described in the previous section depended on the historical data provided. We now wish to substitute the historical data with data that we have generated ourselves in order to represent other plausible admission and discharge scenarios.

Algorithm 4.3: Transfer Proportion Calculation Algorithm

```
Step 1. Set $T_{l,k,c,p} = \mathcal{T}_{l,k,c,p} = 0 \; \forall l \in L, k \in K, c \in C, p \in P$.
Step 2. For each count day in chronological order perform:
  Step 2.1. For each part, in chronological order, admission group, and patient class perform:
    Step 2.1.1. Calculate the number of patients that are being cared for by the admission group as a whole by adding the new admissions the admission group is responsible for to the number carried over from the previous parts, multiplied by $1 - Q_{k,c,p}$:
      $$\mathcal{N}_{o,k,c,p} = \left(1 - Q_{k,c,p}\right)\left(U_{k,c,p,o,\mathrm{A}} A_{k,c,p} + \sum_{(\hat{k},\hat{c},\hat{p}) \in j(k,c,p)} \left(\mathcal{N}_{\hat{k},\hat{c},\hat{p}}\right)\right).$$
  Step 2.2. For each admission group perform:
    Step 2.2.1. Calculate the number of 'New' patients, $\mathcal{N}^{(n)}_{o,c}$, that each admission group has at the end of the day (in the 'Night' part):
      $$\mathcal{N}^{(n)}_{o,c} = \sum_{k \in K^{(n)}} \left(\mathcal{N}_{o,k,c,\mathrm{N}}\right).$$
    Step 2.2.2. If $\mathcal{N}^{(n)}_{o,c}$ is greater than the limit of 'New' patients for that admission group (15 for inpatient wards, 6 for ADCU) multiplied by the number of teams admitting on the count day, then set the number of patients that need to be removed, $\mathcal{R}$ as the difference between $\mathcal{N}^{(n)}_{o,c}$ and the limit and go to Step 2.2.3, otherwise go to Step 2.2.4.
    Step 2.2.3. For each 'New' patient class in the order given by $K^{(n)}$ perform:
      Step 2.2.3.1. If $\mathcal{R} > 0$ go to Step 2.2.3.2, otherwise go to Step 2.2.4.
      Step 2.2.3.2. If $\mathcal{R} \geq \mathcal{N}_{o,k,c,\mathrm{N}}$ go to Step 2.2.3.3, otherwise go to Step 2.2.3.4.
      Step 2.2.3.3. Set $\mathcal{T}_{o,k,c,\mathrm{N}} = \mathcal{N}_{o,k,c,\mathrm{N}}$, and set $\mathcal{R} = \mathcal{R} - \mathcal{N}_{o,k,c,\mathrm{N}}$, skip Step 2.2.3.4.
      Step 2.2.3.4. Set $\mathcal{T}_{o,k,c,\mathrm{N}} = \mathcal{R}$, and set $\mathcal{R} = 0$.
```

Algorithm 4.3: Continued

Step 2.2.4. If the admission group is ADCU then all ‘Old’ patients get transferred at the end of the day so set $\mathcal{T}_{o,k,c,\mathrm{N}} = \mathcal{N}_{o,k,c,\mathrm{N}}\ \forall k \in K^{(o)}$. and add these transfers from the ADCU to the admissions to the in-patient wards

$$A_{k,c+1,\mathrm{D}} = \sum_{(\hat{k},\hat{c},\hat{p})\in j(k,c,p)} \left(\mathcal{T}_{o,\hat{k},\hat{c},\hat{p}}\right)\ \forall k \in K^{(o)}.$$

Step 2.2.5. Calculate the proportion of patients transferred for the workload groups that correspond to the admission group using

$$T_{l,k,c,\mathrm{N}} = \frac{\mathcal{T}_{o,k,c,\mathrm{N}}}{\mathcal{N}_{o,k,c,\mathrm{N}}}\ \forall l \in L_o.$$

Each scenario is determined by: 1) the number of patients that are admitted in each part (of a day); 2) the length of stay of each patient; and 3) the initial classification of each patient by the GM physicians. From this information the discharges for parts (of a day) and patient classes can be determined, and therefore all of the remaining input parameters.

In order to generate scenarios we wish to use the historical data to inform the random generation of: 1) numbers of admissions; 2) lengths of stays; and 3) initial classifications for each part in the planning horizon. We decided to use random sampling from the historical data with replacement to determine both the number of admissions in a part, and the lengths of stay of these admissions. Once the lengths of stay are known the patient class can be determined by randomly allocating an initial classification based on the accuracy probabilities provided by the hospital. Hence, all three components required for the scenario data can be determined.

To perform the resampling procedure a pool of potential samples is needed for each part of each day. To determine the appropriate pool the number of admissions in each part of each day in the historical data was visualised in figure 4.2.

Figure 4.2 shows that there is a difference between the number of patients admitted in the day, evening, and night parts. In addition there is a clear seasonal fluctuation in the number of admissions in the day parts, and small seasonal fluctuation for the evening and night parts. This suggests that when sampling the number of admissions for a part, both the type of part (day, evening, or night)

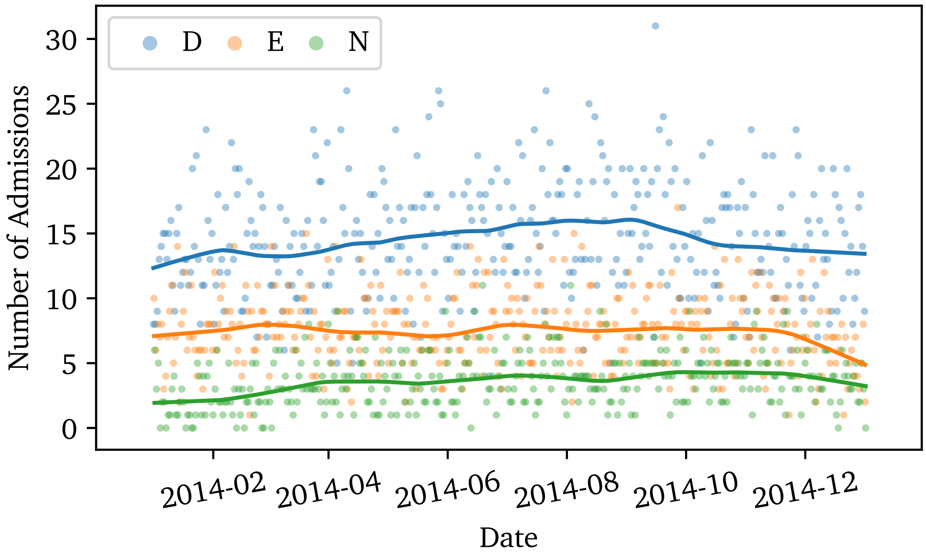


Figure 4.2: Admission Time Series

and the time of year of the day need to be taken into account. To determine whether the day of the week also needed to be accounted for, the distributions of the number of admissions for each day of the week and part combination were visualised in figure 4.3.

Figure 4.3 shows that the distribution of the number of admissions is different for the weekdays compared to the weekend for both the day and evening parts, where the weekend days have fewer admissions on average. The night part does not appear to have as much variability in the number of admissions between the days of the week.

Since the month of the year, the day of the week, and the part of the day have all been shown to influence the number of admissions an appropriate pool to draw samples from for a given part on a given day is the set of the number of admissions in parts of the same type, on the same day of the week, in the same month of the year. When the number of admissions in each part of each day is determined in this way, by randomly resampling from a pool, it is assumed that the number of admissions in any part is independent of the number in any other part. To assess whether this is the case, the auto-correlation of the time series of the number of admissions in each part was calculated and is shown in figure 4.4.

The auto-correlations shown in figure 4.4 are calculated by first removing the seasonal and day of the week effects from the number of admissions in each part,

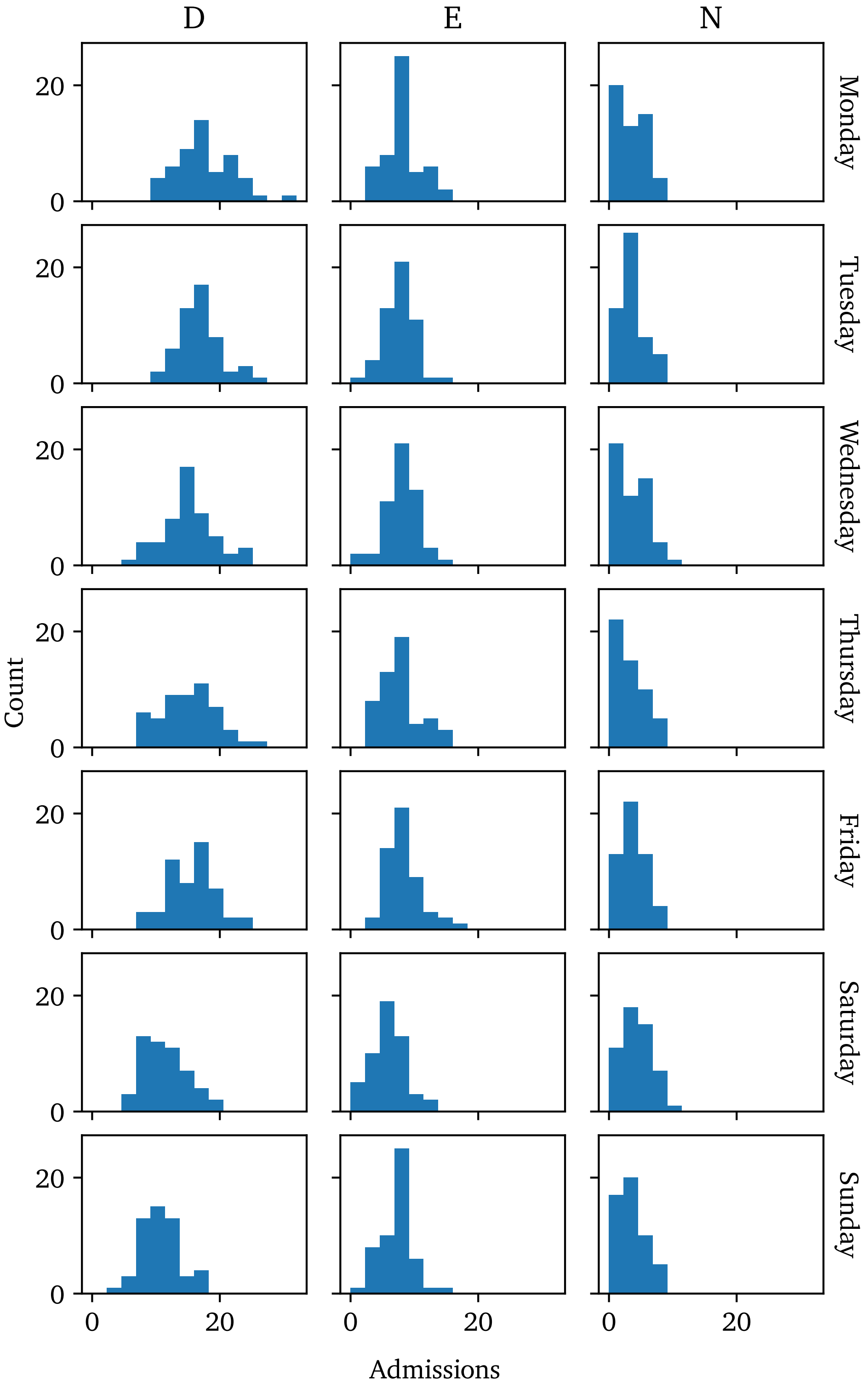


Figure 4.3: Histograms of Admission Counts

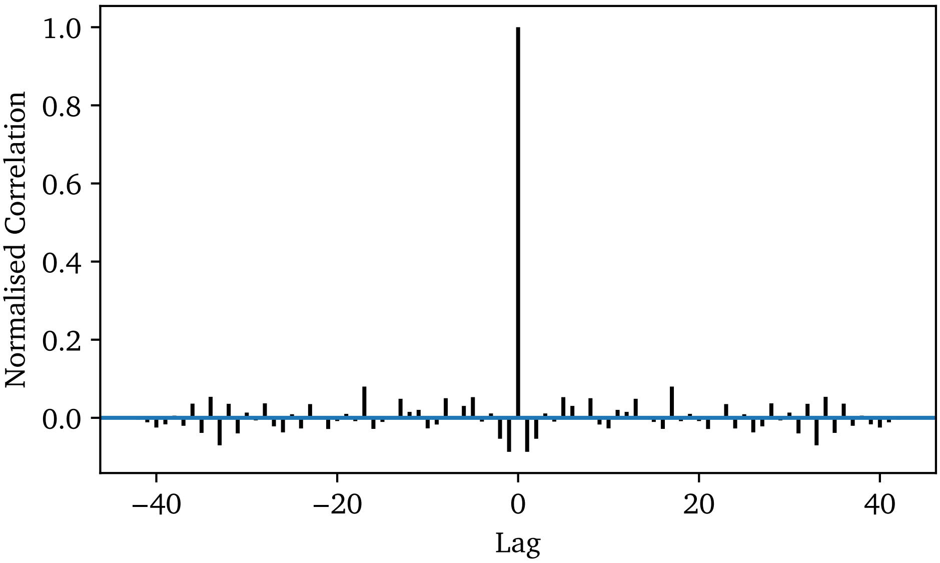


Figure 4.4: Auto-Correlation of Admission Counts

resulting in adjusted admission numbers. The correlations displayed are determined by calculating the correlation between the vector of adjusted admissions and itself with a lag applied, and then normalising resulting correlations. The lags shift the elements in the vector around by a given number of positions (on the x-axis in the graphs). This allows us to see if the number of admissions in a part is correlated with the number of admissions in the parts on either side of it.

The only significant correlations that figure 4.4 reveals are; a slight negative correlation between the number of patients admitted in a part, and the number admitted in the preceding and following parts; and a slight positive correlation with the part 17 parts before and after the given part. However both of these correlations are small, with magnitudes of 0.08. A lack of correlation does not guarantee independence between random variables, only linear independence, however we believe that the assumption that the number of admissions in any given part is independent of the number in any other part is appropriate in this case.

The same analysis was performed for the length of stay of patients in GM. Figure 4.5 shows that both the seasonal variation and the differences between the parts of the day are much smaller for length of stay than for the number of admissions, however we decided to still take the month of admission into account as there is still a small seasonal effect.

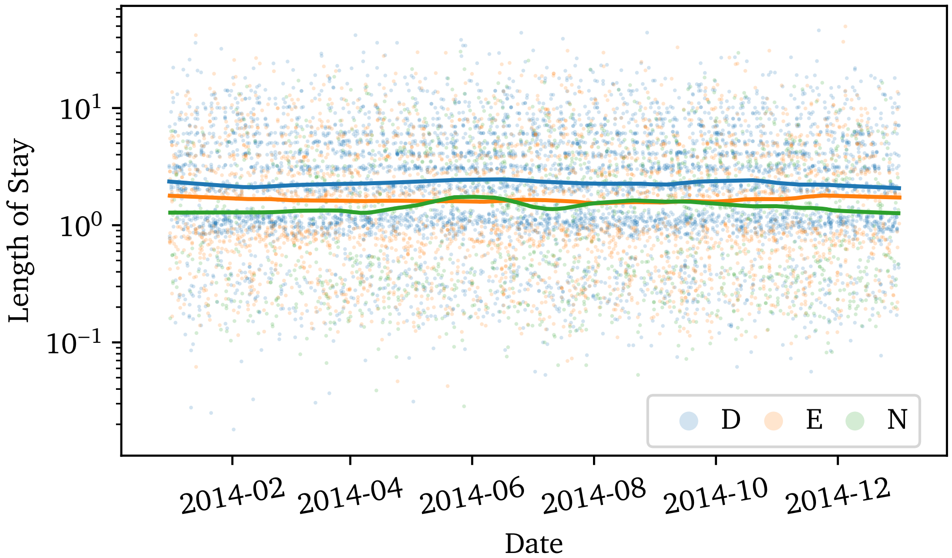


Figure 4.5: Length of Stay Time Series

Figure 4.6 shows that there are differences in the length of stay between the days of the week, and the parts of the day. Length of stays that correspond to patients being discharged during the weekend are much less likely; for example between 5 and 7 days length of stay on Monday, or between 2 and 4 days on Thursday. The pool of samples for the length of stay of a patient that arrives in a given part on a given day was chosen to be the set of lengths of stays of patients that arrived in the same part of the day, on the same day of the week, on days in the same month.

Figure 4.7 shows the auto-correlation of the vector of seasonally-adjusted and normalised length of stays. There is less correlation between neighbouring admissions length of stays, than for neighbouring parts number of admissions. Therefore we once again believe that the assumption that the length of stay of a patient is independent of the length of stay of other patients, which is required for the resampling procedure, is appropriate in this case.

Given the above assumptions the scenario generation process is described in algorithm 4.4.

At the conclusion of algorithm 4.4 we have a new set of patient data that includes the number and class of new admissions to GM, or in other words the value of $A_{k,c,p}$ for all 'New' patient classes. This process was used to generate 2000 patient data scenarios for this case study. In addition the first 1500 scenarios were averaged to create an 'Expected Scenario'. The total number of patients

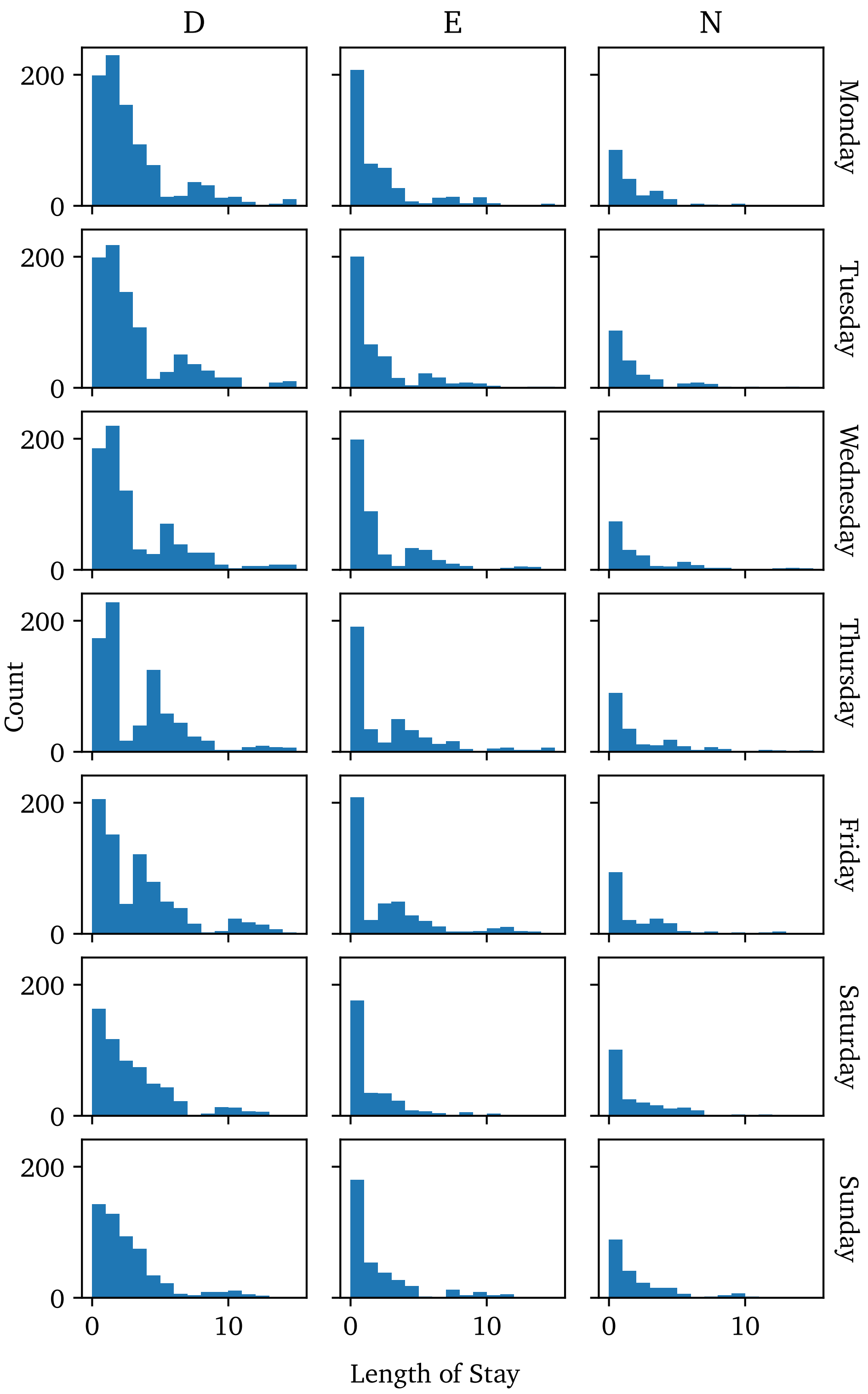


Figure 4.6: Histograms of Length of Stays

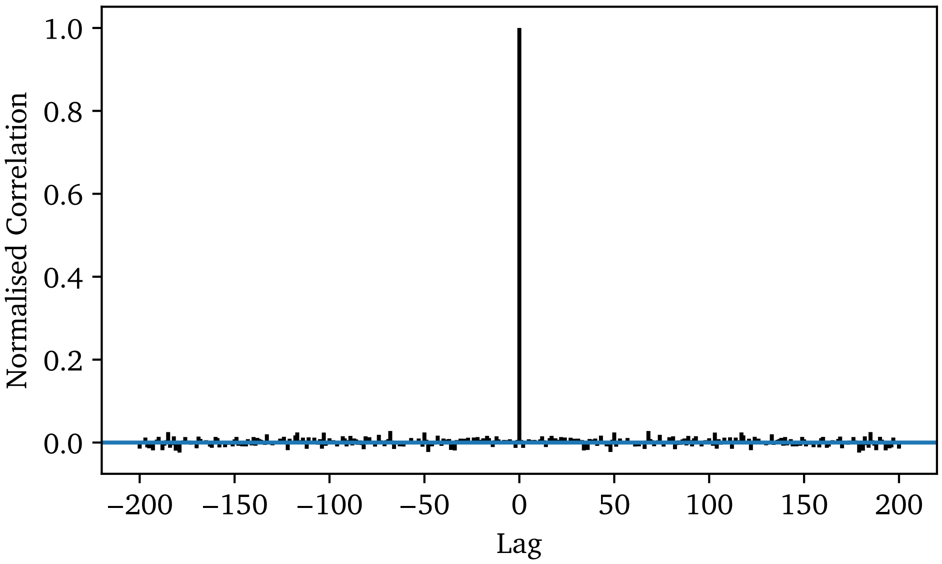


Figure 4.7: Auto-Correlation of Length of Stay

Algorithm 4.4: Scenario Generation Algorithm

Step 1. For each part of each day randomly choose the number of patients that arrive from the set of the number of patients that arrived in the same part of the day, on the same day of the week, on days in the same month of the year.

Step 2. For each of these patients randomly choose, with replacement, a length of stay from the set of lengths of stay of patients that arrived in the same part of the day, on the same day of the week, on days in the same month of the year.

Step 3. For each patient determine the first part of their patient class in 'X/X New' by checking whether their length of stay is greater than (L) or less than (S) 48 hours.

Step 4. For each patient determine the second part of their patient class in 'X/X New' by randomly choosing if they are accurately classified or not based on the classification probabilities given by WTH.

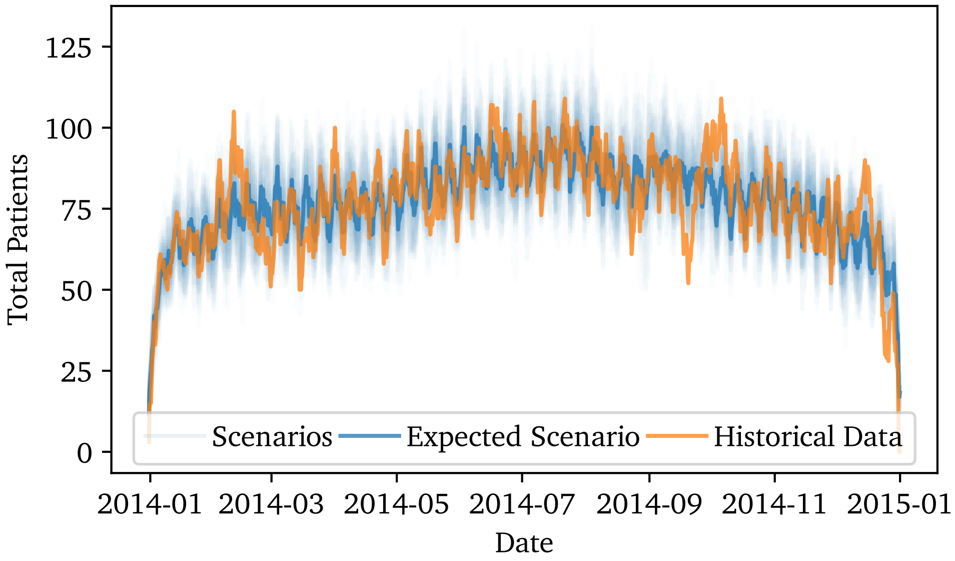


Figure 4.8: Total Patients in 50 Scenarios and from Historical Data

in GM in each part of each day in the planning horizon is shown in figure 4.8, for 50 scenarios, the expected scenario, and the historical data. Each scenario is shown in light blue, the expected scenario in darker blue, and the historical data in orange.

Once these scenarios are generated the data processing described in the previous section can be applied to create the $\mathcal{B}$ matrix, for each inpatient and ADCU team configuration.

First the number of admissions in each part of each day, $A_{k,c,p}$, for all 'New' classes is calculated from the scenario data using algorithm 4.1. Second the proportion of patients discharged in each part of each day, $Q_{k,c,p}$, is calculated using algorithm 4.2. Third the proportion of patients that each shift admits, $U_{k,c,p,o,s}$, is determined from the number of inpatient ward and ADCU teams. Fourth the proportion of patients transferred, $T_{l,k,c,p}$, is calculated using algorithm 4.3. Fifth and finally, the matrix $\mathcal{B}$ is determined by substituting the values of these parameters into equations (3.10). At the end of this process we now have, for each configuration of the GM teams and for each scenario, a matrix $\mathcal{B}$ that links the roster of the GM physicians to their workloads. This matrix can then be used in the sub-problems of the Benders' decomposition.

4.1.5 Benders' Decomposition Implementation

As discussed in the literature review, improving the performance of Benders' decomposition has been the subject of a considerable amount of research. One of the most important factors influencing the decomposition's performance is the decision of where and when in the solution process Benders' cuts are generated. Here this decision is referred to as the 'cut frequency' of a particular implementation of Benders' decomposition; we investigate five different cut frequencies.

Another decision that contributes to the performance of the decomposition is whether to use a single cut aggregated over the sub-problems (SPs) or to disaggregate the cut and add one cut for each SP. The use of both aggregated and disaggregated cuts with each of the five frequencies is investigated.

The first frequency, called 'Optimal', is an implementation of the classical Benders' algorithm where the restricted master problem (RMP) is solved to optimality (hence 'Optimal') at each iteration and cuts (either aggregated or disaggregated) are generated from this solution. These cuts are then added to the RMP which is solved again in the next iteration.

The remaining four frequencies all solve a single branch and bound tree but differ in where in the tree cuts are generated.

The second frequency, called 'Integer', begins with the RMP with no Benders' optimality constraints as defined by (3.29). Cuts are then generated only when an integer solution is found and are added to the problem as 'lazy' constraints through a callback routine. 'Lazy' constraints are similar to traditional cutting planes with the added ability to remove parts of the feasible region. However, due to the dynamic nature of 'lazy' constraints, some reductions and transformations performed by state-of-the-art mixed integer programme (MIP) solvers cannot be used in conjunction with 'lazy' constraints.

The third frequency, called 'RootInteger', is the same as the second frequency except instead of starting with no Benders' optimality constraints an initial pool of cuts is included. These cuts are generated by using the root relaxation as the RMP in the first frequency, and recording the cuts that are required to solve this linear relaxation. Subsequent cuts are added in the same way, only at integer nodes.

The fourth frequency, called '100Nodes', is the same as the third frequency except cuts are generated not only at integer nodes but also at every 100th node, even if it is fractional. Every 100 nodes was chosen as a middle ground between just the root node, and the next frequency which is at every node.

The fifth and final frequency, called 'AllNodes', is the same as the third frequency except cuts are generated at all nodes in the branch and bound tree.

For each of these solution methods the Benders' cuts are generated using the methods described in section 3.3. For the methods that use aggregated cuts, the cut described in (3.31) is used, and the cut described in (3.41) is used for disaggregated cuts.

For both aggregated and disaggregated cuts each SP first requires the calculation of the matrix multiplication $\mathscr{B}^{(z)}\bar{x}$, which then provides the vector $N^{(z)}_{l,k,d,p}$. For each workload group, in each part of each day a sum over the patient classes is then performed to calculate their total workload, which gives another vector $\hat{N}^{(z)}_{l,d,p}$.

Next, the part, day, and workload groups where the largest difference in workloads occurs are found by subtracting the minimum workload from the maximum on each part of each day, as in (3.32)–(3.34). Once the part and day on which the greatest workload difference occurs have been identified, and the workload groups this difference occurs between have been determined, the Benders' cuts can be formed. The method for creating the disaggregated cuts is given in algorithm 4.5.

Algorithm 4.5 requires the $\mathscr{B}$ matrix for each SP. For a problem with 12 workload groups, 10 patient classes, 3 parts in each day, and a planning horizon of 1 year (364 days), this matrix will have $12 \times 10 \times 3 \times 364 = 131{,}040$ rows. If there is also a roster group with 12 physicians and 6 possible shifts on the 7 days of the week there would be $12 \times 6 \times 7 = 504$ columns. However, because the admission shift is the only shift that has non-zero values of $U_{k,c,p,o,s}$ – it is the only shift that contributes to the physicians' workloads – this can be reduced to $12 \times 7 = 84$ columns. This results in each matrix requiring approximately about 15 megabytes of data storage. If we wished to solve a problem with 100 scenarios then 1.5 gigabytes of memory would be required just to store the scenario matrices. Instead of reading all of the matrices into memory at the beginning of solving a problem, each matrix is only read in when it is needed to solve a SP. Constantly reading the matrices does slow down the solution process, however it enables problems with more scenarios to be solved as they do not all have to be stored in memory while solving the problem.

Algorithm 4.5: Benders' Disaggregated Cut Algorithm

Step 1. Given a scenario, z, the matrix for the SP corresponding to that scenario, $\mathcal{B}^{(z)}$, and the current solution to the RMP, $\bar{x}$, calculate $N^{(z)} = \mathcal{B}^{(z)}\bar{x}$.

Step 2. Calculate the total workloads by summing the patient classes, $\hat{N}^{(z)}_{l,d,p} = \sum_{k\in K}\left(N^{(z)}\right)$.

Step 3. Find the day and part on which the maximum difference in workloads occurs using:

$$\bar{c}^{(z)}, \bar{p}^{(z)} = \arg\max_{c\in C, p\in P}\left(\max_{l\in L}\left(\hat{N}^{(z)}_{l,d,p}\right) - \min_{l\in L}\left(\hat{N}^{(z)}_{l,d,p}\right)\right).$$

Step 4. Find the workload groups with the largest and smallest workloads using:

$$l^{+(z)} = \arg\max_{l\in L}\left(\hat{N}^{(z)}_{l,\bar{c}^{(z)},\bar{p}^{(z)}}\right),$$
$$l^{-(z)} = \arg\min_{l\in L}\left(\hat{N}^{(z)}_{l,\bar{c}^{(z)},\bar{p}^{(z)}}\right).$$

Step 5. Add the following cut to the RMP:

$$\eta^{(z)} \geq \sum_{k\in K}(\mathcal{B}^{(z)}_{l^{+(z)},k,\bar{c}^{(z)},\bar{p}^{(z)}}x) - \sum_{k\in K}(\mathcal{B}^{(z)}_{l^{-(z)},k,\bar{c}^{(z)},\bar{p}^{(z)}}x).$$

All of the optimisation models are written in Python 2.7 using the 'gurobipy' package to interface with the Gurobi solver, version 7.0.1. Callbacks were used to implement cut frequencies 2-5 which required cuts to be added during the branch and bound process as lazy constraints. Since lazy constraints were used in callbacks the lazy parameter is set to 1 for the MIP models when solving with these frequencies. In addition, all of the problems were solved on a Windows 64-bit computer with an Intel Core i7-4790 CPU @ 3.60 GHz and 8 GB of RAM. Initial attempts were made to solve these problems with Gurobi using the default settings and no decomposition techniques. Standalone Gurobi performed much worse than the techniques presented here taking 8, 48, and 93 seconds to solve problems with 6, 8, and 10 inpatient ward teams respectively and one admissions scenario. In comparison the average of Benders' decomposition based techniques on the corresponding problems is 0.5, 4.5, and 3.1 seconds.

Table 4.9: Maximum and Mean Range of Potential Rosters

Roster	Maximum Range	% Proposed	Mean Range	% Proposed
Proposed: 10 Inpatient – ADCU Configuration 1	18.5	100	10.0	100
Optimised: 10 Inpatient – ADCU Configuration 1	16.4	88	8.6	86
Optimised: 8 Inpatient – ADCU Configuration 2	33.9	183	17.6	176
Optimised: 8 Inpatient – ADCU Configuration 3	25.6	138	15.0	150

4.1.6 Results

Three sets of experiments were conducted to evaluate the performance of the rostering model. First rosters created using just the historical data were compared to a proposed roster. Second rosters created using two objective functions and scenarios of admission and discharge data were compared. Third the effectiveness of different solution approaches were compared.

Historical Data Rosters

The rosters created using the three staff configurations being considered by WTH are compared to the proposed roster. Each of the optimised rosters are created using a single scenario that contains the actual historical data. The planning horizon used for this comparison was 6/01/2014–28/12/2014. The difference between the maximum and minimum workloads in each segment of each day in the planning horizon is referred to as the 'range' of the workloads in that segment. The objective of the optimisation models is to minimise the largest range of the workloads in any segment in the planning horizon. Table 4.9 shows the maximum and mean range of the proposed roster and the optimised rosters, and the percentage in relation to the proposed roster's value.

The Optimised: 10 Inpatient – ADCU Configuration 1 roster performs 12% better than the proposed in terms of the maximum workload range and 14% better in terms of the mean workload range. The other two optimised rosters, which both have 8 inpatient ward teams, performed worse than the proposed roster both in terms of maximum and mean workload range.

Figure 4.9 shows the maximum, minimum, and range of workloads in each segment of each day in the planning horizon for each of the four rosters. The two rosters with 10 inpatient ward teams display much less variability in the maximum workload, which in turn causes less variability in the range of the workloads. There is no obvious time of the year at which the maximum range of workloads occurs for all of the rosters, and each of the rosters have many points in the planning horizon that are close to the maximum range of workloads, rather than one obvious spike.

Scenario Rosters

The comparison using historical data is not entirely fair for the proposed roster. This is because the optimised rosters use the historical data to find the minimum possible largest range of workloads for that data. In other words the optimised rosters are 'trained' on the historical data, and then compared against the proposed roster on that same data. A fairer comparison could be made if the optimised rosters were trained on a set of scenarios and then tested, along with the proposed roster, on another set of scenarios. Utilising training and testing sets of scenarios also allows us to estimate the improvement in objective function value on the test scenarios that is obtained by using the training scenarios individually rather than the expected scenario problem.

To perform the fairer comparison we generated 2000 scenarios using the method described in section 4.1.4. These scenarios were split into two groups: the training group consisting of 1500 scenarios that were used when solving the problems; and the testing group of 500 scenarios that were used to evaluate the solutions found. The 1500 training scenarios were further split into a group of 1000 optimisation scenarios and a group 500 selection scenarios. The 1000 optimisation scenarios were again split into 10 samples of 100 and were used to define 10 optimisation problems with 100 scenarios in each problem. Figure 4.10 depicts how the 2000 scenarios that were generated were split up.

Each of the 10 problems was solved and the 10 optimal rosters were recorded. The performance of each of the 10 rosters was then evaluated on the 500 selection scenarios. The roster with smallest mean objective function value, out of the 10 rosters, was deemed to be the best roster. The process of creating a roster from the 1500 training scenarios is given in algorithm 4.6. Algorithm 4.6 also includes the possibility of using the conditional value at risk (CVaR) objective function at 95%, which means the objective is to minimise the expected value of the largest difference in workloads in the worst 5% of scenarios. The two differences in the process are the objective function in the optimisation problems

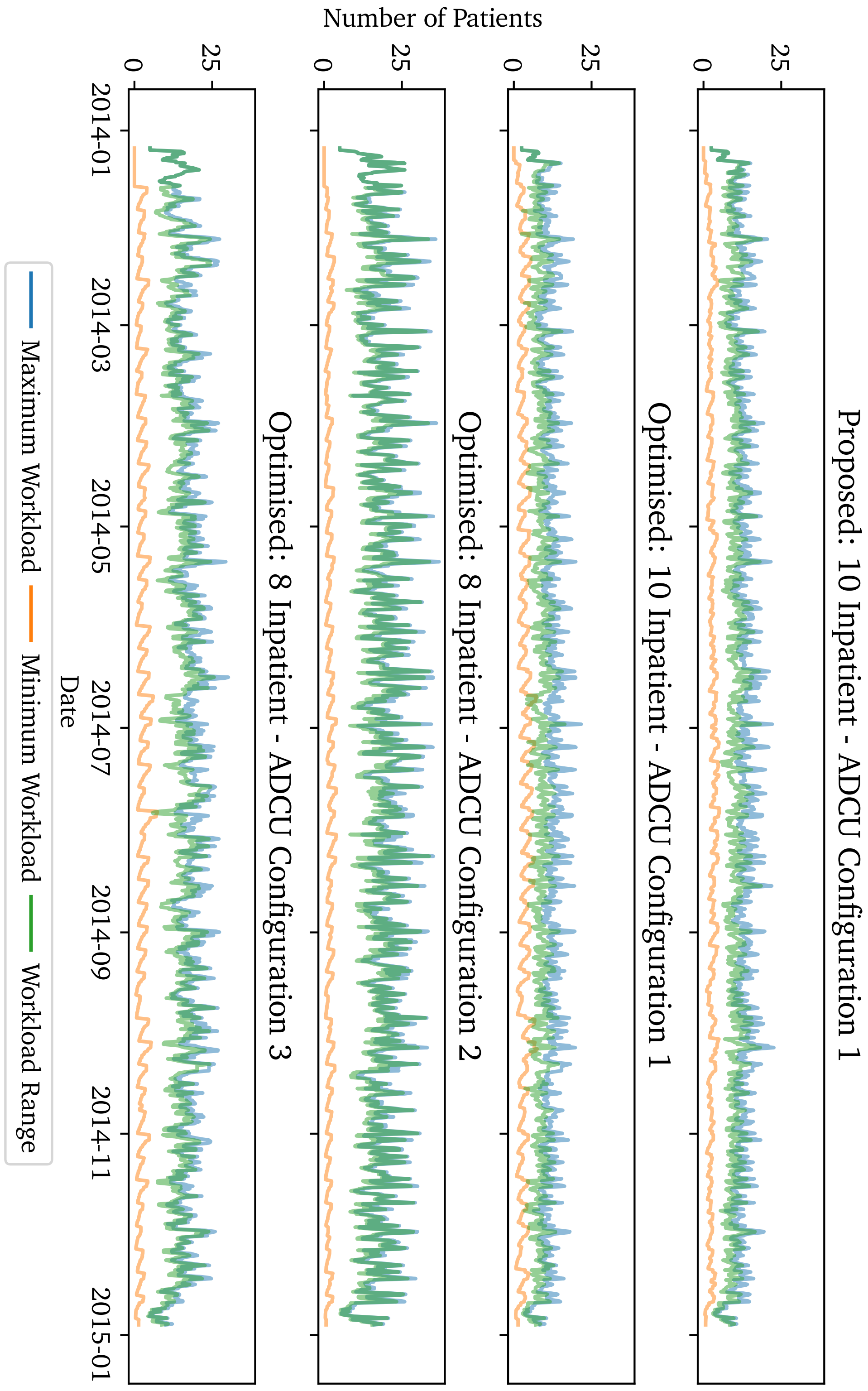
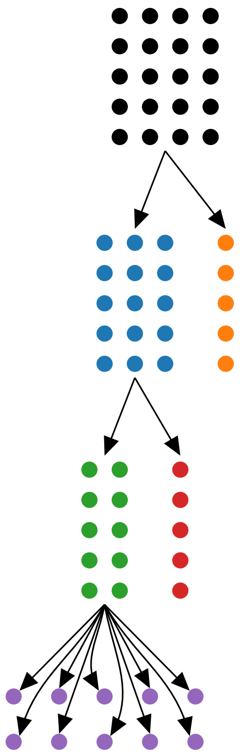


Figure 4.9: Maximum, Minimum, and Range of Workloads



2000 total scenarios were created, each circle represents 100 scenarios

These are split into 1500 training scenarios and 500 testing scenarios

The training scenarios are further split into 1000 optimisation scenarios and 500 selection scenarios

The optimisation scenarios are again split into 10 groups of 100 problem scenarios. Each group of 100 problem scenarios is used in an optimisation model to create a single roster

Figure 4.10: Scenario Usage

and, when choosing the best roster out of the 10 created, the roster with the smallest maximum difference in workloads when considering only the worst 5% of scenarios is chosen. This means that when solving the optimisation problems only 5 scenarios contribute to the objective function (and 25 when evaluating on the 500 selection scenarios).

We first used algorithm 4.6 to create rosters for the three problems for which rosters were previously compared to the proposed roster, that is: 10 inpatient ward teams and ADCU configuration 1, and 8 inpatient ward teams and ADCU configurations 2 and 3. Both the Mean and CVaR objective were used to create rosters for these three problems, resulting in six rosters. This means we have a total of seven rosters to compare: the proposed roster, and two rosters for each problem instance: the Mean roster, and the CVaR @ 95% roster. The workloads of the physicians for these seven rosters were calculated on the 500 testing scenarios and the values of the two objective functions, Mean and CVaR @ 95% were calculated. These results are given in table 4.10. A similar pattern is seen as in table 4.9, where the rosters with 10 inpatient ward registrars perform better

Algorithm 4.6: Roster Creation Algorithm

Step 1. Beginning with the 1500 training scenarios they are first split into two groups: 1000 optimisation scenarios and 500 selection scenarios.

Step 2. The 1000 optimisation scenarios are further split into 10 groups of 100 problem scenarios.

Step 3. Each group of 100 problem scenarios is used to define an optimisation problem to create a roster, as formulated in section 3.2.

Step 4. Each of the 10 problems is solved to create 10 rosters, using either the Mean or CVaR objective.

Step 5. The performance (largest difference in workloads) of the 10 rosters is evaluated on the 500 selection scenarios.

Step 6. The roster with the best performance under the corresponding objective function, either Mean or CVaR, that was used in the optimisation problems is selected as the best roster.

than those with 8. In addition, the rosters trained on 1500 scenarios using algorithm 4.6 perform 8% better than the proposed roster using the Mean objective, and 10% better using the CVaR objective.

Table 4.10: Roster Comparison Over Test Scenarios

Inpatient Teams	ADCU Configuration	Roster	Mean	CVaR @ 95%
10	1	Proposed	20.31	23.28
		Mean	18.67	21.14
		CVaR @ 5%	18.67	21.03
8	2	Mean	37.17	41.97
		CVaR @ 5%	37.28	41.91
8	3	Mean	29.88	34.55
		CVaR @ 5%	29.88	34.55

To evaluate the performance of the model on a wider range of problem instances, in addition to solving the problems that correspond to the staffing configurations being considered by WDHB, problems with 6, 8, and 10 inpatient ward registrars combined with each of the three ADCU configurations were also solved, resulting in nine problem instances in total.

Table 4.11: Mean Objective Function Value Over Test Scenarios

Inpatient Teams	6			8			10		
ADCU Configuration	1	2	3	1	2	3	1	2	3
Mean	35.98	36.45	30.09	36.40	37.17	29.88	18.67	18.81	15.59
CVaR @ 95%	35.98	36.90	30.09	36.40	37.28	29.88	18.67	18.81	15.65
Expected Scenario	36.20	36.66	30.09	36.53	37.69	29.88	19.02	18.84	15.65
Historical Scenario	36.69	36.68	30.78	36.67	37.49	30.13	19.22	18.84	16.03

Table 4.12: CVaR @ 95% Objective Function Value for Each Roster Over 500 Scenarios

Inpatient Teams	6			8			10		
ADCU Configuration	1	2	3	1	2	3	1	2	3
Mean	40.78	42.12	34.47	41.11	41.97	34.55	21.14	21.45	18.02
CVaR @ 95%	40.78	41.55	34.47	41.11	41.91	34.55	21.03	21.45	18.04
Expected Scenario	40.97	41.66	34.47	41.24	42.54	34.55	21.82	21.41	18.02
Historical Scenario	41.80	41.55	35.18	41.37	42.42	34.79	21.35	21.41	18.80

For each of these nine instances algorithm 4.6 was applied with both the Mean and CVaR objective functions, which resulted in two rosters for each problem instance, one with the objective to minimise the expectation over all of the scenarios, and the other to minimise the expectation in the worst 5% of scenarios. In addition to these, two more rosters were also found for each of the twelve problem instances. Both of these additional rosters are found by solving a single scenario version of the problem. The first uses an ‘Expected Scenario’ which is an average of all of the 1500 training scenarios. The second uses the historical patient data directly in a ‘Historical Scenario’.

The performance of these rosters on the 500 testing scenarios is given in tables 4.11 and 4.12. For the problems considered there is very little value in solving the problems with many scenarios when compared to the equivalent with an expected scenario. The rosters optimised in expectation over 1500 scenarios only performed 0.60% better than the expected scenario rosters on average, and the rosters optimised for the worst 5% of scenarios performed 0.44% better than the expected scenario rosters in the worst 5% of scenarios. There is a small advantage in using the expected scenario approach rather than the historical data directly, as the expected scenario performed 1% better on average.

Computational Experiments

To evaluate the performance of the cut frequencies and types of cut (aggregated or disaggregated) each combination was used to solve each problem instance, with the number of scenarios varying from 1 to 50. As the number of scenarios increases the same scenarios that were in the previous problem are used again, with one new scenario added.

In addition to the nine problem instances considered in the previous section three more instances were included which correspond to the three ADCU configurations with 12 inpatient ward registrars. These problems were added to test the performance of the methods on more difficult problems. A time limit of ten minutes was enforced for every solve. If a particular combination of cut frequency and type reached the time limit for a given problem instance for five consecutive numbers of scenarios then that combination was not used to solve that problem instance for any larger number of scenarios.

The total time taken to solve each problem instance for each number of scenarios is given in figure 4.11. For the problems with 6 inpatient teams there is little difference between the techniques examined and nearly every technique solves every problem in less than 200 seconds.

For the problems with both 8 and 10 inpatient teams the 'AllNodes' cut frequency performs the worst overall, and disaggregated cuts performs better than aggregated cuts.

For the problems with 12 inpatient teams, 'Integer' with disaggregated cuts is the only combination that solved ADCU configurations 1 and 2 within the time limit for all 50 scenarios, and the only combination that solves more than 1 scenario for ADCU configuration 3 within the time limit.

In addition to the total time taken to solve each problem the following metrics were recorded; the time spent solving the master problem (MP Time), the time spent solving sub-problems (SP Time), the number of Benders' cuts added, and the number of nodes explored in the branch and bound tree(s).

Table 4.13 gives the performance on all twelve problem instances averaged over the five solution techniques, and the number of scenarios. Only problems that were solved to optimality are included, although this skews the averages towards the better performing techniques as they are able to solve more of the problems. As the number of inpatient teams increases so to does the total time to solve the problem, the time spent solving the master problem, and the number of nodes explored. In addition ADCU configuration 3 took longer to solve, and more nodes were explored, for all numbers of inpatient ward teams except 12.

With the exception of the 6 inpatient teams instances, however, as the number of inpatient teams increases the time spent solving the sub-problems and the

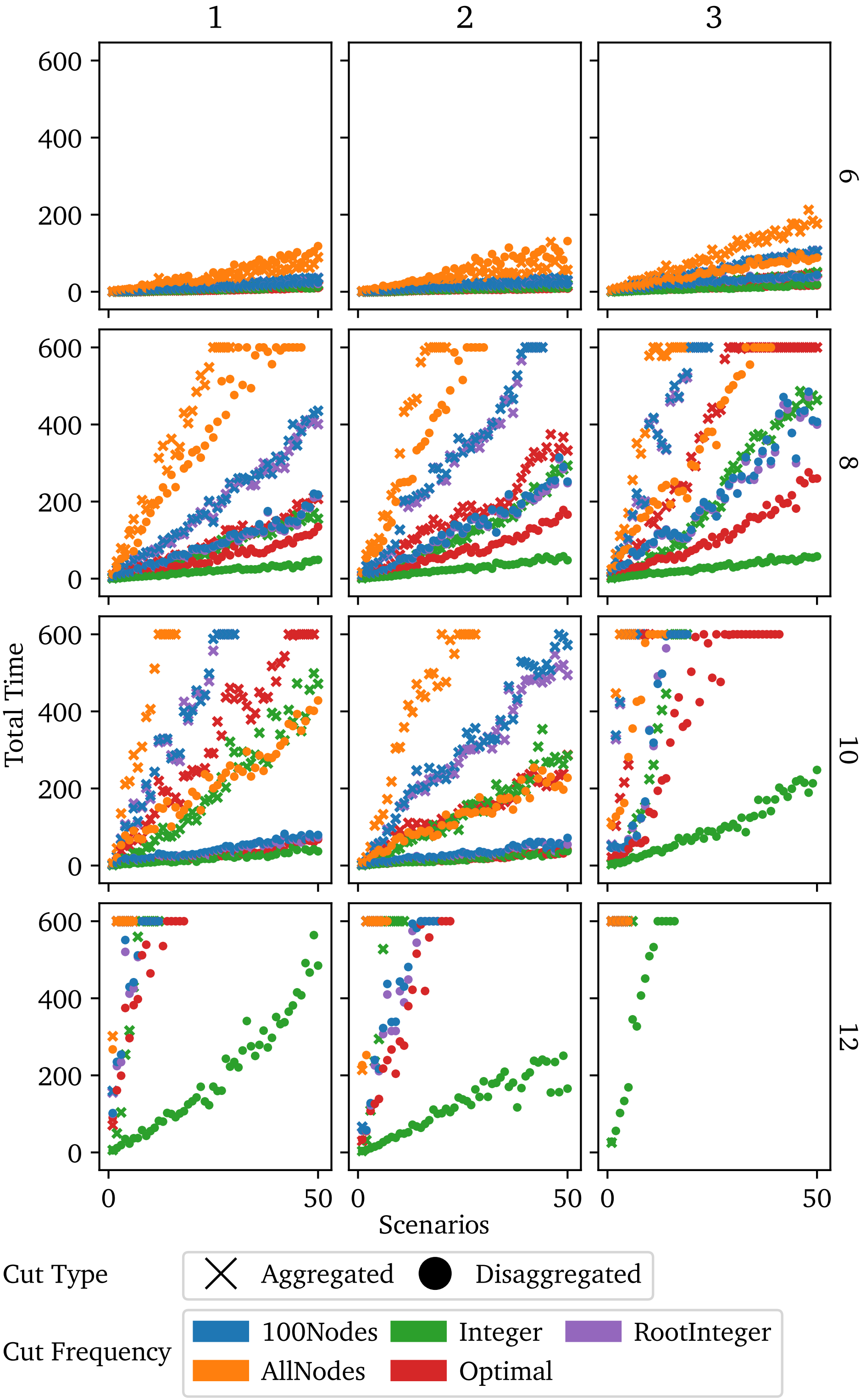


Figure 4.11: Total Time Taken to Solve Each Problem

Table 4.13: Computational Performance by Problem Instance

Inpatient Teams	ADCU Configuration	Total Time	MP Time	SP Time	Benders' Cuts	Nodes Explored
6	1	16	2	14	700	1000
	2	18	2	16	700	1000
	3	37	9	27	600	4000
	Mean	24	5	19	700	2000
8	1	127	44	83	2700	66000
	2	152	69	83	1900	75000
	3	192	113	79	1900	76000
	Mean	155	73	82	2200	72000
10	1	137	50	87	2300	140000
	2	129	31	98	1200	71000
	3	200	150	50	2100	273000
	Mean	143	56	87	1800	127000
12	1	248	199	49	2600	477000
	2	212	163	48	2300	321000
	3	155	131	25	1300	169000
	Mean	225	177	48	2400	383000
Grand Mean		105	47	58	1500	74000

number of Benders' cuts does not increase and the time spent solving the SPs decreases for 12 inpatient teams.

Table 4.14 gives the performance of the solution techniques averaged over the 12 problems and the number of scenarios. Using disaggregated cuts reduces the total time by a factor of 1.5, and the number of nodes explored by a factor of 2.1 on average when compared to aggregated cuts. On the other hand it increases the number of Benders' cuts added by a factor of 3.8. Using disaggregated cuts with the 'Integer' cut frequency has the smallest average total time, however both 'RootInteger' and 'Optimal' spend significantly less time solving SPs and add fewer Benders' cuts.

4.1.7 Discussion

The results of the experiments that were conducted are now discussed, and we once again distinguish between the three sets of experiments that were performed to: compare rosters created with the historical data; compare rosters created with scenarios; compare solution techniques in computational experiments.

Table 4.14: Computational Performance by Cut Frequency and Type

Cut Type	Cut Frequency	Total Time	MP Time	SP Time	Benders' Cuts	Nodes Explored
Aggregated	100Nodes	156	61	95	400	146000
	AllNodes	143	30	113	900	76000
	Integer	105	3	102	300	8000
	RootInteger	149	60	90	400	145000
	Optimal	110	55	55	200	167000
	Mean	130	41	89	400	107000
Disaggregated	100Nodes	87	70	16	700	62000
	AllNodes	155	32	123	10600	20000
	Integer	51	25	26	1900	8000
	RootInteger	85	71	13	400	67000
	Optimal	75	66	9	40	95000
	Mean	86	52	34	2400	50000
Grand Mean		105	47	58	1500	74000

Historical Data Rosters

When comparing the maximum, minimum, and range of workloads for the proposed roster and three rosters optimised on the historical data, figure 4.9 showed that the maximum workload fluctuated much more for the rosters with 8 inpatient teams than those with 10. Figure 4.12 explores this further, showing box plots of the distribution of the maximum, minimum, and range of workloads over the planning horizon for each of the four rosters. The box plots in figure 4.12 are slightly modified from standard box plots so that more box plots can fit side by side in a plot, and colouring of the box plots is more obvious. The same information is still presented as in a standard box plot: the top of the top line (or whisker) represents the maximum value, the bottom of the top line the upper quartile, the dot between the two lines the median, the top of the bottom line the lower quartile, and the bottom of the bottom line the minimum value. This altered version of a box plot is used throughout this thesis and is referred to as a box plot.

The two rosters with 10 inpatient teams have much lower maximums of both workload range, and maximum workload than the two rosters with 8 inpatient teams. In addition the interquartile range of both of these measures is much smaller. This difference is mainly due to the 10 inpatient team rosters having two registrars performing the admitting shift each day, rather than just one each day for the 8 inpatient team rosters. The 8 inpatient team rosters cannot have two teams admitting each day as there are not enough teams to cover two admission

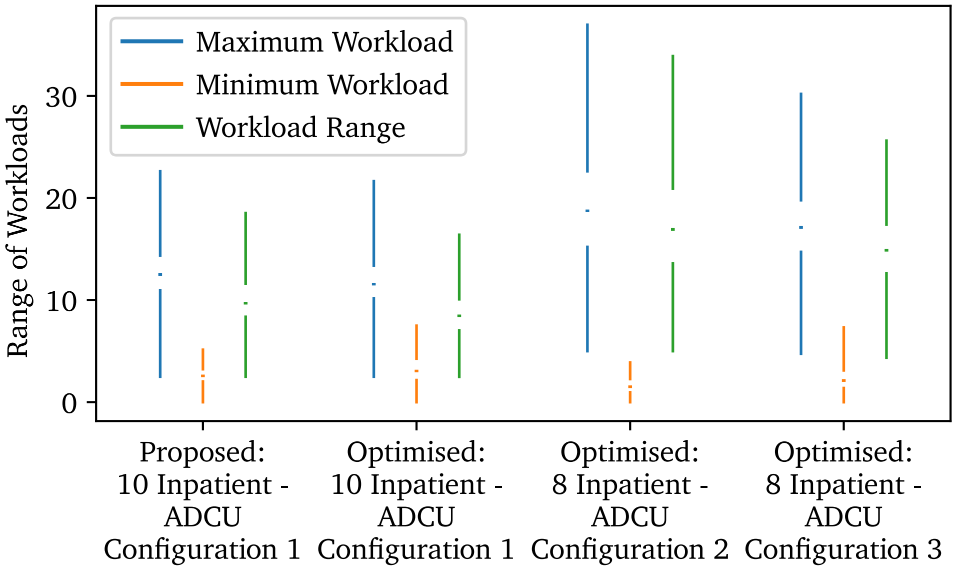


Figure 4.12: Box Plots of Maximum, Minimum, and Range of Workloads

shifts and two post-acute shifts on the weekend, as well as the night shift and having every second weekend off, however the 10 inpatient ward team rosters do.

Having two teams admitting each day means that the patients are split more evenly between the teams, and in particular the workload of the team with maximum workload is reduced on average, and in addition the variability is also reduced as seen by the smaller interquartile ranges, and total ranges. This means that the maximum of the workload range is smaller for the rosters with 10 inpatient teams, which is the objective of the optimisation models. Furthermore the mean and median of the workload range is also smaller for these rosters, and for the optimised 10 inpatient roster in comparison to the proposed roster.

The fact that the mean and median of the workload range is smaller for the optimised 10 inpatient roster than the proposed roster is a side effect, as the optimisation model minimises the maximum workload range. Other measures of the workload range may be of greater importance than the maximum workload range in other situations, and even in the case of addressing continuity of care concerns the maximum workload range is not the only possible objective, as discussed at the beginning of chapter 3. In particular when equity of patient workload between the teams is a priority a measure that is less influenced by extreme values than the maximum, such as the mean or median, or even a risk measure such as CVaR, may be more appropriate.

The three rosters created using the historical data were presented to the management of the GM department at WTH alongside our analysis of the performance of the rosters compared to the proposed roster, the rosters created using generated scenarios were not developed in time to be offered. The rosters were validated by clinicians as being feasible to be implemented in the GM department, and the optimised 10 inpatient roster was selected and put in place by the department.

Scenario Rosters

The results using 2000 scenarios generated from the historical data showed that there is little value in the stochastic solution, and the additional effort required to solve the stochastic problem is probably not worthwhile in these instances. One possible reason for the lack of value in the stochastic solution is that any roster that performs well on a single scenario is likely to perform well on any scenario. This type of behaviour could be because the constraints on the roster do not allow the rosters to be tailored specifically to one scenario, for example all of these rosters are cyclic which limits the ability of the roster to be adjusted to high workload stretches. Another possibility is that the uncertainty represented in the scenarios does not result in large differences between the rosters that are optimal for each scenario, so a roster that is optimal for one scenario is close to optimal for many other scenarios. Further, because the objective is to minimise the largest difference in patient workloads in any of the segments, only the worst segment contributes to the objective function. This means that many of the differences between scenarios do not affect the objective function.

To investigate whether there is a single (or at least a small number) of rosters that are optimal for many of the scenarios the problem with 10 inpatient teams and ADCU configuration 1 was solved using the group of 1000 optimisation scenarios and the Mean objective. However, instead of splitting the scenarios into 10 groups of 100, each scenario was used individually to produce 1000 rosters, of these there were 450 unique rosters. This means that there is not one roster that is best for all scenarios, but some rosters are best in more than one scenario, and the constraints on the roster are, in general, not preventing the roster from being tailored to a specific scenario.

The performance of the 450 unique rosters on the 500 test scenarios was then measured. The best roster had a mean objective value of 18.67 which is the same as the roster that was optimised in expectation. The worst roster had a mean of 25.27, and the average of all the rosters was 19.50. This shows that it is possible to find a roster using just a single scenario that performs well in many other scenarios. Therefore a better approach may be to find many rosters using small samples of scenarios (perhaps up to 10) and then evaluate the performance of these rosters on a much larger number of scenarios (100s if not 1000s), as this is less computationally expensive and leads to rosters of similar quality.

Computational Experiments

The computational experiments using different variations of Benders' decomposition showed that the problems become more difficult to solve as the number of inpatient ward teams and scenarios increases. This is to be expected as increasing the number of inpatient teams increases the number of variables and constraints in the master problem, and increasing the number of scenarios increases the number of sub-problems that need to be solved.

The 'Integer' technique had the smallest average time over all of the problems considered, and solved by far the most 12 inpatient team problems out of all of the techniques. However both 'RootInteger' and 'Optimal' spent less time solving sub-problems and added fewer Benders' cuts than 'Integer'. This suggests that if an improved method of solving the master problem was developed these techniques would benefit the most and may become competitive with 'Integer'.

The slight decrease in time spent solving sub-problems for 12 inpatient teams is likely because most of those problems reached the time limit and if they had been allowed to solve to optimality would have spent more time solving sub-problems.

Nevertheless the results for 8 and 10 inpatient teams suggest that even as the problems become more difficult, about the same number of sub-problems need to be solved, and Benders' cuts need to be added. Therefore to improve performance on even more difficult problems the focus should be on solving the master problem more quickly, for example using a column generation approach.

4.2 Auckland City Hospital

The Auckland District Health Board (ADHB) provides health services to the Central Auckland area. ADHB operates Auckland City Hospital (ACH), which is New Zealand's largest hospital. Just as at Waitakere Hospital (WTH) there is a GM department which diagnoses and treats patients that do not require surgery. Unfortunately the rules for the roster of the physicians in the GM department at ACH constrain the roster in a way that makes minimising the difference in workloads almost redundant.

This is mainly because there is always the same number of physicians taking new admissions from each workload group during the week, and it is only during weekends where there is a difference in patients admitted to each workload group. Even then the weekend duties are completely constrained by the rules and the structure of a cyclic roster.

Therefore the purpose of this section is to provide an additional example of the rostering model applied to a real world hospital department, by describing the rules that govern the roster and deriving the values of the corresponding model parameters. The details of the patient pathways are not included as they do not influence which rosters are possible, only the workloads of the teams.

4.2.1 Problem Description

Unlike at WTH, the patients assigned to GM at ACH are all cared for in the inpatient wards. The structure of the teams is the same as at WTH with the rosters of the registrars once again determining the rosters of the other team members. The physicians who work in GM are divided into teams, the members of the teams that need to be considered for rostering purposes are called registrars. There are 16 registrars working in the GM department, and these registrars are divided into four groups of four. Each of the groups, called White, Gold, Black, and Red, share a single ward in GM and share responsibility for all of the patients that any of the registrars in that group admit.

During the day (8 a.m.–4 p.m.) on weekdays all of the registrars perform their normal duties such as treating, ordering tests for, admitting, and discharging patients. It is only during the evening and night on weekdays, and on weekends, that these duties are assigned to a specific registrar, or registrars. The details of these shifts are given below:

- Long shift (L). The registrar works a long day finishing at 10 p.m. instead of 4 p.m.;
- Night shift (N). The registrar works the night shift 10 p.m.–8 am;
- Sleep shift (Z). After working a week of night shifts registrars are given time to catch up on sleep;
- Weekend One (W1). The registrar works 8 a.m.–4 p.m. on Saturday, and 12 p.m.–10 p.m. on Sunday;

- Weekend Two (W2). The registrar works 12 p.m.–8 p.m. on Saturday, and 8 a.m.–8 p.m. on Sunday;
- Weekend Three (W3). The registrar works 8 a.m.–10 p.m. on Saturday, and 8 a.m.–4 p.m. on Sunday;
- Weekend Four (W4). The registrar works 12 p.m.–12 a.m. on Saturday, and 12 p.m.–10 p.m. on Sunday;
- No shift (X). The registrar is given the day off.

In addition to being a 16 week cyclic roster, the roster of the above shifts for the physicians in GM at ACH must adhere to the following rules:

1. Each weekday one registrar from each of the four colour groups works a long shift;
2. Two long shifts cannot be worked in a row, a long shift also cannot be worked on the Monday after a weekend that has been worked;
3. A night run consists of 5 consecutive night shifts from Sunday to Thursday, followed by three days off.
4. From Monday to Friday of the night run (and one day off) registrars can be assigned two shifts since a relief registrar from another department covers their other duties;
5. One of each of the four types of weekend shifts must be performed each weekend;
6. One registrar of each colour must be working every weekend.
7. Registrars can only work one in every four weekends, this does not include doing the night shift on Sunday.

4.2.2 Parameter Values

We now define the values of the parameters of the rostering model that relate to the roster's rules, which do not include the parameters that come from the patient data, and other parameters required for the patient pathways such as the set of count days, classes of patients and the class transition function.

There is only one group of physicians that need to be rostered, the registrars in the inpatient wards. Therefore the set of roster groups is defined as $O = \{\text{Inpat}\}$. The sets of physicians in the roster group is defined as the set of integers from 1 to 16, so $R_{\text{Inpat}} = \{1, 2, 3, \ldots, 16\}$ and the set of weeks in the roster groups' cyclic roster is equal to the sets of physicians so $W_{\text{Inpat}} = R_{\text{Inpat}}$.

The shifts considered are: long shift, night, sleep, weekend one, weekend two, weekend three, weekend four, and a day off. Therefore the set of shifts for the inpatient teams is defined as $S_{\text{Inpat}} = \{\text{L, N, Z, W1, W2, W3, W4, X}\}$.

Because all of the registrars in each of the colour groups share responsibility for the patients in their ward, there are four workload groups, one for each colour. The set of workload groups is then defined as $L = \{\text{White}, \text{Gold}, \text{Black}, \text{Red}\}$. In the previous case study each registrar had their own workload group, so the order of assigning the registrars to the workload groups did not matter. In this case, however, the order is important because each registrar belongs to one of four workload groups and there are rules that apply over the workload groups.

Since each registrar can only work one in every four weekends, and there must be one registrar from each workload group working each weekend, the feasible orderings are restricted, but there are still several possibilities (4 basic structures are possible). Here we assume that the first four registrars are in the White workload group, the next four Gold, then Black, then Red. Therefore the indicator parameter, $V_{\text{Inpat},r,l}$, that maps physicians to workload groups is one if $1 \leq r \leq 4$ and l = White, or if $5 \leq r \leq 8$ and l = Gold, or if $9 \leq r \leq 12$ and l = Black, or if $13 \leq r \leq 16$ and l = Red. Otherwise it is zero.

Alternatively, the order that registrars are assigned to workload groups are assigned can be modelled explicitly and the indicator parameter replaced by an additional index for the workload group on the shift assignment variable so it becomes $x_{o,l,s,w,d}$, and is 1 if shift s is assigned to day d in week w in the cyclic roster of roster group o, and week w is worked by a registrar in workload group l on the first week of the planning horizon. Hence, the workload group of each registrar in the roster is determined as the roster is created. The general parameters for the ACH problem are summarised in table 4.15.

The values of the parameters $F, B, \mathcal{L}, G, E, \mathcal{R}$, and i, which represent the fixed, banned, linked, following, exclusive, and required shifts, are derived from the rules described in the previous section.

The night run is fixed to start on the Sunday night in the first week, and finish on Thursday night in the second week. The night tour can be fixed here without

Table 4.15: Auckland City Hospital General Sets and Parameters

Symbol	Example
O	{Inpat}
R_{Inpat}	{1, 2, 3, . . . , 16}
W_{Inpat}	{1, 2, 3, . . . , 16}
S_{Inpat}	{L, N, Z, W1, W2, W3, W4, X}
L	{White, Gold, Black, Red}
$V_{\text{Inpat},r,l}$	1 if $1 \leq r \leq 4$ and l = White or if $5 \leq r \leq 8$ and l = Gold or if $9 \leq r \leq 12$ and l = Black of if $13 \leq r \leq 16$ and l = Red and 0 otherwise
D	{Mon, Tue, Wed, Thu, Fri, Sat, Sun}

eliminating any possible solutions, because any solution with the night tour in a different place can be 'rotated' around the cyclic roster so that the night tour once again starts in the first week. Therefore F_{Inpat} must contain all of the shift, week, day tuples that correspond to that night tour, so F_{Inpat} = {(N, 1, Sun), (N, 2, Mon), (N, 2, Tue), (N, 2, Wed), (N, 2, Thu), (Z, 2, Fri), (Z, 2, Sat), (Z, 2, Sun)}.

For the banned shifts it is more convenient to define subsets $b^1 \ldots b^n$ for the roster group 'Inpat', and then let $B_{\text{Inpat}} = \bigcup_{i=1}^{n} b^i$. The night and sleep shifts are only used during and after the night run. This means that they are banned at all other times in the roster so, recalling that $\wedge$ is a logical "and", $b^1 = \{(s, 1, d) : s \in \{\text{N, Z}\} \wedge d \in \{\text{Mon, Tue, Wed, Thu, Fri, Sat}\}\}$, $b^2 = \{(\text{Z}, 2, d) : d \in \{\text{Mon, Tue, Wed, Thu}\}\}$, $b^3 = \{(\text{N}, 2, d) : d \in \{\text{Fri, Sat, Sun}\}\}$, and $b^4 = \{(s, w, d) : s \in \{\text{N, Z}\} \wedge w \in W_{\text{Inpat}} \setminus \{1, 2\} \wedge d \in D\}$. In addition the L shift cannot be worked on weekends, and the W1, W2, W3, W4, and X shifts cannot be assigned during the week so $b^5 = \{(\text{L}, w, d) : w \in W_{\text{Inpat}} \wedge d \in \{\text{Sat, Sun}\}\}$, and $b^6 = \{(s, w, d) : s \in \{\text{W1, W2, W3, W4, X}\} \wedge w \in W_{\text{Inpat}} \wedge d \in \{\text{Mon, Tue, Wed, Thu, Fri}\}\}$.

Linked shifts are used to make sure that when a registrar works a weekend shift on Saturday they work the same numbered weekend shift on Sunday. Therefore $\mathcal{L}_{\text{Inpat},s,w,\text{Sat}} = \{(s, w, \text{Sun})\} \forall s \in \{\text{W1, W2, W3, W4}\}, w \in W_{\text{Inpat}}$.

Following shifts are defined so that if a registrar works in a weekend (not including the night shift on Sunday), then they must have the following three weekends off. They can have the following weekends off even if they did not work the that weekend, however. The following shifts are therefore defined as

$G_{\text{Inpat},s,w,\text{Sat}} = \{(\text{X}, f_3(w+\delta, \text{Inpat}), \text{Sat}), (\text{X}, f_3(w+\delta, o), \text{Sun}) : \delta \in \{1, 2, 3\}\} \forall s \in \{\text{W1, W2, W3, W4}\}, w \in W_{\text{Inpat}}$.

The exclusive shifts prevent two long shifts from being worked two days in a row, or after a weekend. If a long shift (or a weekend shift on a Sunday) is worked then the long shift on the next day is the only element of the only subset of the set of exclusive shifts for that long shift. Therefore $E_{\text{Inpat},\text{L},w,d_1} = \{\{(\text{L}, w, d_2)\}\} \forall w \in W_{\text{Inpat}}, (d_1, d_2) \in \{(\text{Mon, Tue}), (\text{Tue, Wed}), (\text{Wed, Thu}), (\text{Thu, Fri})\}$, and $E_{\text{Inpat},s,w,\text{Sun}} = \{\{(\text{L}, f_3(w + 1, \text{Inpat}), \text{Mon})\}\} \forall w \in W_{\text{Inpat}}, s \in \{\text{W1, W2, W3, W4}\}$.

Required shifts are used for the inpatient roster group to ensure that one of each weekend shift is worked each weekend, and they are used for the workload groups to ensure that one long shift is worked each day, and a weekend shift is worked each weekend by a member of the workload group. Therefore $\mathcal{R}_{\text{Inpat}}$ is $\{(s, d) : s \in \{\text{W1, W2, W3, W4}\} \wedge d \in \{\text{Sat, Sun}\}\}$, and $i_{\text{Inpat},s,d} = 1 \forall s \in \{\text{W1, W2, W3, W4}\}, d \in \{\text{Sat, Sun}\}$. In addition the required shifts for the workload groups are defined as $\mathcal{R}_l = \{(\text{L}, d) : d \in \{\text{Mon, Tue, Wed, Thu, Fri}\}\} \cup \{(s, d) : s \in \{\text{W1, W2, W3, W4}\} \wedge d \in \{\text{Sat, Sun}\}\} \forall l \in L$, and $i_{l,\text{L},d} = 1 \forall d \in \{\text{Mon, Tue, Wed, Thu, Fri}\} l \in L$, and $i_{l,s,d} = 1 \forall s \in \{\text{W1, W2, W3, W4}\}, d \in \{\text{Sat, Sun}\} l \in L$.

Finally because Monday to Friday of the night run is covered by a relief registrar two shifts are able to be assigned to the registrars during this period, for the remainder of the roster only one shift can be assigned to each registrar on each day. Therefore $y_{\text{Inpat},2,d} = 2, \forall d \in \{\text{Mon, Tue, Wed, Thu, Fri}\}$, and 11 otherwise. The values of these parameters that are dependent on the hospital's rules are summarised in table 4.16.

This section has presented a second case study for rostering GM physicians, which acts as an example of deriving the parameters required to define the roster's rules given a written description. In particular the use of exclusive shifts and required shifts over workloads groups distinguishes the rules in this case study from those in section 4.1.

Table 4.16: Auckland City Hospital Sets and Parameters Dependent on Hospital Rules

Parameter	Value
F_{Inpat}	{(N, 1, Sun), (N, 2, Mon), (N, 2, Tue), (N, 2, Wed), (N, 2, Thu), (Z, 2, Fri), (Z, 2, Sat), (Z, 2, Sun)}
b^1	$\{(s, 1, d) : s \in \{\text{N, Z}\} \wedge d \in \{\text{Mon, Tue, Wed, Thu, Fri, Sat}\}\}$
b^2	$\{(\text{Z}, 2, d) : d \in \{\text{Mon, Tue, Wed, Thu}\}\}$
b^3	$\{(\text{N}, 2, d) : d \in \{\text{Fri, Sat, Sun}\}\}$
b^4	$\{(s, w, d) : s \in \{\text{N, Z}\} \wedge w \in W_{\text{Inpat}} \setminus \{1, 2\} \wedge d \in D\}$
b^5	$\{(\text{L}, w, d) : w \in W_{\text{Inpat}} \wedge d \in \{\text{Sat, Sun}\}\}$
b^6	$\{(s, w, d) : s \in \{\text{W1, W2, W3, W4, X}\} \wedge w \in W_{\text{Inpat}} \wedge d \in \{\text{Mon, Tue, Wed, Thu, Fri}\}\}$
B_{Inpat}	$\bigcup_{i=1}^{6} b^i$
$\mathcal{L}_{\text{Inpat},s,w,\text{Sat}}$	$\{(s, w, \text{Sun})\} \forall s \in \{\text{W1, W2, W3, W4}\}, w \in W_{\text{Inpat}}$
$G_{\text{Inpat},s,w,\text{Sat}}$	$\{(\text{X}, f_3(w + \delta, \text{Inpat}), \text{Sat}), (\text{X}, f_3(w + \delta, \text{Inpat}), \text{Sun}) : \delta \in \{1, 2, 3\}\} \forall s \in \{\text{W1, W2, W3, W4}\}, w \in W_{\text{Inpat}}$
$E_{\text{Inpat},\text{L},w,d_1}$	$\{\{(\text{L}, w, d_2)\}\} \forall w \in W_{\text{Inpat}}, (d_1, d_2) \in \{(\text{Mon, Tue}), (\text{Tue, Wed}), (\text{Wed, Thu}), (\text{Thu, Fri})\}$
$E_{\text{Inpat},s,w,\text{Sun}}$	$\{\{(\text{L}, f_3(w + 1, \text{Inpat}), \text{Mon})\}\} \forall w \in W_{\text{Inpat}}, s \in \{\text{W1, W2, W3, W4}\}$
$\mathcal{R}_{\text{Inpat}}$	$\{(s, d) : s \in \{\text{W1, W2, W3, W4}\} \wedge d \in \{\text{Sat, Sun}\}\}$
$i_{\text{Inpat},s,d}$	$1 \forall s \in \{\text{W1, W2, W3, W4}\}, d \in \{\text{Sat, Sun}\}$
$\mathcal{R}_l$	$\{(\text{L}, d) : d \in \{\text{Mon, Tue, Wed, Thu, Fri}\}\} \cup \{(s, d) : s \in \{\text{W1, W2, W3, W4}\} \wedge d \in \{\text{Sat, Sun}\}\} \ \forall l \in L$
$i_{l,\text{L},d}$	$1 \forall d \in \{\text{Mon, Tue, Wed, Thu, Fri}\}, l \in L$
$i_{l,s,d}$	$1 \forall s \in \{\text{W1, W2, W3, W4}\}, d \in \{\text{Sat, Sun}\}, l \in L$
$y_{\text{Inpat},w,d}$	$2 \ \forall w \in \{2\}, d \in \{\text{Mon, Tue, Wed, Thu, Fri}\}$ 1 otherwise

Conclusion

Two case studies of rostering GM have been presented in this chapter, both to assess the value of using historical data in the rostering process, and demonstrate the application of the rostering model to different real world problems.

The first case study described the staff structure and patient pathways in the GM department at Waitakere Hospital (WTH). These descriptions were used, alongside historical patient data, to determine the values of the input parameters required for the rostering optimisation model described in chapter 3. The complete and accurate definition of these parameters is a tedious process, however it enables a simpler formulation of the mathematical model.

The historical data provided by WTH in relation to the number and time of admissions and discharges of patients was also analysed and techniques were developed to generate possible admission scenarios for the optimisation problem from the historical data.

Comparing the rosters created using the optimisation model with one proposed by the hospital showed that the rosters optimised using the historical data performed 12% better than the proposed roster on the historical data. When rosters trained on scenarios we generated from the historical data were compared to the proposed roster on a separate set of scenarios the optimised rosters still performed 8% better using the Mean objective, and 10% better using the CVaR objective. This shows that there is value in using historical data when creating rosters for physicians, at least in this application.

On the other hand, including the uncertainty in admissions into the roster building process using scenarios was not significantly better than using an average of the scenarios. The roster created using a 1500 scenarios performed only 0.6% better than a roster created using an average of those scenarios, and the expected scenario problem in turn only performed 1% better than using the historical data directly.

The Benders' decomposition developed in section 3.3 was shown to solve the optimisation models efficiently. In particular a method was presented that solves the restricted master problem (RMP) in a single branch and bound tree, with no optimality cuts initially included. When integer solutions are encountered in the branch and bound procedure disaggregated optimality cuts are found by solving a sub-problem (SP) for each scenario and are then added to RMP.

The second case study considered a separate rostering problem at Auckland City Hospital (ACH), and defined the parameters for the optimisation model related to the rules of the roster. Unlike the first case study, which investigated the value of utilising historical data and accounting for uncertainty in the rostering process, this case study served to highlight features of the rostering model that were not required in the first case study. We have now completed our discussion of physician rostering and turn our attention in the next chapters to surgery scheduling.

— Chapter 5 —

Surgical Session Scheduling Model

The scheduling of surgical sessions, that is assigning surgical operations to sessions in which they are performed, is a very important process in most hospitals. Scheduling sessions, where a session, as defined in section 2.2, refers to a period with a specified start and end time, in a particular operating room (OR), on a particular day and is sometimes referred to as a 'block' in the literature, is critical both for the welfare of the patients and the finances of the hospital. Inefficient scheduling can lead to patients waiting longer for their operations, potentially leading to further complications, and it also causes under-utilisation of the surgeons and OR suites, which is costly for the hospital. On the other hand overbooking sessions leads to some operations being cancelled if earlier operations run long, and surgical staff working overtime causes additional expenses. Within the New Zealand health system most hospitals are publicly operated and, although cost still plays a large part in their decision making, more weight is placed on the patient-related outcomes. In addition, the New Zealand Ministry of Health has set a target to improve access to elective surgery by increasing the number of operations performed by 4,000 per year (The New Zealand Ministry of Health 2011).

New Zealand OR managers would therefore like to schedule their surgical sessions in such a way that: patients do not wait too long for their operations; the number of operations that are cancelled or moved due to sessions running overtime is reduced; more operations are performed in total; and sessions do not run overtime too often or by too much.

In general, performing more operations in total and preventing patients from waiting too long for their operation go hand-in-hand. If more operations are performed in total over a year then the hospital is able to get through their waiting list more quickly, which reduces the average time that patients wait. However, if maximising the number of operations performed in a given time frame is the sole criteria for scheduling operations then it will always be preferable to complete two short operations over a single long operation. This can lead to unacceptable wait times for long operations. If, instead, the average lateness of all the operations is minimised, shorter operations may still be preferred (depending on the relative cost of delaying the operations), but there is now a penalty for making a long operation wait for a longer time. Another option is to consider the due dates of operations as constraints, however this often leads to infeasibility, where all of the operations cannot be performed before their due date. This is particularly likely to be the case in health systems that are over capacity and have long waiting lists, as is the case in New Zealand.

The approach taken by most OR suite managers, and most of the OR scheduling literature, to address the issue of sessions running overtime is to constrain the duration of operations assigned to a session. At the most basic level this involves constraining the sum of the durations of the operations to be less than or equal to the duration of the session. However this approach does not take into account the uncertainty in the operations' durations and does not afford the management any control over the proportion of sessions that run overtime. We consider four methods to prevent sessions from running overtime, each of which utilises historical patient data in a different way.

Many models proposed in the literature assume, for their scheduling problems, a very limited window in which to schedule the operations – for example the next week or perhaps month. In addition it is assumed that a subset of operations from the total set of operations on the waiting list is selected to be scheduled in that time window. These assumptions are attractive from a computational point of view as they reduce the number of operations and sessions to be considered, and therefore the combinatorial complexity of scheduling the operations. Further it often accurately reflects how the OR management actually performs their scheduling. However these assumptions also raise two questions. First, how are the operations to be performed in the next period chosen from the list of all waiting operations? Second, what if considering more sessions and operations allowed the determination of a better assignment of operations to sessions? We address these issues by considering all operations currently waiting and all future sessions (up to a point far enough in the future that we are confident the session will not be utilised), even if the OR management does not actually plan that far into the future. Considering all operations and sessions provides the added benefit of seeing exactly how long it will take to get through the entire current waiting list. If this is determined regularly it provides the OR management insight regarding whether demand for operations is exceeding resources.

We only consider the scheduling of elective surgeries, which are part of what the New Zealand Ministry of Health calls 'Planned Care services'. These services are defined as 'health care that is provided more than 24 hours after a decision to proceed with treatment' (The New Zealand Ministry of Health 2019a). Emergency and more urgent operations are assumed to be taken care of by dedicated emergency ORs, or even at another medical facility. In addition we assume that the only factor that restricts the sessions that an operation can be assigned to is that the specialty assigned to the session is the same as the specialty of the operation. Other factors, such as the availability of a surgeon to perform an operation, are assumed to not limit the assignment of operations to sessions.

5.1 Problem Description

We assume that there is a set O of operations that need to be performed. Each operation, o, has a related due date d_o, mean duration μ_o, and standard deviation of the duration σ_o.

There is also a set S of sessions in which the operations can be performed. Each session, s, has a start date-time b_s, and length l_s. There is also assumed to be a constant amount of time t between operations within a session, called the turn around time.

There is a cost, $c_{o,s}$, of assigning operation o to session s. The cost is a function of how early or late an operation would be in relation to its due date, if it were performed in a given session. We assume the function is of the form given below:

$$c_{o,s} = \min(b_s - d_o, 0) + (\max(b_s - d_o, 0))^{\lambda}. \tag{5.1}$$

These costs, referred to as the assignment cost of operation o to session s, can be pre-calculated for every operation-session pair. The exponent on the lateness of an assignment, λ, is a parameter of the model, and can be increased to penalise later assignments even more.

We wish to assign all of the operations to a session in a way that minimises the above costs, while ensuring that the operations assigned to a session do not cause the session to run overtime more often that desired. Initially the sum of all of the costs is considered, however we also propose (in section 5.2.1) a risk averse objective where the expected value of costs above a given quantile of cost is minimised.

The sum of the expected durations of the operations (and the turn around times between the operations) assigned to a session is referred to as the ‘assigned session duration’ and denoted by M_s. Similarly the sum of the variances of the durations of the operations assigned to a session is referred to as the ‘assigned session variance’ and denoted by V_s. We note that a distinction is made between the aforementioned assigned session duration, the realised session duration, and the session length. The realised session duration, L_s, is a random variable that represents the time taken to perform all of the operations in the session, given that the durations of each of the operations are themselves random variables. The session length, l_s, is the planned amount of time that is allocated to that session.

Constraints are required to prevent too many operations being assigned to a session, and creating sessions that run overtime too frequently. We consider constraints of the form $M_s \leq f(V_s, l_s, \upsilon)$, where the assigned session duration must be less than a function of the assigned session variance, the session length, and the maximum allowed probability of overtime, υ. We propose four methods to restrict the assigned session duration, which are discussed in section 5.2.2. Each method has a corresponding function f.

Constraints such as those proposed in section 5.2.2 are known as ‘chance constraints’, because they constrain the probability of an event occurring, in this case a session running overtime. Utilisation of chance constraints in a mathematical programme requires the definition of cumulative distribution functions, which in the case of surgery scheduling take into account the uncertainty in the durations of the operations. Scenarios of operation duration can instead be used to account for this uncertainty, however it would require the addition of a constraint on the duration of a session for each scenario. Since cumulative distribution functions that link the operations assigned to a session to the probability that the session runs overtime can be estimated in a straightforward manner, the use of scenarios in surgery scheduling is unnecessary.

5.2 Mathematical Formulation

Given the definitions in the previous section the decision variables of the problem are:

$$x_{o,s} = \begin{cases} 1, & \text{if operation } o \text{ is assigned to session } s \\ 0, & \text{otherwise} \end{cases} .$$

The problem, which we refer to as 'SurSched', can be expressed as the following mixed integer programme (MIP):

$$\text{(SurSched)} \min \sum_{o \in O, s \in S} c_{o,s} x_{o,s}, \tag{5.2}$$

$$\text{s.t.} \sum_{s \in S} x_{o,s} = 1, \ \forall o \in O, \tag{5.3}$$

$$\sum_{o \in O} ((\mu_o + t) x_{o,s}) - t \leq f\left(\sum_{o \in O} (\sigma_o^2 x_{o,s}), l_s, v\right), \ \forall s \in S, \tag{5.4}$$

$$x \in \{0, 1\}. \tag{5.5}$$

5.2.1 Risk Averse Objective Function

In the SurSched problem defined in the previous section, the objective was to minimise the sum of the earliness/lateness of all of the operations; we refer to this objective as the 'Total' objective. If, instead, the management of the ORs is concerned with the patients whose operations are performed the largest amount of time after their due dates, then a risk averse objective such as the conditional value at risk (CVaR) objective given in (5.6) may be appropriate. This objective minimises the expected cost of the operations in the worst $(1-\beta)$% of assignment cost. In this formulation a variable α is used to calculate the β^{th} percentile of the assignment costs, and the variable γ_o is 0 if operation o is not in the worst $(1-\beta)$% of assignment costs, otherwise it is the amount that the cost of operation o exceeds α. The CVaR objective can then be modelled by replacing the objective function in (5.2) with (5.6) given below, and adding constraints (5.7) and (5.8), as shown by Rockafellar and Uryasev (2000):

$$\min \ \alpha + \frac{1}{|O|(1-\beta)} \sum_{o \in O} \gamma_o. \tag{5.6}$$

The following additional constraints ensure the definitions of α, and γ_o:

$$\gamma_o \geq \sum_{s \in S} \left(c_{o,s} x_{o,s}\right) - \alpha \qquad \forall o \in O, \tag{5.7}$$

$$\gamma_o \geq 0. \tag{5.8}$$

5.2.2 Methods for Constraining Assigned Session Duration

Four methods for constraining the assigned session duration are considered. The first two methods are based on the assumption that the durations of the operations are normally distributed, whereas the third and fourth rely on sampling from historical data.

GND Method

Assumption 5.1 (Grand normal distribution). *All operation durations are independent and follow a single normal distribution with mean $\bar{\mu}$ and standard deviation $\bar{\sigma}$.*

Given Assumption 5.1, the first method – called the 'grand normal distribution (GND)' method – calculates the amount of time that should be reserved on average in a session of length l_s such that the session does not run overtime with a probability greater than υ.

Since the durations of the operations are normally distributed, the actual session duration, L_s, is also normally distributed, with mean $n_{l_s,\upsilon}(\bar{\mu} + t) - t$ and standard deviation $\sqrt{n_{l_s,\upsilon}}\bar{\sigma}$, given that at most $n_{l_s,\upsilon}$ operations are assigned to a session of length l_s with overtime probability υ.

The probability that the actual session duration does not exceed the session length can be calculated using a z-statistic, where $\mathcal{Z}\left(\frac{x-\mu}{\sigma}\right)$ denotes the probability that a random variable following a normal distribution with mean μ and standard deviation σ will take a value $\leq x$, and is given by:

$$\mathcal{Z}\left(\frac{l_s - (n_{l_s,\upsilon}(\bar{\mu} + t) - t)}{\sqrt{n_{l_s,\upsilon}}\bar{\sigma}}\right). \tag{5.9}$$

Therefore the expected number of operations assigned can be calculated by equating this probability to one minus the allowed probability of overtime:

$$\mathcal{Z}\left(\frac{l_s - (n_{l_s,v}(\bar{\mu}+t) - t)}{\sqrt{n_{l_s,v}}\bar{\sigma}}\right) = 1 - v, \tag{5.10}$$

which gives

$$n_{l_s,v} = \sqrt{\frac{-\omega\bar{\sigma} + \sqrt{\omega^2\bar{\sigma}^2 + 4(\bar{\mu}+t)(l_s+t)}}{2(\bar{\mu}+t)}}, \tag{5.11}$$

where $\omega = \mathcal{Z}^{-1}(1-v)$. For each session the limit on the assigned session duration can then be calculated as $n_{l_s,v}(\bar{\mu}+t) - t$. Therefore:

$$f\left(\sum_{o\in O}(\sigma_o^2 x_{o,s}), l_s, v\right) = n_{l_s,v}(\bar{\mu}+t) - t,$$

and the constraints in (5.4) are replaced with

$$\sum_{o\in O}(\mu_o + t)x_{o,s} \leq n_{l_s,v}(\bar{\mu}+t).$$

SND Method

Assumption 5.2 (Session normal distribution). *Each operation's duration is normally distributed with its own mean, μ_o, and standard deviation, σ_o, and is independent of the duration of all other operations.*

Instead of Assumption 5.1, i.e., that all operation durations come from a single normal distribution, we can use Assumption 5.2 to define the second method for constraining the assigned session duration, called the 'session normal distribution (SND)' method.

Given an assignment of operations to sessions, $x_{o,s}$, the realised duration of a session, L_s, is normally distributed with mean $\sum_{o\in O}(\mu_o + t)x_{o,s} - t$, and standard deviation $\sqrt{\sum_{o\in O}\sigma_o^2 x_{o,s}}$, because the sum of independent normally distributed random variables is also normally distributed. The probability that a normally distributed random variable with this mean and standard deviation does not exceed l_s is given by:

$$\mathcal{Z}\left(\frac{l_s - \sum_{o\in O}(\mu_o + t)x_{o,s} + t}{\sqrt{\sum_{o\in O}\sigma_o^2 x_{o,s}}}\right). \tag{5.12}$$

If we require this probability to be greater than or equal to $1 - \upsilon$ we obtain the following form for f, where once again $\omega = \mathcal{Z}^{-1}(1 - \upsilon)$:

$$f\left(\sum_{o \in O} \sigma_o^2 x_{o,s}, l_s, \upsilon\right) = l_s - \omega \sqrt{\sum_{o \in O} \sigma_o^2 x_{o,s}} \quad . \tag{5.13}$$

These constraints are non-linear and non-convex so they are approximated using a piecewise linear function, and SOS2 constraints (Beale and Tomlin 1969) are used to implement them in the optimisation model.

DED Method

The third method, called the 'duration empirical distribution (DED)' method, samples sessions and operations from the historical data to build a probability density function to determine a link between the assigned session duration and the probability of running overtime, instead of assuming that the operation durations are normally distributed, e.g., Assumption 5.1 or 5.2.

A sample set of sessions is created to build an empirical conditional probability distribution. This sample contains sessions from two sources. First, historical sessions are used directly, including the operations that were assigned to the sessions. Second, for each historical session operations were randomly assigned to the session. The randomly generated sessions were added to the historical sessions to create a sample that has more uniform distributions of assigned session duration and variance. The combined session sample does not have perfectly uniform distributions, however it is better than each of the other samples individually. Hopefully using this combination removes some of the bias the empirical distribution has towards sessions that are scheduled in the way that has been performed historically by the hospital.

To create the random sessions the following steps were performed for each historical session:

1. A random number of operations was chosen to be assigned to that session, from the historical distribution of the number of operations in a session;
2. Operations were chosen randomly, from the historical operations, to be assigned to that session, up to the previously chosen number.

Including the randomly created sessions gives a sample of sessions that includes two assignments of operations for each historical session, one with the actual historical assignment of operations, and one with randomly assigned operations.

For each session, s, in the historical data, there are two sets of operations: the operations that were performed historically $O_s^{(h)}$; and a randomly assigned set $O_s^{(r)}$. The assigned session duration M_s, and assigned session variance V_s can be calculated for either set of operations as follows:

$$M_s = \sum_{o \in O} (\mu_o + t) - t \qquad \text{and} \qquad V_s = \sum_{o \in O} (\sigma_o^2),$$

where $O \in \{O_s^{(h)}, O_s^{(r)}\}$. Figure 5.1 shows the distribution of the assigned session duration and assigned session variance in the historical sessions, the randomly assigned sessions and the combined session sample, for sessions with a length of 210 minutes.

Realisations of the session duration, L_s, can be created by sampling durations for each of the operations in the session from historical data, and then adding them together along with the turn around times. For each session in the sample 1000 realisations of the session duration were created using 1000 samples of the operation durations from the historical data The method used for sampling the operation durations is not presented here, but is given in detail in algorithm 6.1. Figure 5.2 shows 5 realised session durations for each of the sessions in the sample.

The entire process for creating the data for the empirical distribution for each session in the historical data is given in algorithm 5.1 and realises the operations that were performed historically, $O_s^{(h)}$, and the randomly assigned set of operations, $O_s^{(r)}$, respectively. For each of these two sets of operations both the assigned session duration, M_s, and the assigned session variance, V_s, and 1000 realisations of L_s are calculated.

For the DED method the realisations of session duration are paired with the assigned session duration, and then used to estimate $G_{L_s|M_s}$, the conditional cumulative distribution function of realised session duration given assigned session duration. $G_{L_s|M_s}$ is estimated using the 'KDEMultivariateConditional' function from the 'statsmodels' python package. This function takes the lists of realised session duration and assigned session duration and estimates the conditional CDF at given values of L_s, and M_s using the techniques described in both chapter 6 of Li and Racine (2006), and Liu and Yang (2008). Since we are concerned with

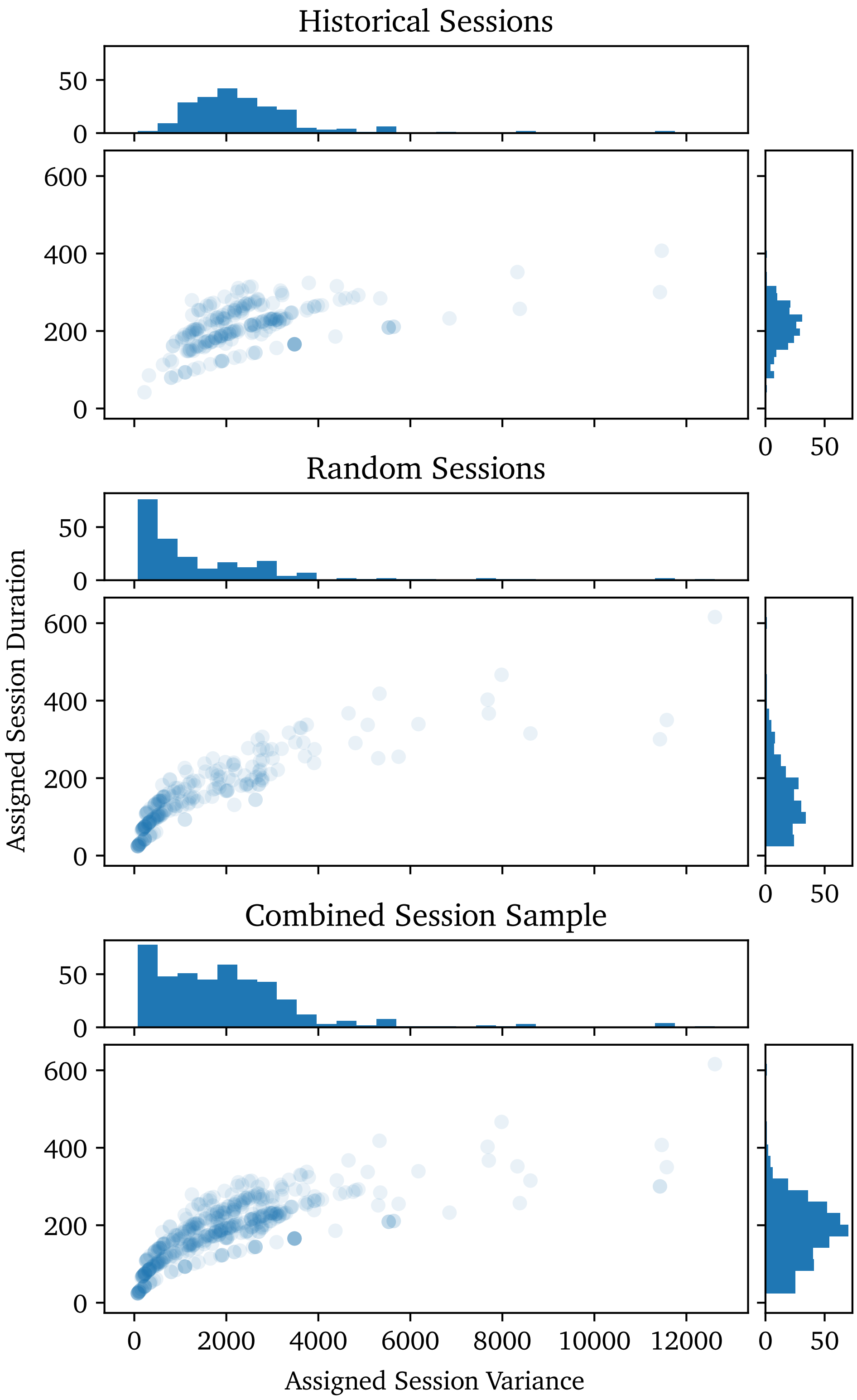


Figure 5.1: Example Sample of Sessions for Empirical Constraint Methods

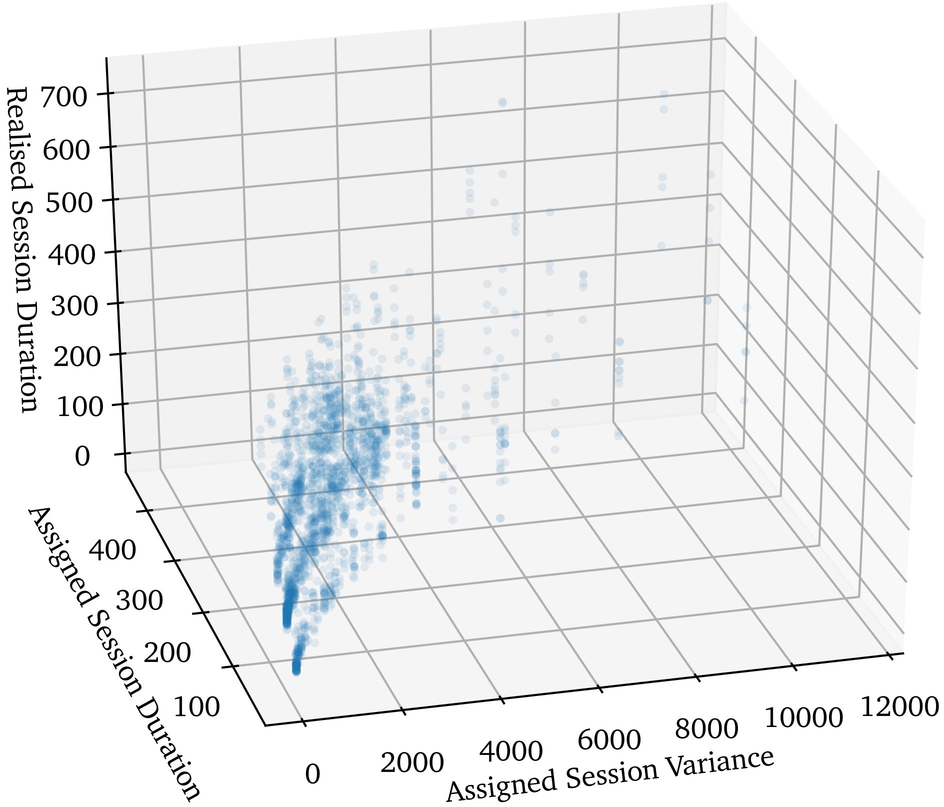


Figure 5.2: 5 Realised Session Durations for Example Session Sample

whether the session runs overtime, we evaluate this conditional CDF at l_s, and 100 values of M_s uniformly spaced from the smallest to largest values in the session sample. This gives us the probability that the realised session duration is less than or equal to the length of the session for 100 values of the assigned session duration. Figure 5.3 shows an example of this conditional CDF and compares it to the probability obtained under the GND assumption (which assumes that all operation durations come from a single normal distribution), for a 210 minute session ($l_s = 210$). We then define another function $h_{l_s}(M_s) = 1 - G_{L_s|M_s}(l_s|M_s)$, which is the probability that a session of length l_s runs overtime given that it has an assigned session duration M_s.

The value of $f\left(\sum_{o \in O} \sigma_o^2 x_{o,s}, l_s, \upsilon\right)$ is then estimated using the inverse of this function, $h_{l_s}^{-1}(\upsilon)$, and interpolating linearly between the values of overtime probability nearest to υ that $h_{l_s}(M_s)$ is estimated at. In relation to figure 5.3 this is equivalent to drawing a horizontal line across the graph at the level corresponding to υ. Then drawing a straight line between the two vertically closest blue points, since the blue points represent the empirical function we are inte-

Algorithm 5.1: Session Sampling Algorithm

Step 1. Given a session, s, determine from the historical data the set $O_s^{(h)}$, the set of operations that were performed in the session historically.

Step 2. Determine the set $O_s^{(r)}$, a set of randomly assigned operations by:

Step 2.1. Choosing a number of operations to be assigned to that session. The number is chosen randomly from the historical distribution of the number of operations in a session.

Step 2.2. Choosing operations randomly, from the historical operations, to be assigned to that session, up to the previously chosen number of operations.

Step 3. For both $O_s^{(h)}$ and $O_s^{(r)}$ determine the assigned session duration using $M_s = \sum_{o \in O_s}(\mu_o + t) - t$ and the assigned session variance using $V_s = \sum_{o \in O_{s(h)}}(\sigma_o^2)$.

Step 4. For both $O_s^{(h)}$ and $O_s^{(r)}$ create 1000 realisations of the session duration L_s by taking 1000 samples of each operation in the set (using algorithm 6.1), and for each sample calculating the resulting session duration using the same formula as for M_s in the previous step.

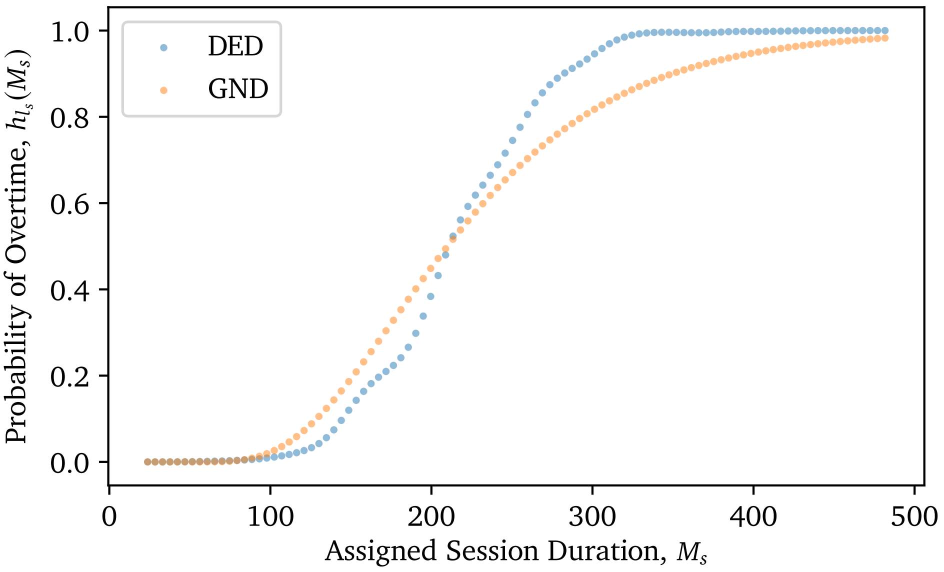


Figure 5.3: Example Overtime Probability Function

rested in. The x-axis value at which these two lines intersect is the value used for $h_{l_s}^{-1}(v)$.

DVED Method

The fourth method, called the 'duration variance empirical distribution (DVED)' method, also uses historical data to estimate the probability of a session running overtime, however instead of being only dependent on the assigned session duration it can now also depend on the assigned session variance.

The same procedure is used whereby the assigned session duration and variance are calculated for each of the historical and randomly generated sessions, and then realisations of the session duration are created by sampling durations for each of the operations. For the DVED method, however, the realised session durations are combined with both the assigned session duration and the assigned session variance to estimate $G_{L_s|M_s,V_s}$, the conditional cumulative distribution function.

Once again the 'KDEMultivariateConditional' function from the 'statsmodels' python package is used to estimate the conditional cumulative distribution function of realised session duration given both assigned session duration and variance, $G_{L_s|M_s,V_s}$, this time on a uniformly spaced grid of both assigned session duration and variance. The probability of overtime given assigned session duration and variance for a particular session length, $h_{l_s}(M_s, V_s)$ is again estimated at each of the points on the grid by subtracting the value of the conditional CDF at that point and session length from 1. An example of the estimated probability of overtime for a 210 minute session given assigned session duration and variance is shown in figure 5.4.

The function, $h_{l_s}(M_s, V_s)$, is converted into constraints for the optimisation problem for a given v by finding, for each 'column' of points corresponding to a single value of assigned session variance, the first point that exceeds the desired overtime threshold. These points define a piecewise linear function of assigned session duration in terms of assigned session variance. These functions are nonconvex and piecewise linear so SOS2 constraints are used to implement them in the optimisation model. To reduce the number of additional variables and constraints used to model these functions as SOS2 constraints a simpler piecewise linear function using a maximum of 11 points was fit to the original functions, and it is this function that is used as $f\left(\sum_{o\in O} \sigma_o^2 x_{o,s}, l_s, v\right)$ in the optimisation model. Figure 5.5 compares an example of these piecewise linear functions for

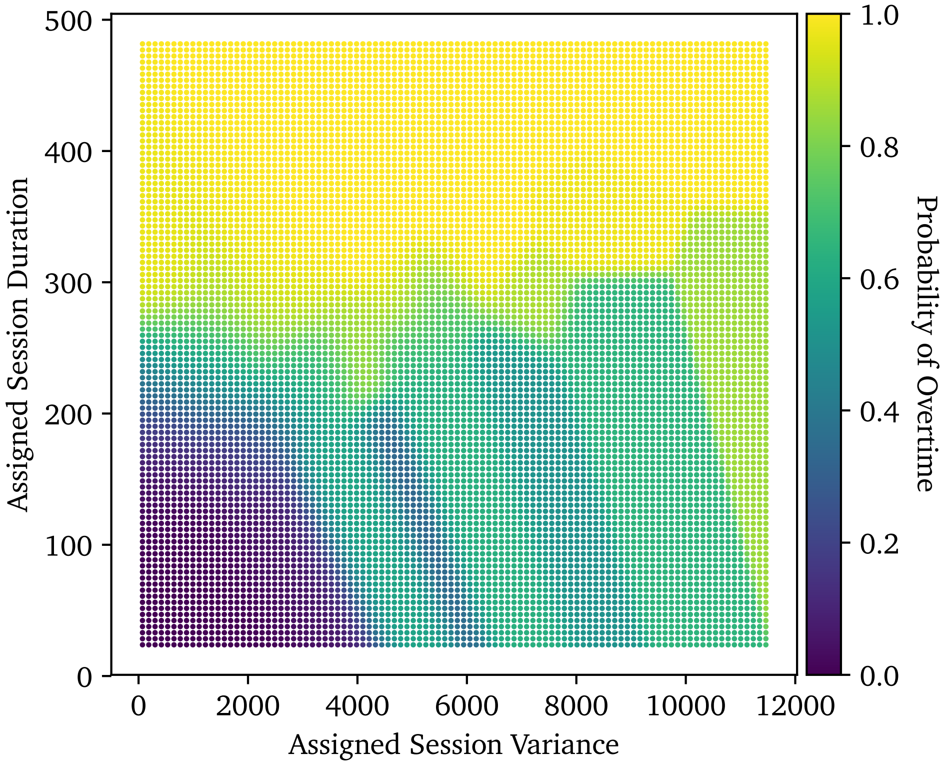


Figure 5.4: Probability of Overtime Given Assigned Session Duration and Variance for a 210 Minute Session

both the DVED and SND methods for five levels of overtime threshold: 0.5, 0.4, 0.3, 0.2, and 0.1.

Generally the constraints formed using the SND method are less restrictive than those formed using the DVED method, however much of the added feasible region is the area with a high assigned session variance and low assigned session duration (in the bottom right of figure 5.5). When scheduling sessions this region is impossible to reach as assigned session variance increases with assigned session duration (as seen in figure 5.1).

Using SOS2 constraints that describe a simpler, fitted function – as described above – is not the only way of obtaining constraints from the overtime probability function shown in figure 5.4. For example another approach could be to take the convex hull of the points for which $h_{l_s}(M_s, V_s) \geq \upsilon$ holds. This method will always provide a constraint that is more conservative than strictly necessary, potentially removing many sessions that would be feasible under the method used here.

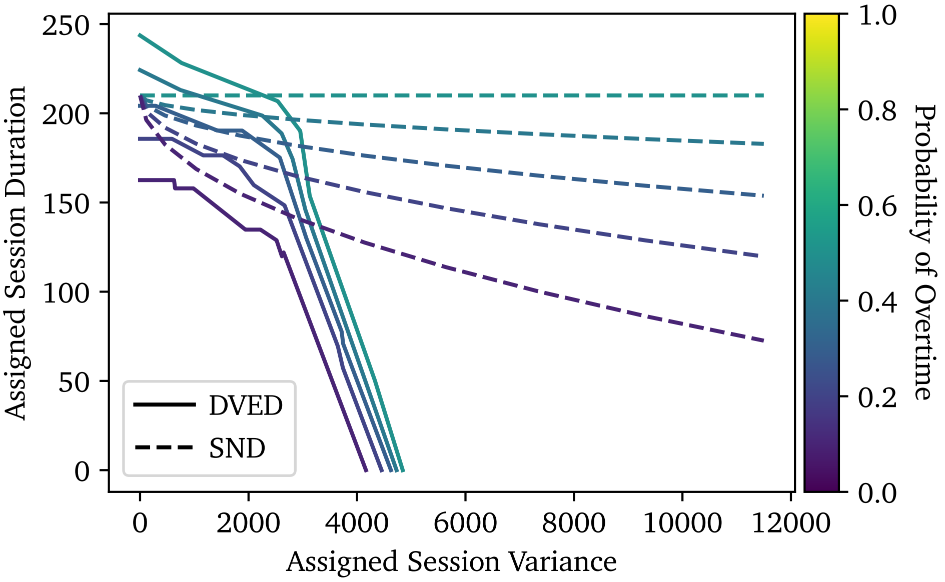


Figure 5.5: Example Constraints for DVED and SND Methods for a 210 Minute Session

5.3 Rescheduling Problem

The surgery scheduling problem described in section 5.1 and formulated as SurSched in section 5.2 assumes that there is a fixed set of operations, O, that need to be scheduled once. The problem that OR management actually face is more dynamic with each day bringing new patients requiring operations. We model this evolving problem as series of optimisation problems at discrete points in time.

Suppose we have a set of points in time I and at each point i in this set the date-time is τ_i. At each point in time i we assume there is a set of operations needing to be performed O_i, and sessions in which to perform them S_i. Given a first time point, $i = 1$, the SurSched problem described in section 5.2 can be solved using the sets O_1 and S_1, where O_1 is the set of all operations that are currently waiting to be performed at that point in time and S_1 is the set of all available sessions in the future. The set of sessions can be truncated at some reasonable point such that there are enough sessions to fit all of the operations, but not so many as to introduce a lot of unnecessary variables into the optimisation problem.

Solving the first problem gives an optimal assignment of operations to sessions, $\hat{x}^1_{o,s}$. If we then step to point $i = 2$ an amount of time $\tau_2 - \tau_1$ will have passed, the sessions where $b_s \leq \tau_2$ will have been completed, and the operations in them will have been performed. We denote the sessions completed and operations performed in the time between points 1 and 2 as $K_1 = \{s \in S_1 : b_s \leq \tau_2\}$ and $P_1 = \{o \in O_1 : \hat{x}^1_{o,s} = 1 \wedge s \in K_1\}$. Note that this assumes that all of the operations assigned to each session are completed, and are never cancelled due to the session running overtime or to enable emergency operations.

In addition we refer to the set of new patients that arrive and need an operation in the time between points 1 and 2 as A_1. If the set S did not include all future sessions then the point in time at which the set was truncated should be moved forward and the extra sessions included are referred to as E_1.

At the point in time $i = 2$ we then have a problem where the set of operations to be scheduled is $O_2 = \{O_1 \setminus P_1\} \cup A_1$, and the sessions in which they are to be performed is $S_2 = \{S_1 \setminus K_1\} \cup E_1$. Further we have the solution from the previous problem $\hat{x}^1_{o,s}$ which allows us to define costs that include a penalty if an operation is moved from the session it was assigned to at time point 1. The new cost of assigning operation o to session s at time point 2 is given below:

$$c^2_{o,s} = \begin{cases} \min(b_s - d_o, 0) + (\max(b_s - d_o, 0))^\lambda, & \text{if } o \in A_1 \\ \min(b_s - d_o, 0) + (\max(b_s - d_o, 0))^\lambda + \rho(1 - \hat{x}^1_{o,s}), & \text{otherwise} \end{cases} \tag{5.14}$$

The SurSched problem described in section 5.2 can then be solved again, this time using the newly defined costs $c^2_{o,s}$ and the sets O_2 and S_2. This procedure is then repeated until the point $|I| + 1$ is reached. The process can be summarised in the steps in algorithm 5.2.

Algorithm 5.2: Rescheduling Algorithm

Step 1. Set $i = 1$.

Step 2. Solve SurSched with costs defined by (5.1) and sets O_1 and S_1, giving an optimal solution $\hat{x}^1_{o,s}$.

Step 3. Increment i ($i = i + 1$).

Step 4. Update sets of operations and sessions, $O_i = \{O_{i-1} \setminus P_{i-1}\} \cup A_{i-1}$ and $S_i = \{S_{i-1} \setminus K_{i-1}\} \cup E_{i-1}$.

Step 5. Update the costs using

$$c^i_{o,s} = \begin{cases} \min(b_s - d_o, 0) + (\max(b_s - d_o, 0))^\lambda, & \text{if } o \in A_{i-1} \\ \min(b_s - d_o, 0) + (\max(b_s - d_o, 0))^\lambda + \rho(1 - \hat{x}^{i-1}_{o,s}), & \text{otherwise} \end{cases}$$

Step 6. Solve SurSched with costs $c^i_{o,s}$ and sets O_i and S_i, giving an optimal solution $\hat{x}^i_{o,s}$.

Step 7. If $i \leq |I|$ then go back to 3, otherwise stop.

Conclusion

In this chapter we presented a surgery scheduling problem where a set of operations needs to be assigned to a set of sessions in which they are to be performed. The objective of the problem is to minimise the sum of how late each operation is performed with respect to when it is due. Historical data is used both to estimate the duration of operations, and the probability that sessions will run overtime. The uncertainty of operation durations is taken into account in the constraints that limit the number and type of operations that are assigned to a session. The nature of the uncertainty is approximated using historical data on the durations of the operations.

We included a risk averse version of the objective which limits the scope of the objective function to only the worst $(1 - \beta)\%$ of operations in terms of lateness cost. This can be used to model a situation where the management of the ORs wish to focus on making sure the latest operations are not too late.

In addition four methods for preventing the overbooking of sessions were presented. All of the methods depend on the allowed probability that a session runs overtime, υ. The methods differ in the way that each of them incorporates the uncertainty of the operation durations into the model. The first two, GND and SND, use historical data to define normal distributions either for session length (GND) or surgery duration (SND). The second two, DED and DVED, use the historical data to define empirical distributions of the probability that a session runs overtime.

Finally, a rescheduling problem was defined, that solves the original scheduling model at consecutive points in time with updated values for the costs, set of operations to be performed, and set of sessions in which to perform them. The updated costs include a penalty that discourages operations from being moved from the session that they were scheduled in at the previous time point.

In the next chapter we apply the model and methods presented here to a case study at Middlemore Hospital (MMH).

— Chapter 6 —

Surgery Scheduling Case Study

In chapter 5 we formulated an optimisation model that assigns a set of operations to a set of surgical sessions, and introduced two objective functions and four methods for constraining the operations assigned to a session. In this chapter we present a case study of a surgical centre in Auckland, New Zealand.

First we give some background on the surgical services offered by Counties Manukau District Health Board (CMDHB), and the facilities they operate. We then analyse the historical data on both operations and sessions that was provided to us, in order to estimate the parameters required for the optimisation model. The results of using the model to schedule a year of operations are then presented and discussed in detail.

6.1 Background

The CMDHB provides health services to the Manukau City, Franklin, and Papakura districts. The population served by the CMDHB has a higher proportion of Pacific Island people and has proportionally more people in the most deprived quintile (The New Zealand Ministry of Health 2019b). In addition the population exhibits high rates of smoking, alcohol consumption, and obesity (Board 2015).

CMDHB operates many facilities throughout the South Auckland region, however this case study focuses on the Manukau Surgery Centre (MSC). The MSC provides day surgery services and elective surgery for patients who are not expected to need intensive care. This means that the MSC does not provide emergency or urgent services.

The MSC has 10 operating rooms (ORs) and 78 inpatient beds that are shared between six surgical specialties. These specialties are; General Surgery, Gynaecology, Orthopaedic Surgery, Ophthalmology, Plastic Surgery, and Urology.

The time in each OR is divided into blocks that are called ‘sessions’. Most of the sessions are either a half-day session (a.m. or p.m.) that lasts 210 minutes, or a full day session that that lasts 480 minutes, however there are some sessions that are slightly different lengths (240 or 510 minutes).

Each session is assigned to one of the surgical specialties and only that specialty can perform operations in that session. This form of scheduling is known as block surgical scheduling and the assignment of the sessions to the specialties is known as the master surgery schedule. For the scheduling problems we assume that the master surgery schedule has already been created. Therefore each specialty has its own set of operations to be performed and sessions in which to perform them, and the scheduling of the operations can be decomposed by specialty. If in practice the sessions and operations are split not by specialty but by surgeon, where each surgeon has a set of operations to perform and a set of sessions in which to perform them, then the scheduling of operations could be decomposed by surgeon rather than specialty. For the purposes of this case study, however, we assume that the sessions and operations are partitioned by specialty. Figure 6.1 shows the master surgery schedule for the MSC in February 2016 from the 15th to the 20th, and table 6.1 gives the number of sessions each specialty was given in 2016 and the number of new operations that needed to be performed during that period. In figure 6.1 each block of colour represents a surgical session in a theatre which was assigned to a specialty, the small rectangles represent either a.m. or p.m. sessions, and the larger rectangles represent full day sessions. It can be seen that some theatres are used more than others, and some are dedicated to a single specialty while others are shared between specialties. Further some theatres are used by different specialties in the morning and afternoon, for example theatre 3 on the 15th. In such situations making sure the probability that the morning session runs overtime is within acceptable limits is important as even a small overrun will prevent the second specialty from starting on time.

To assess the performance of the scheduling model developed in the previous chapter it was used to schedule the operations for General Surgery, Ophthalmology, and Urology throughout 2016. These three specialties were chosen because

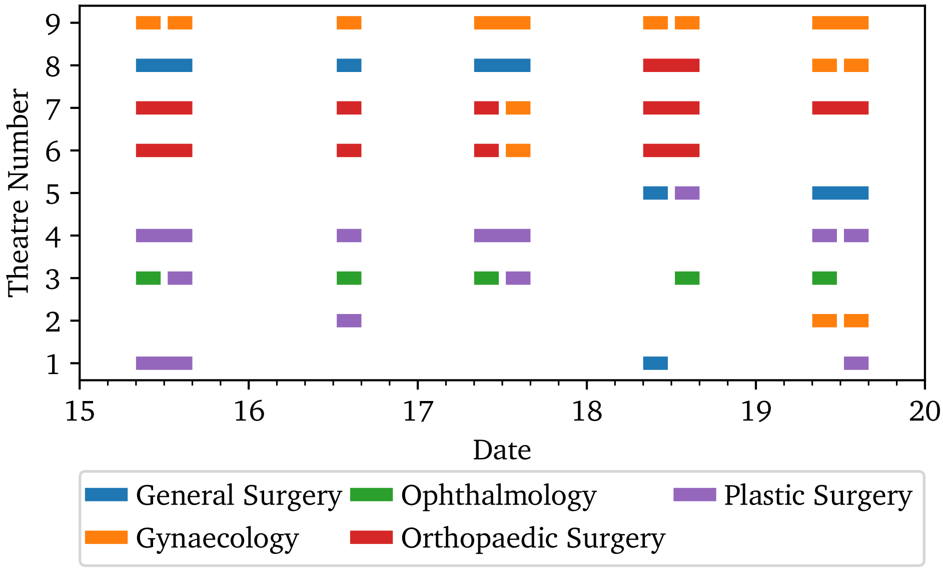


Figure 6.1: Surgical Sessions Coloured by Specialty in February 15th–20th 2016; Each Block Represents a Session in a Theatre

Table 6.1: Number of Sessions and Operations in 2016

Specialty	Sessions	Operations
General Surgery	503	1773
Gynaecology	391	1340
Orthopaedic Surgery	386	1303
Ophthalmology	350	1516
Plastic Surgery	405	1447
Urology	35	269

they represent quite different scheduling problems. Urology is a small specialty that performs relatively few operations per year, making it an easier test problem. General Surgery and Ophthalmology are both larger specialties performing many operations each year, however they are quite different in the average duration of the operations, with Ophthalmology operations requiring much less time than General Surgery operations on average. This means that Ophthalmology needs fewer sessions to perform the same number of operations, and has more operations per session on average.

To make the comparison of the performance of the scheduling model to the actual schedule as fair as possible only information available to the schedulers at the time is used in the model. This means that any estimations of model parameters such as the mean duration of operations, or the empirical distribution functions that constrain the assignment of operations to sessions, are calculated using data from before 2016.

6.2 Historical Data

We were provided with two and a half years of data from CMDHB on the surgical operations that were performed and the sessions they were performed in from January 2015 to July 2017. In an ideal situation every historical operation would come with an estimate of the mean and standard deviation of its duration, as these are both needed for the scheduling model. In practice often all that is available is the amount of time scheduled for the operation, and the actual duration of the operation. Therefore a significant part of this section is dedicated to the estimation of these parameters, even though this is not our primary focus in the process of scheduling operations. The remainder of the section uses the historical data to evaluate the performance of the surgical services of the three specialties of interest during 2016. For each operation in the historical data the following information was available:

- The arrival date, the date that it was added to the waiting list of operations to be performed;
- The due date, the date that the operation should be performed by;
- The surgical specialty that will perform the operation;
- The procedure code, a code such as 'SAB63' that describes the operation that needs to be performed;
- The surgical session that the operation was performed in;
- The scheduled duration of the operation;
- The time that the operation began;
- The actual duration of the operation.

From the operations' data we can directly define the set of operations to be performed for each specialty, O, and the due date of each of the operations d_o. Both the mean duration of each operation μ_o (and the global mean, $\bar{\mu}$, of a particular specialty) and the standard deviation of the duration σ_o (and the global standard deviation, $\bar{\sigma}$, of a particular specialty) require further analysis.

For each session the following information was available:

- The date and time that the session started;
- The surgical specialty performing operations in the session;
- The amount of time that the session was supposed to run for;
- The actual amount of time spent performing operations, including any time between operations.

The session data allows us to define the set of sessions for each specialty, S, the start time of each session b_s, and the length of each session l_s.

To estimate μ_o and σ_o, as well as $\bar{\mu}$ and $\bar{\sigma}$, the operations were categorised by specialty since each specialty has its own scheduling problem. In addition because we will be scheduling the 2016 operations and sessions it would not be a fair comparison if data from this time period was used to inform these estimates, as that data was not available to the schedulers at the time. Therefore we only use data from before 2016 to estimate the parameters, however we do compare this data to that from 2016 onwards to see if there are any significant differences.

First we analysed the durations of the operations of each specialty for both before 2016 and afterwards. Taking the point of view of a scheduler at the beginning of 2016 these two sets are referred to as the ‘Historical’ operations and the ‘Future’ operations.

Figure 6.2 shows histograms of the distributions of the operation durations for each specialty for both the historical and future operations. For the grand normal distribution (GND) method it is assumed that all operation durations come from a single normal distribution with mean and standard deviation equal to the mean and standard deviation of the set of historical operation durations. In order to assess the validity of this assumption the global mean and standard deviation of each set of data $(\bar{\mu}, \bar{\sigma})$ are displayed in addition to the probability density function of a normal distribution with that mean and standard deviation.

Figure 6.2 shows that none of the sets of operations durations are normally distributed. The operation durations for both General Surgery and Ophthalmology have long right tails that increase the mean and standard deviation and result in the approximating normal distribution having a large probability of being less

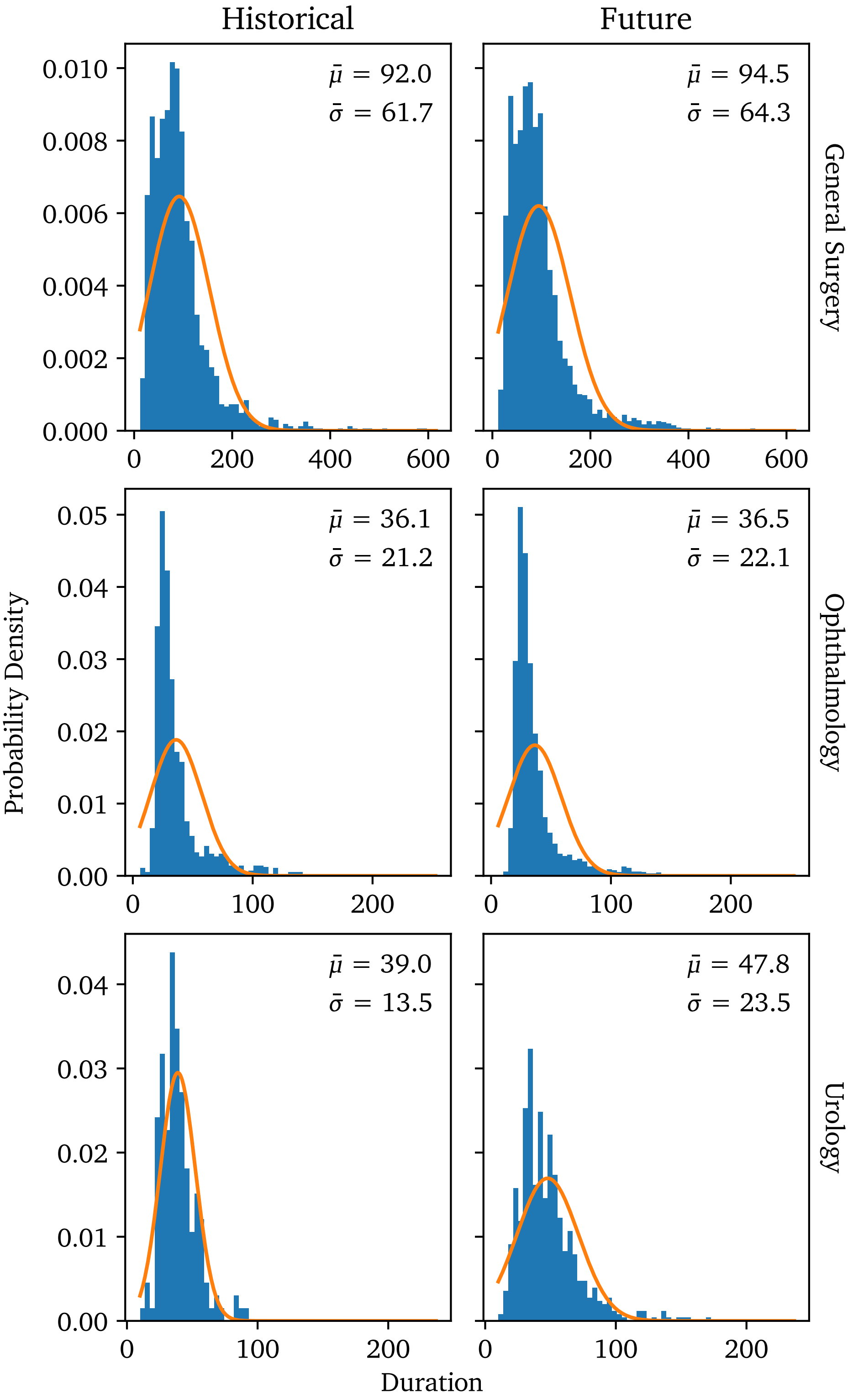


Figure 6.2: Histograms of Operation Duration for Each Specialty

than 0. The Urology operations are closer to being normally distributed, however there is still a large section of each normal distribution probability density function that corresponds to operation durations less than 0.

It is also noted that the mean and standard deviation for the historical and future operations are very similar for both General Surgery and Ophthalmology, but there is a significant difference in these statistics for Urology. This means that estimates made using the historical data for Urology will not be representative of the corresponding estimates for the future Urology data. Since all of the methods for constraining assigned session duration rely on estimates from the historical data, either in the form of normal distribution parameters or empirical distributions of session overtime probability, it would be expected that they do not perform as well for Urology. The reliance on historical data being representative of the future observations could be somewhat alleviated by periodically updating the model parameters that depend on the historical data, as new data becomes available.

We now have estimates for $\bar{\mu}$ and $\bar{\sigma}$ for each specialty, however we still need estimates for μ_o and σ_o for each operation. To estimate μ_o and σ_o the operations are once again split by specialty, because as figure 6.2 shows the distribution of operation duration is different for each specialty. Within each specialty we use a generalised linear model (GLM) that models the duration of the operations as a random variable that follows a gamma distribution. The mean of the gamma random variable is modelled as a so called 'link function' applied to a linear predictor. The linear predictor is the covariates multiplied by a coefficient. The covariates that we used were the procedure code and scheduled duration of the operation, and the link function used was the identity transform. Although the 3-parameter log-normal distribution is more commonly used to model surgery durations (Strum, May, and Vargas 2000; Lamiri, Dreo, and Xie 2007; Oostrum et al. 2008), a gamma GLM was chosen instead as it provided better predictions for the CMDHB data. Further the estimates of the operation durations are estimated here only to serve as input for the scheduling model, and can be replaced with estimates from another model.

GLMs also assume that the variance of the response variable (operation duration) can vary only as a function of the mean of the response variable. Further for a GLM with the response variable modelled by a gamma distribution the variance of the response variable is equal to the mean squared multiplied by a scale factor. Figure 6.3 shows the durations of the operations from before 2016 against the mean of all operations with the same procedure code. Procedure codes where

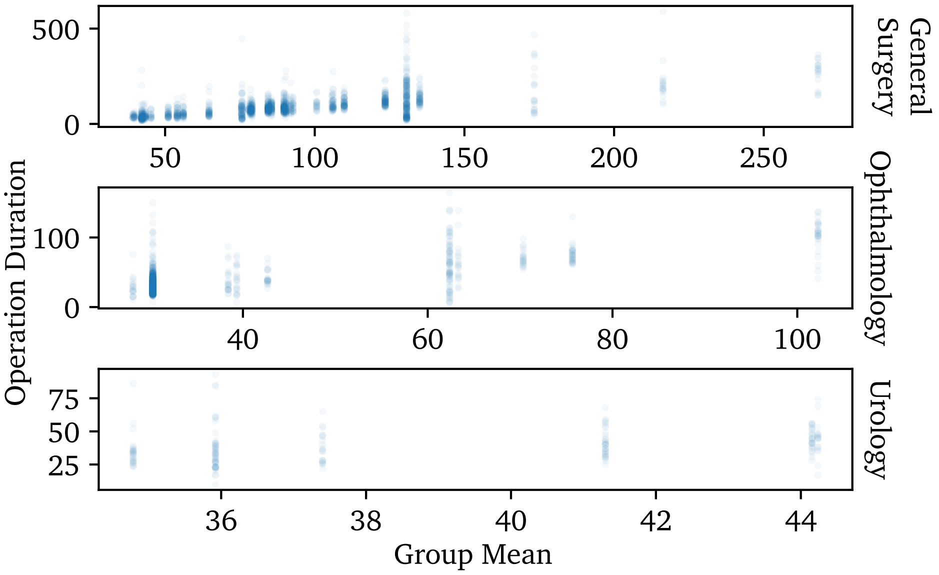


Figure 6.3: Actual Duration of Operations Grouped by Procedure Code

less than 15 operations were observed were grouped together into a general procedure code. For both General Surgery and Ophthalmology variance does appear to increase with the mean duration, although perhaps not quadratically. For Urology there does not appear to be a relationship between group mean and variance, although the range of the group means is much smaller than for the other two specialties.

By modelling the operation durations with a GLM we estimate both the mean and the variance (and therefore standard deviation) of each operation at the same time. However, using a model where the variance of an operation's duration is directly proportional to the mean has several negative consequences. The first is that the session constraint methods that use both the assigned session duration and the assigned session variance (the session normal distribution (SND) and duration variance empirical distribution (DVED) methods) are not being provided with independent pieces of information, and the variance used in these methods could be calculated from the mean. This does not make including the variance useless, as the probability of overtime may have a quadratic relationship with the mean of the assigned operations durations. The second (negative consequence) is that the model assumes that all operations that have the same mean duration have the same variance. This is not always the case in practice, as there may be two types of procedure that take the same time on average but one is very consistent while the other is highly variable.

In addition a contradiction that arises when modelling operation durations using a GLM is that the model assumes that the durations of the operations follow a gamma distribution, whereas both the GND and SND methods assume that the durations are normally distributed. Clearly one of these assumptions is false, and possibly both are.

A GLM has been used before to model the mean and standard deviation of the duration of operations by Shylo, Prokopyev, and Schaefer (2013). We are willing to accept the limitations of this method given that the estimation of the mean and variance of operation durations is only necessary as an input for the scheduling model, and not the main focus of this work.

6.2.1 Resampling Operation Durations

A resampling procedure is used to generate realisations of operation durations. The samples of durations are used both for determining the realised session durations for the empirical distribution methods, and to simulate the sessions obtained by each of the different solution methods.

To perform the sampling for a given operation, a pool of durations to sample from is determined. This pool is the historical operations that share the same procedure code and scheduled duration as the operation for which we wish to sample durations. For a small number of procedure code and scheduled duration combinations (around 10% of the total operations) limiting the pool to only those operations with the same procedure code and scheduled duration leads to a very small pool of possible durations from which to sample. In fact, sometimes there is only the original operation itself.

In the cases where the pool is smaller than 15 durations a different criteria is used. Instead of having the same procedure code and scheduled duration we consider the historical operations that have the same expected duration, as determined by the GLM, as the operation in question. If this pool is still too small then we include operations whose expected durations are within a given threshold of the target operation's. This threshold starts at one minute and is increased a minute at a time until the pool contains at least 15 operations. The resampling process that creates a sample of durations for a given operation is described in algorithm 6.1.

The method in algorithm 6.1 is somewhat unsatisfactory as we are using our previous estimates of expected duration to determine which operations should

Algorithm 6.1: Resampling Algorithm

```
Step 1. Choose an operation $\hat{o}$ with procedure code $p_{\hat{o}}$ and expected duration $\mu_{\hat{o}}$, for which we wish to create a sample of durations.
Step 2. Choose a set of operations $O$ from which the durations are to be sampled.
Step 3. Define the set $O^{(p)} = \{o : o \in O, p_o = p_{\hat{o}}\}$.
Step 4. If $|O^{(p)}| \geq 15$ go to Step 6, otherwise go to Step 5.
Step 5. Set $\Delta = 0$.
    Step 5.1. Increment $\Delta$ ($\Delta = \Delta + 1$).
    Step 5.2. Update $O^{(p)}$, $O^{(p)} = \{o : o \in O, |\mu_{\hat{o}} - \mu_o| \leq \Delta\}$
    Step 5.3. If $|O^{(p)}| \geq 15$ go to Step 6, otherwise go to Step 5.1.
Step 6. Sample operations with replacement from the set $O^{(p)}$ and take the durations of the the sampled operations as the sample of durations for the original operation $\hat{o}$.
```

be included in the pool to sample from, which may bias the sample. Other options were considered for operations where the pool is small, such as: using a parametric distribution such as the normal, triangular or gamma; using the 'General' procedure code instead of the operation's code; or using the entire set of historical operations. All of these options have their own downsides and ideally we would have enough data so that this compromise would not be necessary in the first place; in the absence of sufficient data no solution is ideal. In addition this problem only occurs for a small number of the operations so hopefully our choice for handling the situation has little bearing on the final results.

To summarise, the following steps are performed when scheduling operations given: a starting point in time A, a finishing point in time B, a set of historical sessions that were performed before A, the operations that were performed in those sessions and their durations, the sessions currently planned to be performed between A and B, and the currently known operations waiting to be performed.

1. Use a gamma GLM trained on the operations performed before A to estimate μ_o and σ_o for the operations performed before A and the known upcoming operations;

2. Use algorithm 5.1 to create a sample of session duration realisations;
3. For each of the four session duration constraint methods, GND, SND, duration empirical distribution (DED), and DVED, determine $f\left(\sum_{o\in O}\sigma_o^2 x_{o,s}, l_s, v\right)$, a function of the sum of the variances of the durations of the operations assigned to a session, the length of the session, and the allowed probability of overtime that limits the sum of the means of the durations of the operations assigned to a session, as described in chapter 5;
4. Determine an appropriate set of points in time I (between A and B), and penalty for moving an already scheduled operation to another session ρ;
5. Use algorithm 5.2 to schedule the operations between A and B.

6.2.2 Actual Performance in 2016

To assess the actual performance of the surgical services in 2016 the number of patients waiting for their operation and the total expected duration of those operations throughout the year are shown in figure 6.4.

Both General Surgery and Ophthalmology have between 300 and 400 operations waiting and remain relatively constant throughout the year. Urology begins with 28 patients waiting for operations and increases constantly throughout the year,

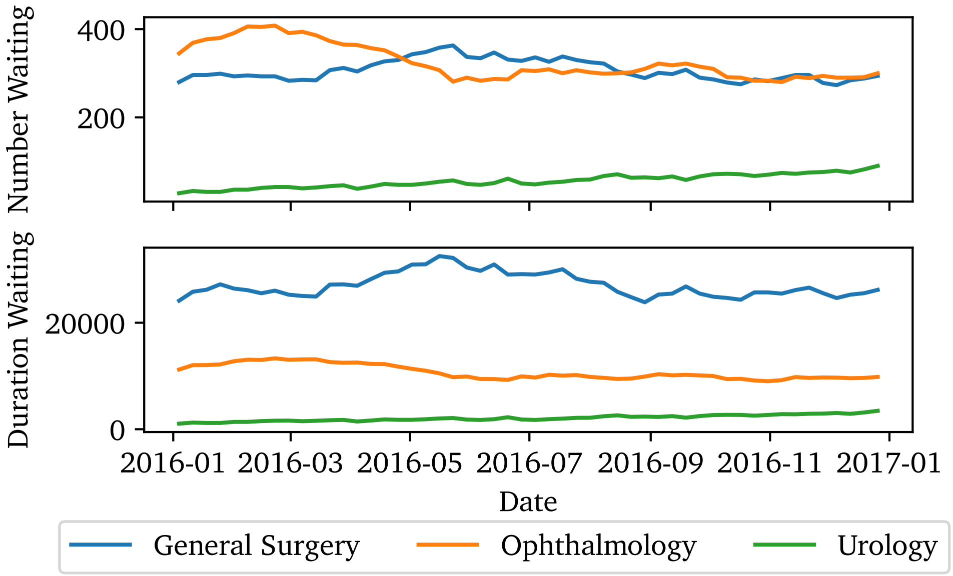


Figure 6.4: Number and Total Duration (mins) of Operations Waiting Throughout 2016

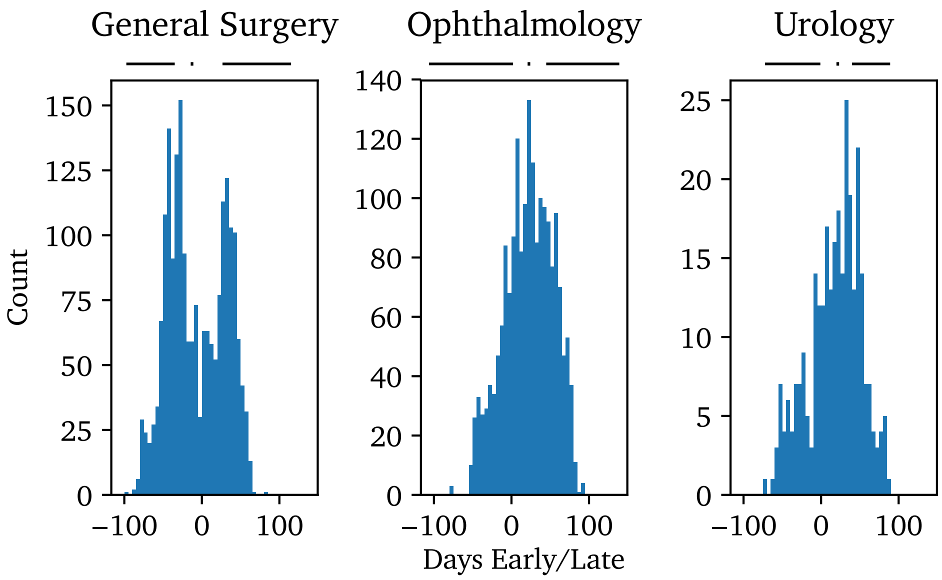


Figure 6.5: Histograms and Box Plots of the Earliness and Lateness of Operations Available for Scheduling in 2016

ending with 90 patients waiting. The total duration of operations waiting displays the same trends, except that there is a large difference between General Surgery and Ophthalmology. The larger duration waiting for General Surgery than Ophthalmology, even though the number waiting is the same, is due to General Surgery operations being longer on average than Ophthalmology operations.

The distribution of the lateness of the operations performed in 2016 by each of the specialties is displayed in figure 6.5 using both histograms and horizontal box plots above them. Ophthalmology and Urology have similar, unimodal distributions of lateness with medians just above zero, whereas General Surgery has a bimodal distribution with its median below zero. For all of the specialties no operation is more than 100 days early or late.

We also analysed the proportion of sessions that ran overtime in 2016, for each of the specialties. We considered both the proportion that actually ran overtime, calculated from the actual historical operation durations, and the proportion that ran overtime calculated from a simulation using 10,000 sampled durations of the operations. Algorithm 6.1 is used to create this duration sample and the operations performed before 2016 are used as the set of operations from which the durations are sampled. The actual and simulated overtime proportions are both given in table 6.2. For all of the specialties there is a significant difference in

Table 6.2: Proportion of Sessions that Run Overtime in 2016

Specialty	Actual Proportion	Simulated Proportion
General Surgery	0.20	0.45
Ophthalmology	0.08	0.24
Urology	0.03	0.51

the actual proportion that ran overtime and the simulated proportion, with the simulated proportion being between 0.16 and 0.48 greater than the actual proportion. This is possibly because in practice the sessions are overbooked and then if the first few operations are running early the remaining operations are allowed to be performed. On the other hand if the first few operations run late, then the last operation(s) can be cancelled. This results in sessions that on average would frequently run overtime, but do not actually run overtime because the last operations were only performed because the first ones were quicker than usual.

In the scheduling model defined in the previous section it is assumed that all operations that are assigned to a session are always performed, and no operations are performed that were not previously assigned. This means that we cannot take advantage of the first operations in a session running early in the same way that the actual OR schedulers can. Consequently, when comparing the proportion of sessions that run overtime from the optimisation model to the actual sessions, using the simulated proportion is a fairer comparison.

Finally, we investigated the distributions of the simulated probability of session overtime for each of the specialties. This is shown in figure 6.6 which contains histograms and horizontal box plots of the simulated probability that sessions ran overtime.

Both General Surgery and Urology have median session overtime probabilities of around 0.5, whereas Ophthalmology has a lower median at 0.2. However even Ophthalmology has a significant fraction of sessions that have a probability of overtime of more than 0.5.

6.3 Results

We first describe the experiments that were performed before presenting the results, for discussions and explanations of the results see section 6.4.

We scheduled the operations throughout 2016 using the optimisation model described in section 5.3, and the corresponding algorithm 5.2, and compared

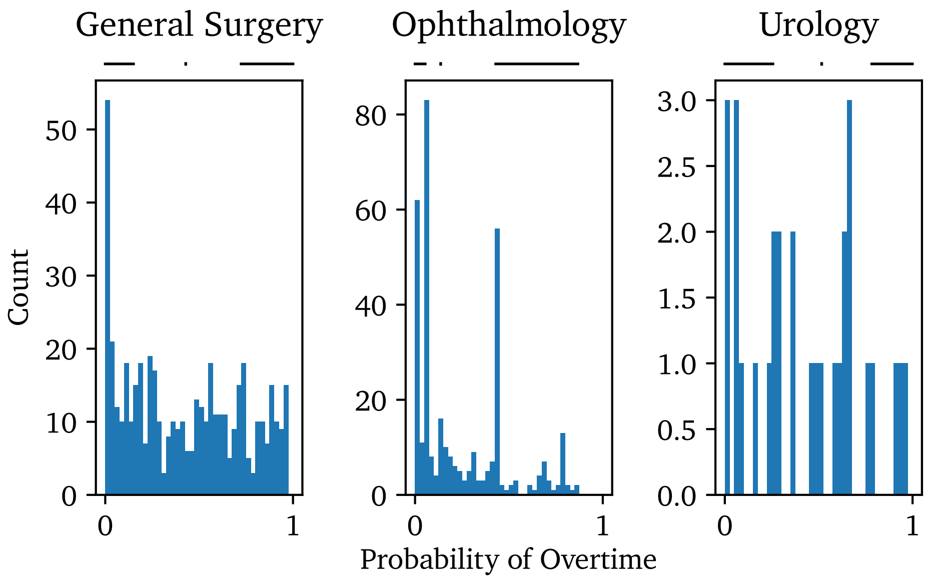


Figure 6.6: Histograms and Box Plots of the Simulated Probability of Overtime of the Historical Sessions in 2016

the solution(s) to the actual schedule to evaluate the model's performance. The operations and sessions in 2016 were chosen to be scheduled because the data available was from January 2015 to July 2017. Scheduling throughout 2016 allowed for: one year of data to be used to train the overtime probability estimation methods, one year of operations to schedule, and have six months of additional sessions left over to schedule operations in at the end of 2016.

To define the rescheduling problem initial sets of operations, O_1, and sessions, S_1, are required, as well as the set of time points, I, at which the model is to be solved. For each specialty, the set of initial operations, O_1, is defined as the all of the operations that have arrived needing an operation before 04/01/2016 but have not yet had their operation performed. The sessions in which these operations are scheduled, S_1, are all of the historical sessions from 04/01/2016 until 01/07/2017. The fourth of January is used as the starting date as it is the first Monday of 2016 and there were no sessions on the previous days in 2016.

The set, I, of time points, i, at which operations are scheduled, is defined for $i = 1 \text{ to } 52$, and $I = \{1, 2, \ldots, 52\}$, representing the weeks in the year 2016. The date-times, τ_i, associated with the time points in I are 12 a.m. in the morning on the Monday of the corresponding week in the year, so $\tau_1 = 04/01/2016\ 00{:}00{:}00$, $\tau_2 = 11/01/2016\ 00{:}00{:}00$, and $\tau_{52} = 26/12/2016\ 00{:}00{:}00$. This means that the rescheduling problem is solved once a week throughout 2016. Any patients that arrive between time points have to wait until the next Monday to be scheduled. In general this may result in some patients not being able to be scheduled before their due date, if their due date was less than a week after when they arrived. All of the operations performed at the MSC, however, are non-urgent elective operations and the smallest time between arrival and due date is two weeks. Therefore patients will always have at least one opportunity to have their operation scheduled before their due date, however they will not necessarily always be assigned to a session before their due date, depending on how many other operations need to be scheduled and their respective due dates.

The rescheduling problem described above was applied to three of the specialties that perform operations at the MSC: General Surgery, Ophthalmology, and Urology. The values of the parameter υ, which determines the maximum allowed probability of overtime for any session, were 0.5, 0.4, 0.3, 0.2, and 0.1. Each of these values of υ was combined with three values for the parameter λ, which is the exponent applied to the lateness of an operation session assignment in the cost function, 1, 1.5, and 2. The value of ρ, which is the penalty applied if an operation is moved to another session after it has already been scheduled in a session was set as equal to being one week late with an exponent of 1, regardless of the value of υ or λ. This means, if the earliness and lateness in the cost coefficients is being measured in days, that $\rho = 7$.

The five values of υ and three values of λ result in 15 combinations of parameter values to be used. In addition each of the four methods for estimating the probability that a session runs overtime, GND, SND, DED, and DVED, were all applied to each of the 15 (υ, λ) combinations, giving 60 parameter combinations. Finally both the original objective and the conditional value at risk (CVaR) objective were used, giving a total of 120 problems for each of the three specialties. For the CVaR objective λ was held constant at 1, and instead the three values 0.8, 0.9, and 0.95 were used for β, where β is the quantile of lateness the operations with lateness above which are averaged in the objective function.

For each of the 120 problems 52 mixed integer programmes (MIPs) had to be solved, one for each of the 52 weeks in 2016, in total among the three specialties 18,720 MIPs were solved. These MIPs were solved using Gurobi version 8.1.1 on a server running Ubuntu version 16.04.4 with 24 cores each at 2.60 GHz, and 48 GB of RAM. For each MIP a relative gap of 0.05 (5%) and a time limit of 5 minutes were used as the terminating criteria. This means that Gurobi would stop searching for new solutions and report the current best solution if the difference between the upper and lower bound divided by the absolute value of the upper bound was less than 0.05, or the time limit of 5 minutes was reached. To ensure that a solution was always returned Gurobi was given a feasible solution at the beginning of the solution process, this feasible solution was found using a first fit heuristic.

The first fit heuristic begins by sorting all of the operations by their due date, with those due first coming first in the list. The heuristic then proceeds through the sorted list of operations assigning each one to the first session (chronologically) in which it fits, given the operations that have already been assigned. Whether or not an operation can fit into a session is determined by using the session's duration, the assigned session duration (and variance) with the operation included, the limit on the probability of overtime, and one of the methods for estimating this probability. For the rescheduling problems only the new operations are assigned to sessions by the first fit heuristic. All operations that remain from the previous period are assigned to the same sessions that they were in previously.

The time limit and relative gap were enforced because 18,720 MIPs needed to be solved. If every problem hit the 5 minute time limit then this would take a total of 65 days. Many of the problems did not reach the time limit and therefore the total computation time was much shorter than this. In a practical setting only one of these problems would need to be solved each week for each specialty, and therefore more time could be committed to solving each problem. In addition, in practice, it would not be necessary to solve problems with many different combinations of parameters. In particular, if the OR management has an acceptable probability of overtime for all sessions, then it can be used for υ. Further only one of the four methods for estimating the probability of overtime would need to be used. It is also possible that the OR management would have a clear idea of their objectives when scheduling operations and therefore the appropriate objective function and value of either λ or β could be determined. The OR management could also have more than one objective, in which case a multi-objective problem could be defined. However, for the purposes of this case study we assume that each problem we solve has a single objective. We investigate the effects of different values of the parameters on the schedule with the aim of understanding which values would be appropriate in a practical application. Before proceeding with the performance of the solutions during 2016 it is useful to first consider the performance of the four methods for estimating the probability that a session runs overtime, presented in section 5.2.2.

6.3.1 Constraint Methods as Classifiers

To analyse the performance of the four methods for estimating the probability that a session runs overtime each of the four methods was used to estimate the probability of overtime in every session in every solution. For example, consider General Surgery. In each of the 120 solutions there are between 500 and 700 sessions to which operations are allocated, depending on the number of operations in each session. For each one of these sessions the assigned session duration and variance can be calculated. Using the assigned session duration and variance the probability of overtime can be estimated by each of the four methods. The probability that each of these sessions runs overtime is then also estimated using 10,000 sampled durations of the operations. Two samples of 10,000 were created, one where the durations are sampled from the same data that methods were trained on, that is operations performed before 01/01/2016, and a second where the durations are sampled from operations that were performed after 01/01/2016. These two samples of the operations' durations are referred to as the 'historical' and 'future' samples (both here and in the following sections) and are equivalent to an in-sample and out-of-sample test. We therefore have for each session four estimates of the probability of overtime from the four methods described in section 5.2.2, and two estimates using simulations based on the historical and future operations.

To assess the performance of each of the four methods they were used as a binary classifier at various levels of overtime probability. This means that for a given value of υ a session is assigned a 'positive' if the estimate for the probability of overtime is greater than υ and a 'negative' if it is less than or equal to it. This classification into positives and negatives is also performed for the estimates from simulation. The accuracy of the estimates from the four methods compared to those from the simulations is calculated by counting the number of sessions that were given the correct label – the estimation method assigns a positive and the simulation does too, or the estimation method assigns a negative and the simulation does too – divided by the total number of sessions. Treating the methods as binary classifiers in this way is similar to how they are used in the optimisation models, where if the method's estimate of the probability that a session runs overtime is greater than υ, then that session is infeasible and not allowed in the solution. Figures 6.7 and 6.8 give the accuracies of the four methods compared to the historical and future simulated probabilities, for υ varying between 0 and 1, and table 6.3 summarises the mean accuracies of the four methods over all values of υ.

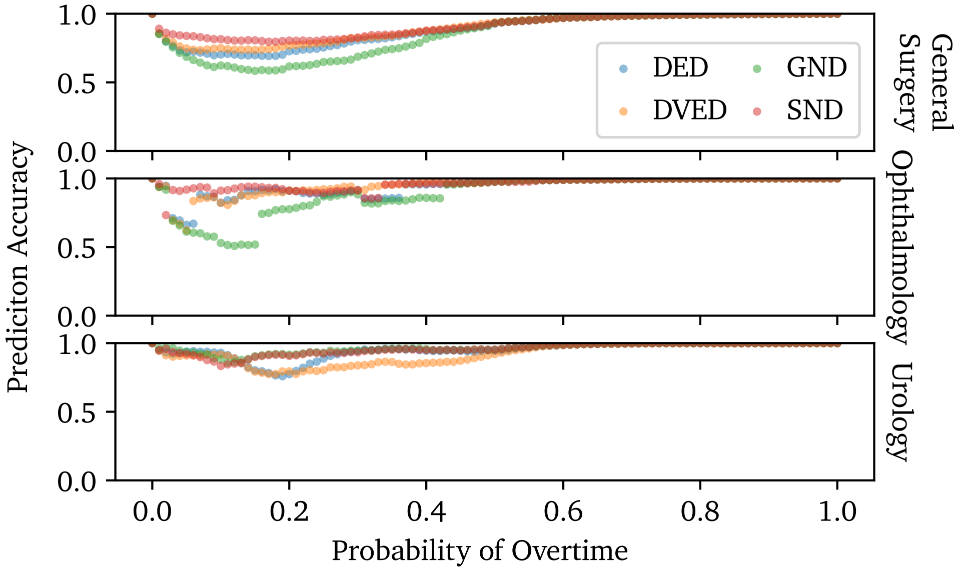


Figure 6.7: Historical Accuracy of Overtime Probability Estimation

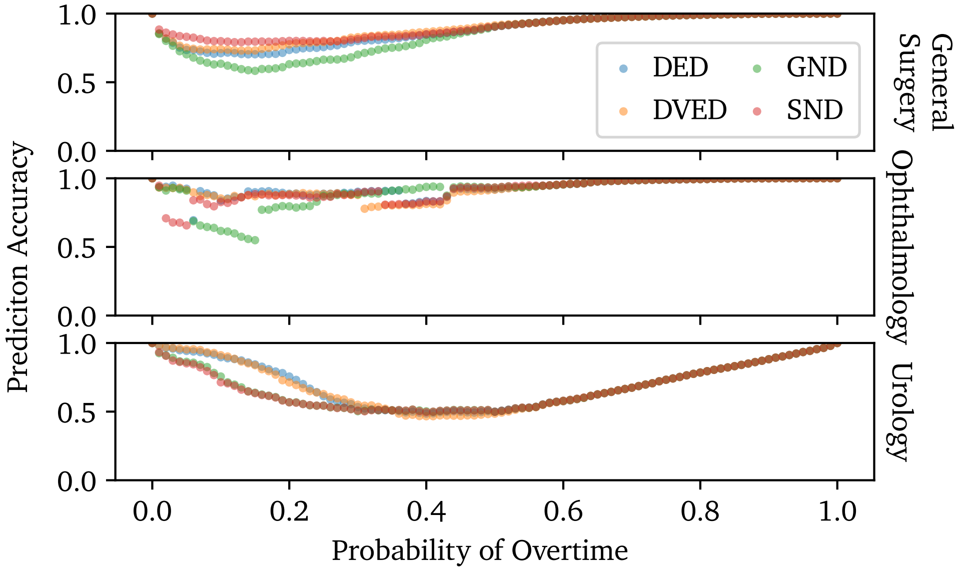


Figure 6.8: Future Accuracy of Overtime Probability Estimation

Table 6.3: Mean Accuracy of Overtime Estimation Methods

Specialty	Method	Historical Accuracy	Future Accuracy
General Surgery	DED	0.89	0.88
	DVED	0.90	0.89
	GND	0.85	0.85
	SND	0.91	0.90
Ophthalmology	DED	0.94	0.93
	DVED	0.95	0.93
	GND	0.89	0.91
	SND	0.96	0.92
Urology	DED	0.95	0.71
	DVED	0.92	0.71
	GND	0.96	0.68
	SND	0.96	0.67

For the historical sample (in sample) the accuracy for all three specialties follows a similar pattern, where it decreases from 1.0 to about 0.8 as the probability of overtime threshold increases from 0.0 to 0.2, and then increases to 1.0 again as the probability of overtime threshold increases from 0.2 to 0.6. General Surgery displays a clear difference in the accuracy of the four estimation methods with SND being the most accurate, then DVED, then DED, and finally GND. For both Ophthalmology and Urology there is less difference in accuracy between the four methods, however, the GND method is slightly less accurate for Ophthalmology than the other methods.

For General Surgery and Ophthalmology the accuracy of the four methods is similar to the historical sample on the future sample (out of sample). For Urology, however, the accuracy on the future sample is much worse than the historical sample for all four methods. This stark difference between the historical and future Urology samples is a recurring observation, first noted in 6.2 and discussed further in 6.4.1.

It was also of interest to investigate whether the four estimation methods were over or under predicting the probability of overtime on both the historical and future sample. To achieve this 20 bins of overtime probability were created, evenly spaced between 0 and 1. Each session was then placed in one of the bins based on its simulated probability of overtime. The mean simulated probability of overtime for the sessions in each bin was then plotted against the mean of each of the estimates (from the methods) for the sessions in the bin, as shown in figures 6.9 and 6.10.

It can be seen that, excluding the future sample for Urology, the methods over

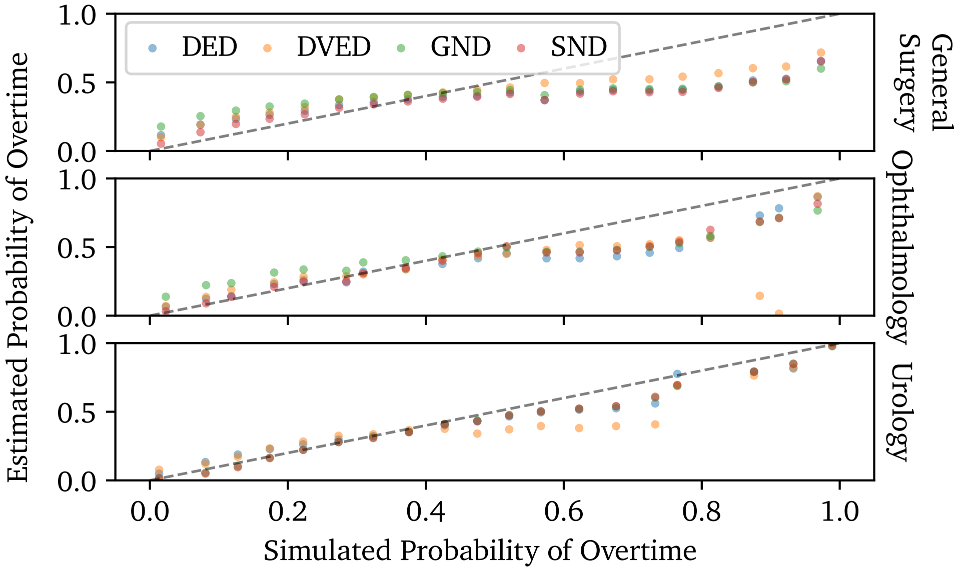


Figure 6.9: Mean Error of Overtime Probability Estimates on Historical Sample

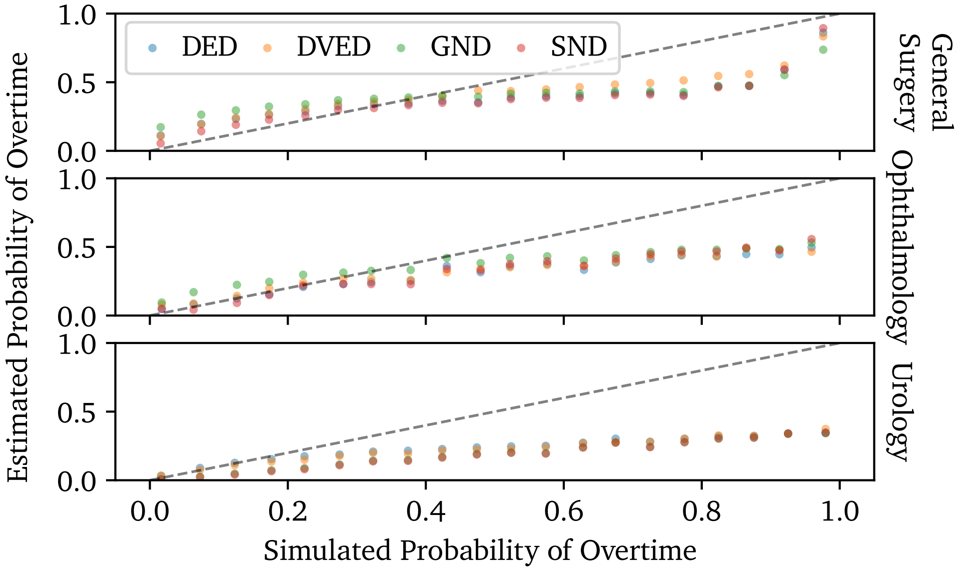


Figure 6.10: Mean Error of Overtime Probability Estimates on Future Sample

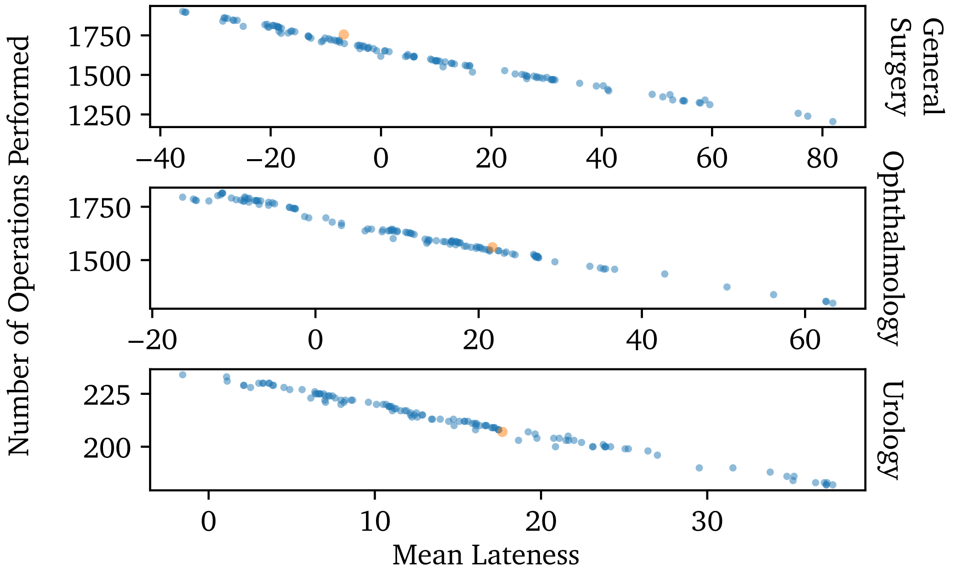


Figure 6.11: Mean Lateness Compared to Number of Operations Performed

estimate the probability of overtime on average for low probability sessions, and under estimate it for high probability sessions. For the future sample for Urology all four methods consistently underestimate the probability of overtime.

6.3.2 Scheduling Operations in 2016

The complete results of solving the 120 problems, arising from the combinations of objective function, probability of overtime estimation method, and model parameters, for each specialty are given in appendix B. Here we first present overall trends in the output metrics of interest, before examining some of them in more detail.

The most obvious trend in the results of solving the rescheduling problems is the relationship between the mean lateness of the operations scheduled, and the number of operations performed in 2016. Figure 6.11 shows this relationship for each specialty, the solutions we found are in blue, and the actual schedule is in orange.

There is a clear negative linear relationship between mean lateness and number of operations performed, that is as the mean lateness increases the number of operations performed decreases.

A similar, although not quite as clear cut, positive linear relationship is displayed between the mean lateness and the total duration (in minutes) of operations

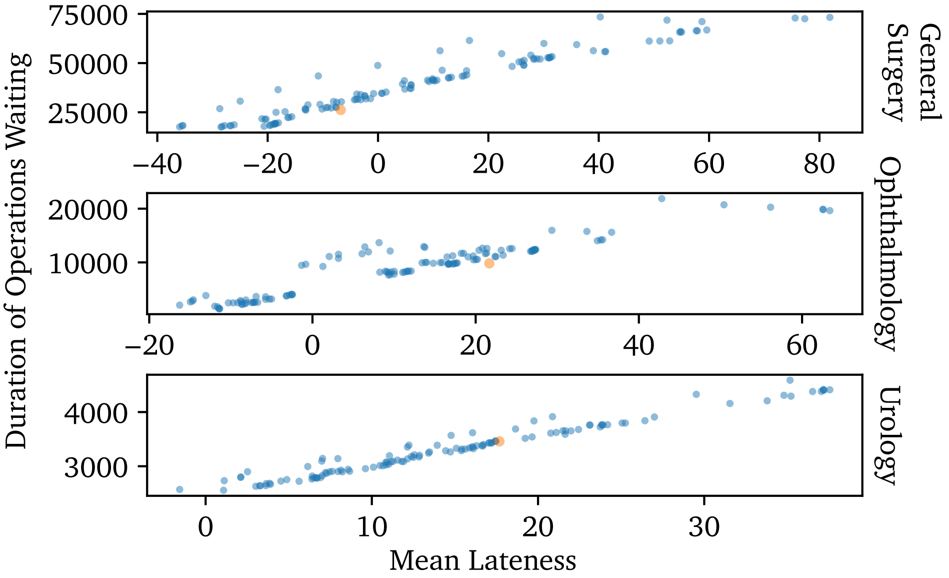


Figure 6.12: Mean Lateness Compared to Total Duration of Operations Waiting

waiting to be performed at the end of 2016, this is shown in figure 6.12.

When mean lateness is compared to CVaR at 0.95 lateness, or in other words the mean lateness of the latest 5% of operations, there is no clear relationship for either Ophthalmology or Urology, however for General Surgery there is a slight positive relationship, as is shown in figure 6.13. For Urology there is a distinct partition of the solutions in terms of CVaR at 0.95 lateness, with a set of solutions around 300, and another around 0, and no solutions in between. A similar partition exists to a lesser extent for Ophthalmology.

Both the mean probability of overtime, and the CVaR at 0.95 of the probability of overtime display negative relationships with the mean lateness, as shown in figures 6.14 and 6.15. The relationship between mean lateness and mean overtime probability is linear for Urology, but for both General Surgery and Ophthalmology the mean probability of overtime flattens out as it approaches 0, this may also happen for Urology but none of the solutions found have a low enough mean overtime probability to tell. The relationship between mean lateness and the CVaR at 0.95 of the probability of overtime is also approximately linear for all of the specialties.

Also of interest is how the distribution of the lateness of the operations, and overtime probability of the sessions, is influenced by the parameters υ, the limit on the probability of overtime, and λ and β, the exponent on lateness and the

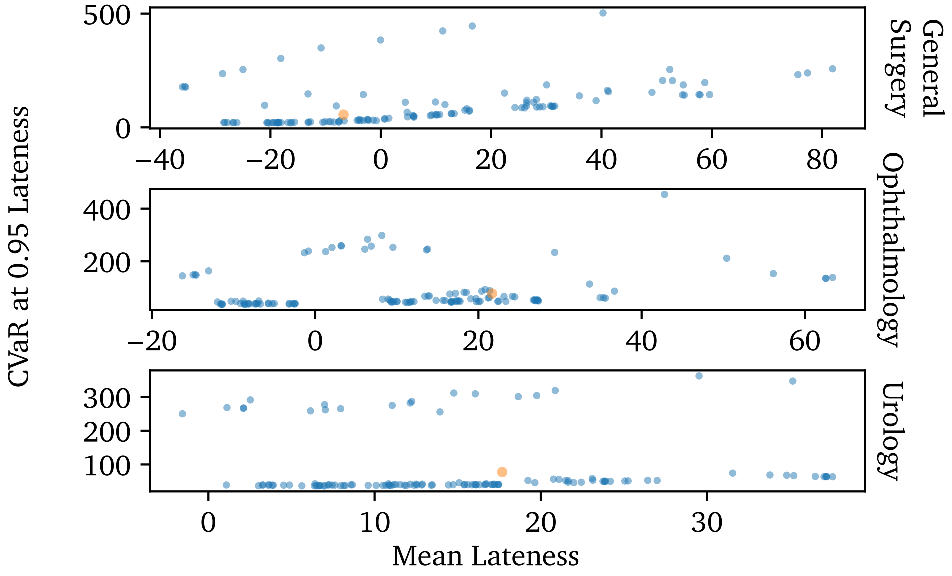


Figure 6.13: Mean Lateness Compared to CVaR at 0.95 Lateness

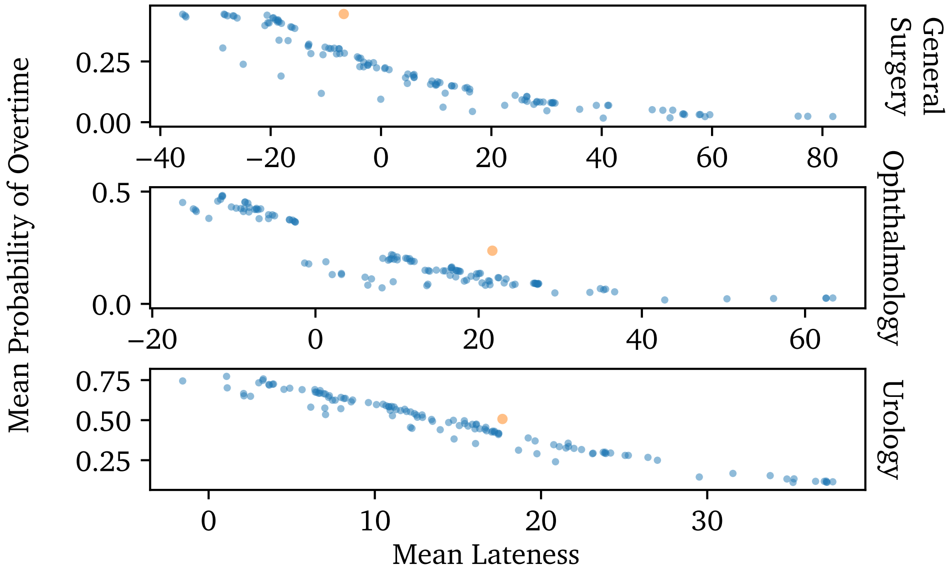


Figure 6.14: Mean Lateness Compared to Mean Probability of Overtime

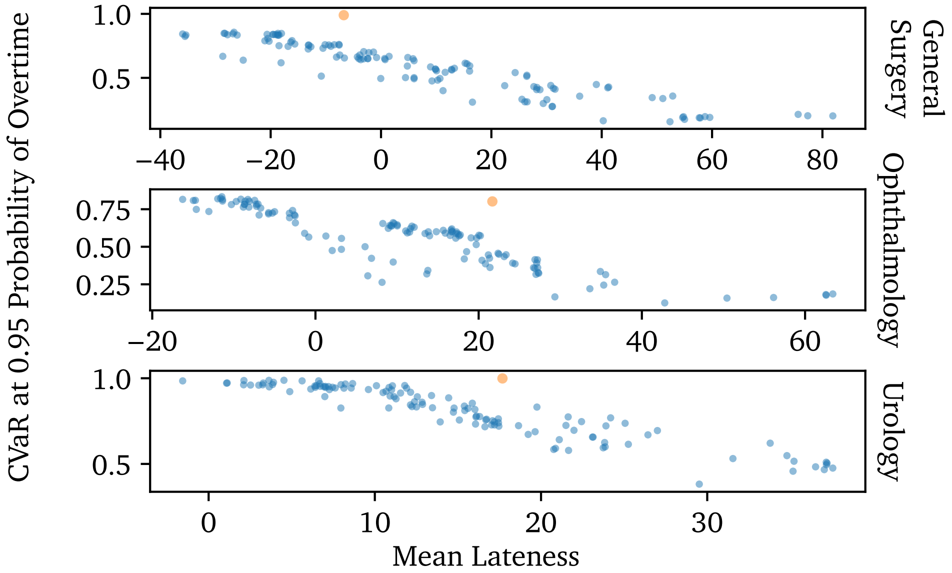


Figure 6.15: Mean Lateness Compared to CVaR at 0.95 Probability of Overtime

CVaR quantile level respectively, as well as by the two types of objective functions. In the interests of brevity only the box plots for General Surgery are shown here, although those for Ophthalmology and Urology can be found in appendix C. To investigate these relationships figures 6.16 and 6.17 show box plots of the distribution of the lateness of the operations for each combination of session constraint method, lateness exponential (or CVaR quantile level), and overtime probability limit for both the Total objective and CVaR objective respectively. The operations that were scheduled in 2016 are shown in the blue box plot and those left for 2017 in orange.

For the Total objective there is a significant difference in the distributions for all of the methods and overtime probability limits when the late exponent is increased from 1.0 to 1.5, but not much difference from 1.5 to 2. When the late exponent increases from 1.0 to 1.5 the lateness of the latest operations is significantly decreased.

For all combinations of parameters the lower quartile, median, and upper quartile of lateness are very close together for operations scheduled in 2016. For operations scheduled in 2017, however there is a larger interquartile range in the solutions found using the DED and GND methods when the late exponent is 1.0.

For most parameter combinations the distribution of lateness is worse for the operations scheduled for 2017 than 2016, and in particular the maximum late-

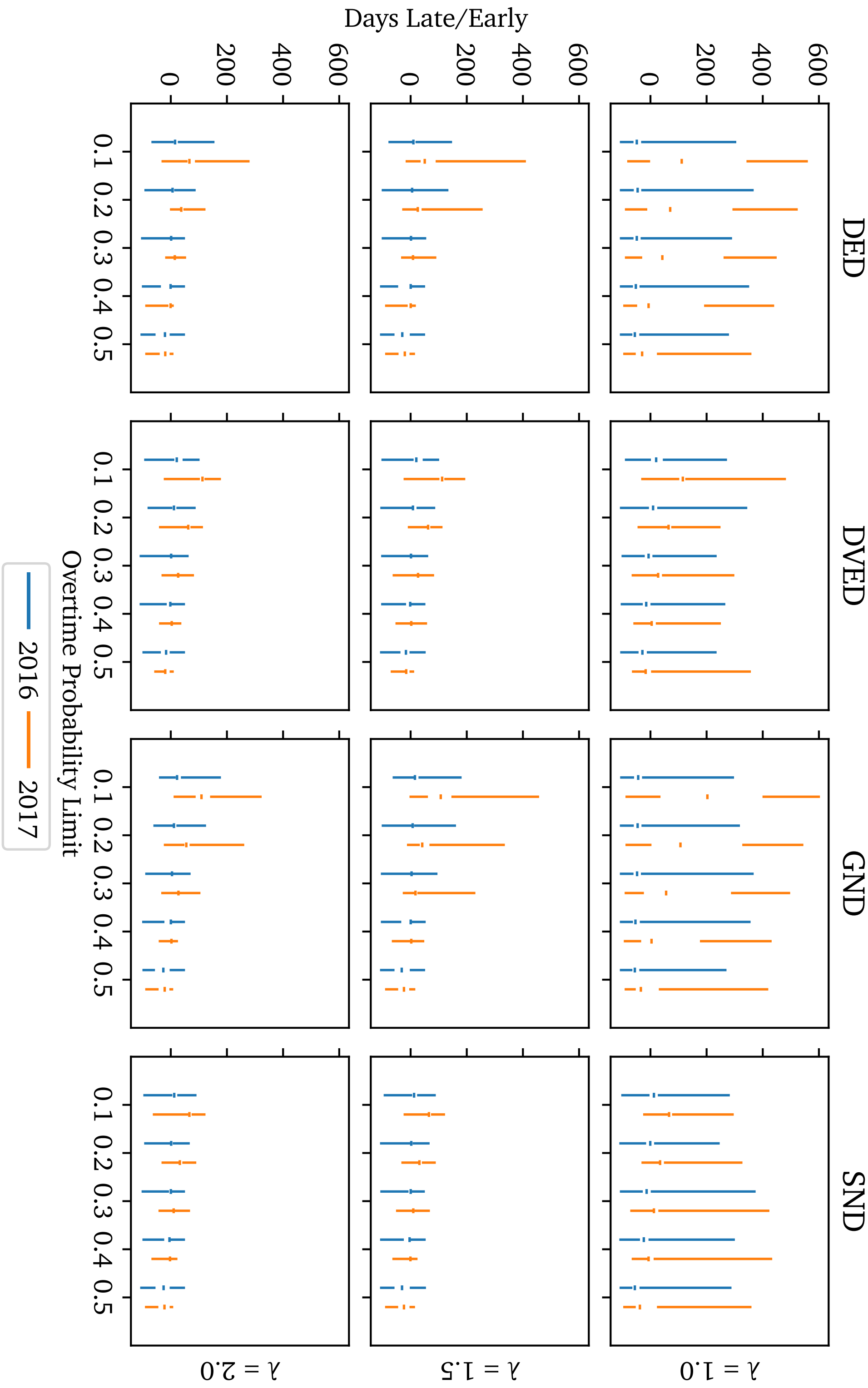


Figure 6.16: Box Plots of Lateness for General Surgery with the Total Objective

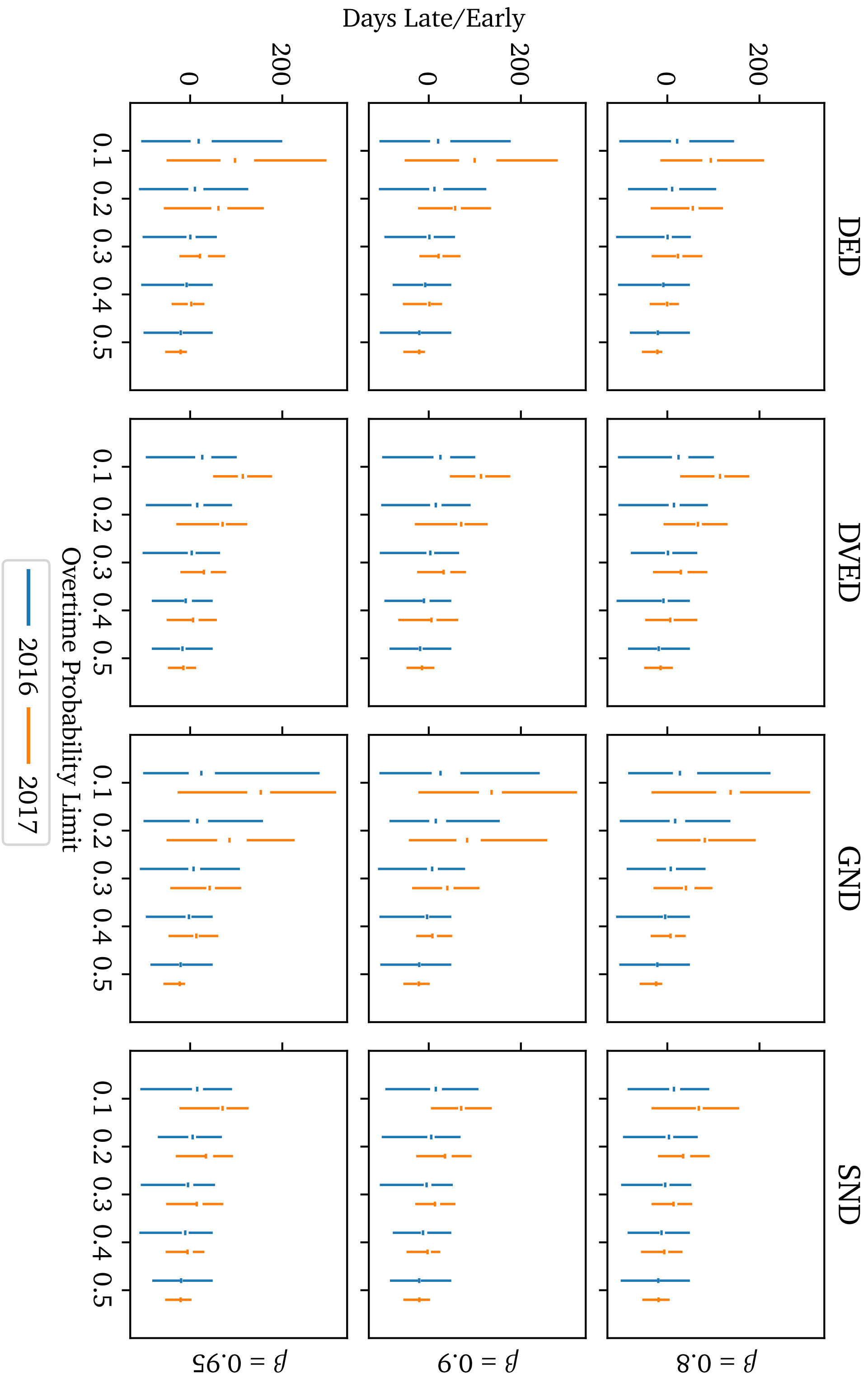


Figure 6.17: Box Plots of Lateness for General Surgery with the CVaR Objective

ness. The exceptions to this are those combinations that have a high limit for the probability of overtime, either 0.4 or 0.5. The relationship between the parameters and the way in which the lateness of the operations changes over time is further investigated later in this section. There is also a relationship, similar to that shown in figure 6.14, where the median lateness decreases as the overtime probability limit increases, which is more obvious in the distributions of the operations scheduled for 2017.

For the CVaR objective there is a very little difference in the distributions for all of the methods and overtime probability limits when the CVaR quantile is increased from 0.8 to 0.9 or to 0.95.

The trend of operations being scheduled later during 2017 than 2016 is once again seen for overtime probability limits of 0.1, 0.2, and 0.3, and sometimes 0.4. Also the trend of a decreasing median lateness, and range of lateness, as the overtime probability limit increases is more pronounced, possibly because the range of the y-axis is smaller.

The main difference between the two objective functions is that with the CVaR objective there are no combinations of parameters that result in a maximum lateness over 250, whereas all of the combinations with the Total objective and a late exponent of 1.0, and some with a late exponent of 1.5, have operations with lateness over 250.

The same analysis of parameter combinations was performed for the overtime probability of the sessions, rather than the lateness of the operations, and the resulting box plots are shown in figures 6.18 and 6.19. Instead of the blue box plots representing operations scheduled on 2016, and the orange in 2017, the blue box plots are the probability of overtime evaluated on the historical surgery data, and the orange box plots the probability of overtime using future surgery data. The historical surgery data being the operations and sessions from before the beginning of 2016, and the future surgery data those coming afterwards.

For the Total objective there is very little difference between the distributions of the probability of overtime of the sessions when the late exponent is changed from 1.0 to either 1.5 or 2.0. The overtime limit, however, does have a significant influence on the distribution of overtime probability. In particular as the overtime limit increases the lower quartile, median, upper quartile and maximum probability of overtime all increase regardless of late exponent or session constraint method.

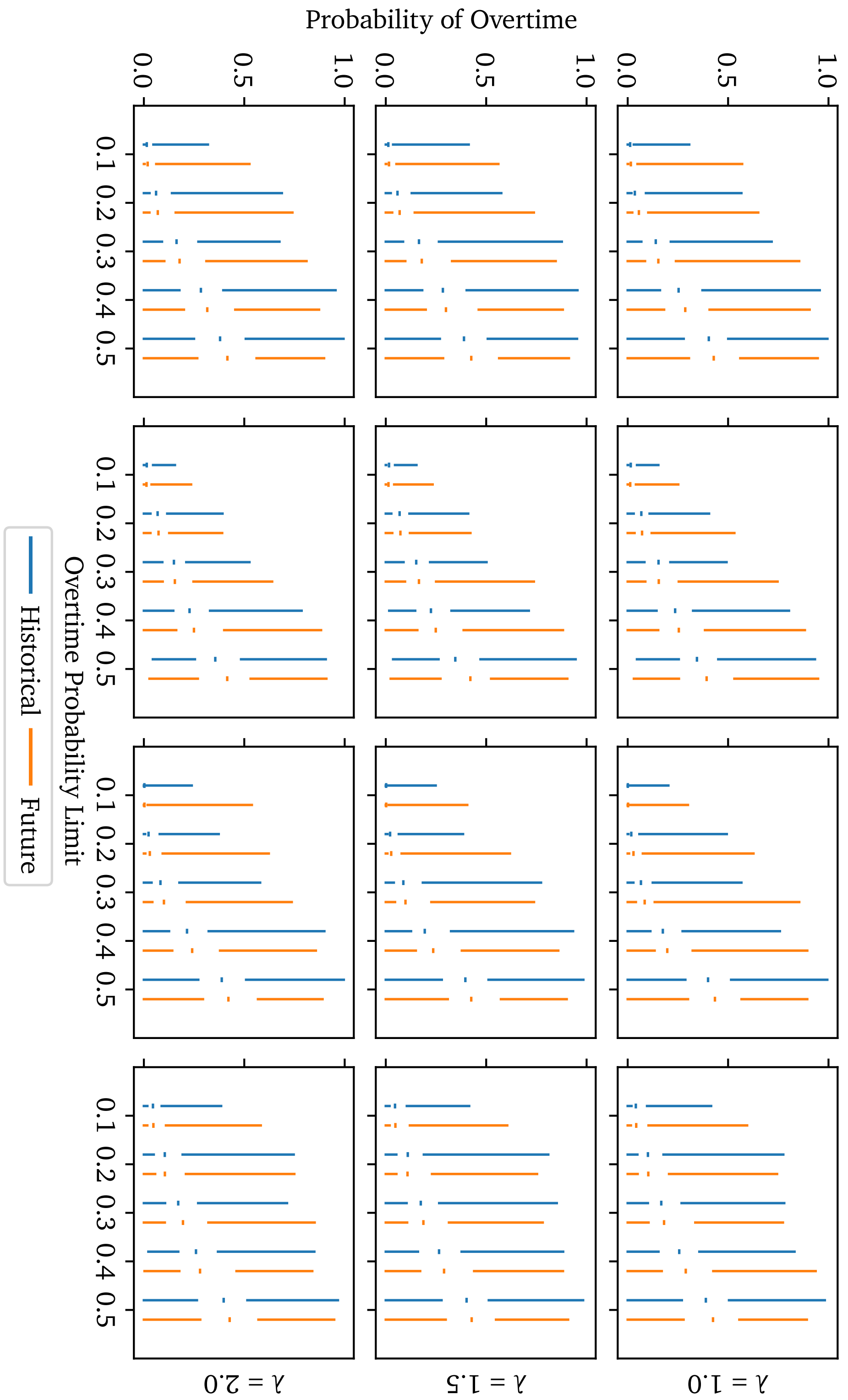


Figure 6.18: Box Plots of Overtime Probability for General Surgery with the Total Objective

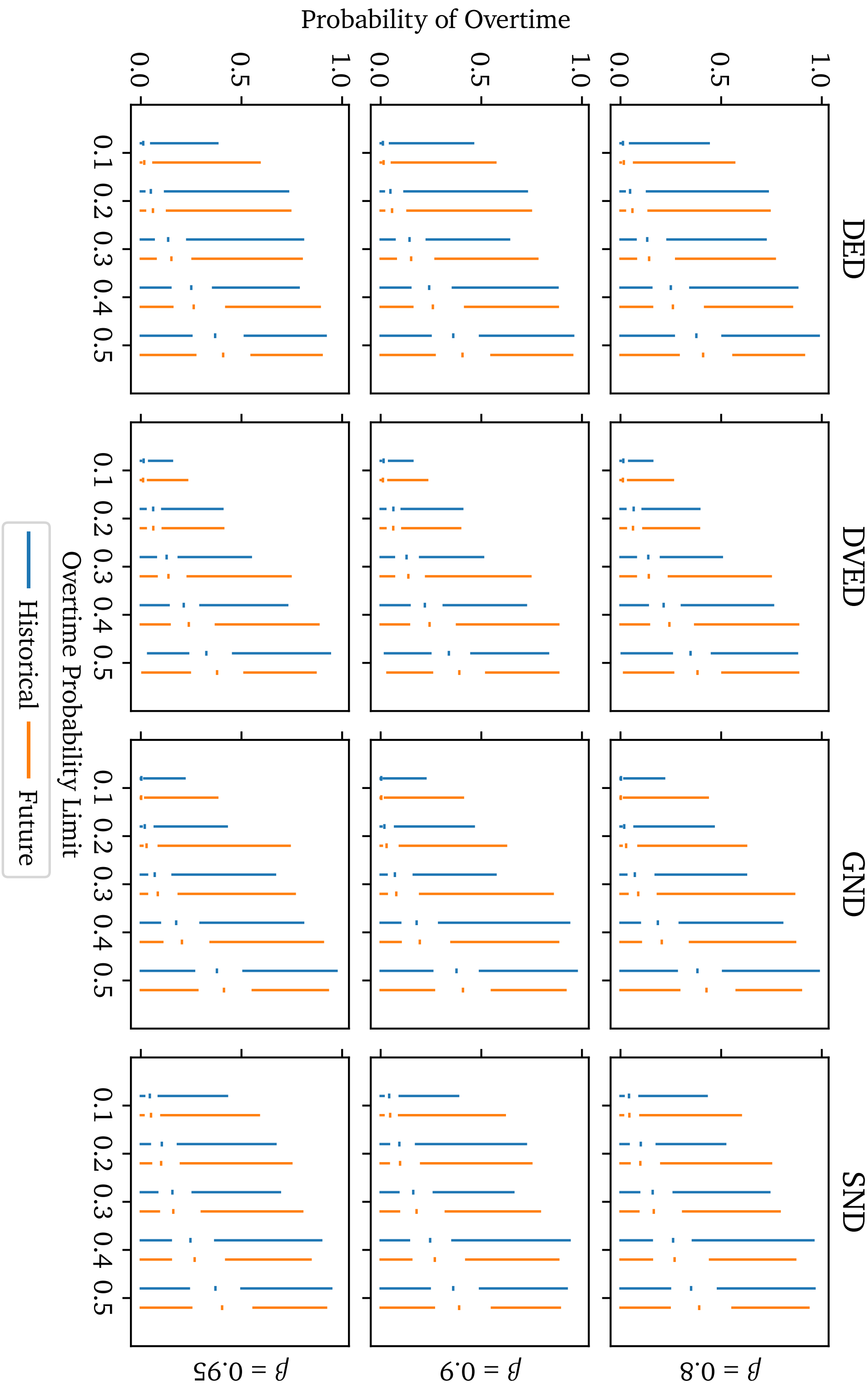


Figure 6.19: Box Plots of Overtime Probability for General Surgery with the CVaR Objective

Additionally the distributions of overtime probability for the historical surgery sample, and the future surgery sample are very similar with no clear trend for both General Surgery and Ophthalmology. For Urology, which is not shown here but is figure C.7 in appendix C, all five metrics (minimum, lower quartile, median, upper quartile and maximum) are larger for the future sample than the historical.

There is not much difference between the four methods for estimating the probability of overtime of a session (DED, DVED, GND, and SND), for the higher overtime limits of 0.4 and 0.5. For the lower overtime limits, however, DVED and GND have slightly lower values for most of the distribution, and for the maximum in particular. The CVaR objective function solutions have distributions very similar to the Total objective function, and the same trends are also apparent.

The final analysis on the 2016 schedules that we performed was to investigate how the lateness of the operations scheduled changed throughout the time period, and what effects the parameters had on this relationship. To do this we smoothed scatter plots, for each parameter combination, of the arrival date of an operation on the x-axis against the lateness of the operation on the y-axis, using a LOESS smoother. The resulting smoothed lines for General Surgery are shown in figure 6.20 for the Total objective and figure 6.21 for the CVaR objective.

For General Surgery and the Total objective function for all four of the probability of overtime estimation methods, the long term trend in lateness against arrival date is almost identical when λ is equal to 1.5 or 2.0. For the DVED and SND methods the trend is also the same when $\lambda = 1.0$, however for the DED and GND methods the trend is different when $\lambda = 1.0$. The difference between the trends when λ is 1.0 and 1.5 (or 2.0) is that the smoothed lateness starts higher and decreases more quickly when $\lambda = 1.0$, whereas when λ is 1.5 or 2.0 the smoothed lateness is lower at the beginning.

Apart from this the four methods overall display similar trends where lateness increases over time for smaller values of the overtime limit, υ, and remains constant or reduces slightly over time for the middle and upper values.

For the CVaR objective there is even less difference between the trends with different values of β than there is between the values of λ for the Total objective. There is also little difference in the trends between the four methods of estimating the probability of overtime. The main factor that influences how the

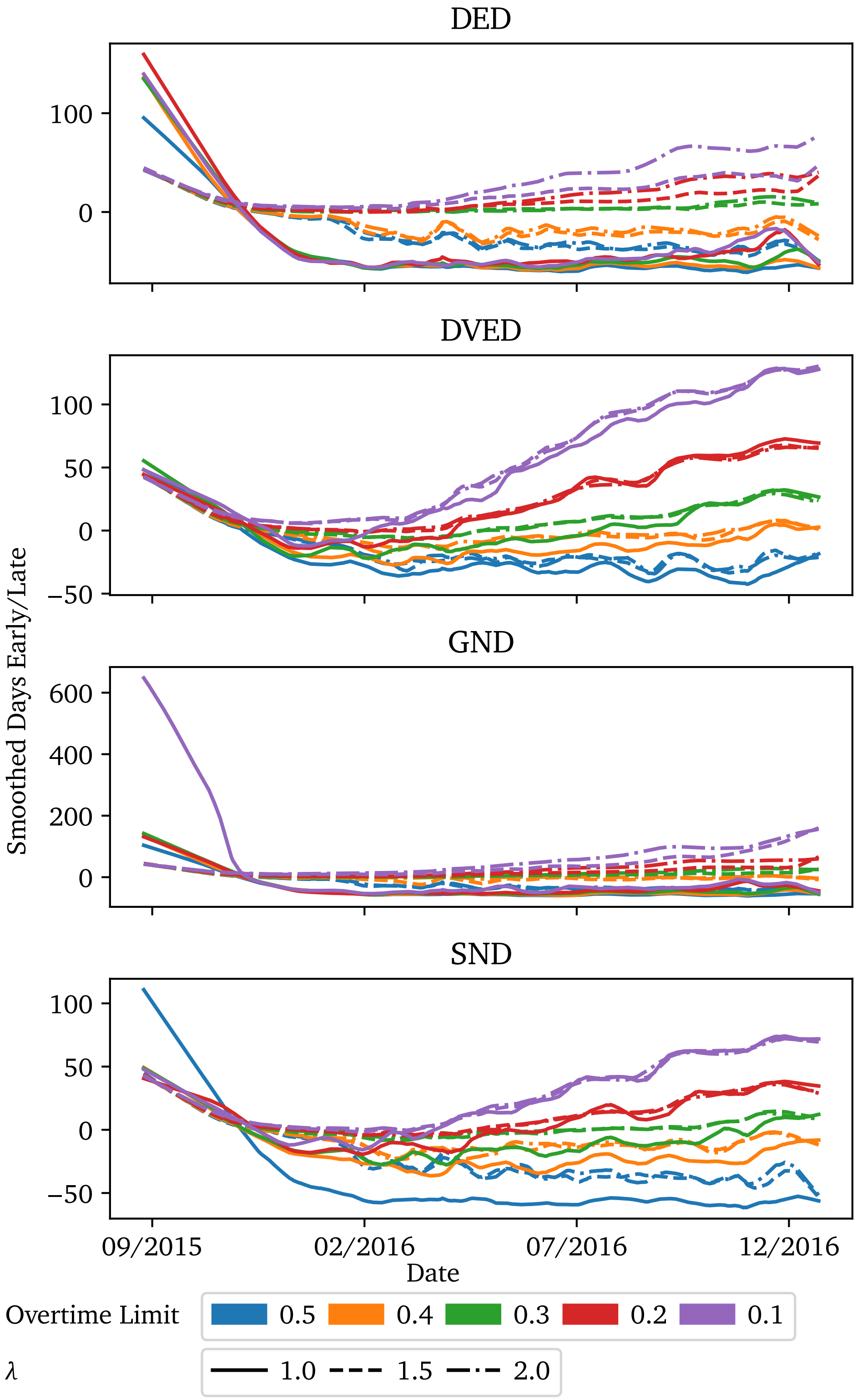


Figure 6.20: Smoothed Scatter Plots of Arrival Date and Lateness for the Total Objective

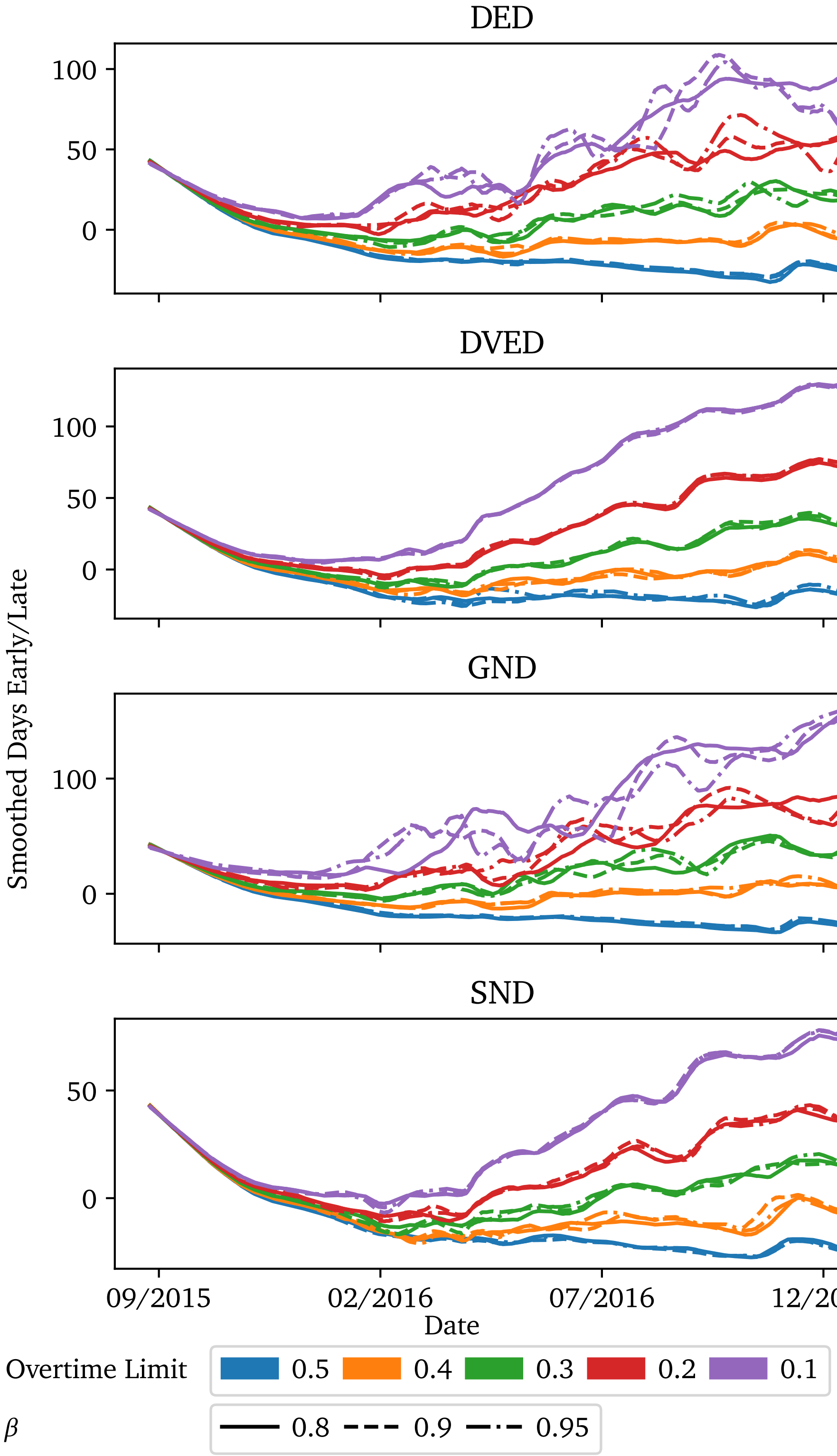


Figure 6.21: Smoothed Scatter Plots of Arrival Date and Lateness for the CVaR Objective

smoothed lateness changes over time is the overtime limit, υ, with larger values displaying a decreasing trend, and lower values an increasing one.

These relationships between the parameters and estimation methods, and the smoothed lateness over time for General Surgery are broadly similar to those seen for Ophthalmology and Urology, the smoothed scatter plots of which are given in appendix D, although the actual trends of smoothed lateness over time are different, for example for Urology even small values of υ show a decreasing lateness over time in figure D.3. Overall these smoothed scatter plots show that some combinations of parameters lead to decreasing lateness over time, while others lead to increasing lateness.

6.3.3 Optimality Gaps

To assess how close the solutions to the MIPs solved by Gurobi are to optimal, and therefore how difficult the problems are to solve, the average absolute, relative and uniform gaps are given for each combination of parameter values and for each specialty. As stated earlier there are a total of 120 combinations of λ (or β), υ, overtime probability estimation method, and objective function for each specialty. For each one of these combinations 52 MIPs were solved, one for each week in 2016, and a relative gap of 0.05 (5%) and a time limit of 5 minutes were used. At the end of the solution process we have a value for the best solution found so far, z^* and a lower bound on the objective function value, z^-.

The absolute gap, ϵ, is defined as the difference between the best solution's objective value and the lower bound $\epsilon = z^* - z^-$. The relative gap, η, is the absolute gap divided by the absolute value of the best solution's objective function value, $\eta = \frac{z^* - z^-}{|z^*|}$. The uniform gap, ψ, is the absolute gap divided by the best solution's objective value plus one, i.e. $\psi = \frac{z^* - z^-}{|z^*|+1}$.

The uniform gap is included in the results because the objective functions can take a wide range of values, and can be both positive and negative. The uniform gap behaves similarly to the relative gap when z^* and z^- are both large, and similarly to the absolute gap when z^* is close to zero. When z^* is close to zero the relative gap can be very large even if the absolute gap is small. This can make the relative gap an unreliable estimate of how good a solution is, particularly when the lower bound is negative but the objective function value of the incumbent is small but positive. As the value of the incumbent solution's objective function gets closer to zero the relative gap increases dramatically, whereas the uniform

Table 6.4: General Surgery Optimality Gap Results for Total Objective

λ	υ	DED			DVED			GND			SND		
		ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ
1	0.5	97	0.04	0.04	1885	2.69	2.66	91	0.04	0.04	120	0.09	0.09
	0.4	267	0.04	0.04	3666	2.48	2.47	349	0.04	0.04	2329	1.92	1.91
	0.3	535	0.04	0.04	6479	1.02	1.02	809	0.04	0.04	3942	1.32	1.32
	0.2	1194	0.04	0.04	12730	0.64	0.64	1697	0.04	0.04	6955	0.73	0.73
	0.1	1747	0.04	0.04	22348	0.49	0.49	4235	0.06	0.06	12710	0.59	0.59
Mean		768	0.04	0.04	9422	1.46	1.46	1436	0.05	0.05	5211	0.93	0.93
1.5	0.5	160	0.04	0.04	1397	0.53	0.53	160	0.04	0.04	192	0.05	0.05
	0.4	126	0.28	0.27	3441	3.33	3.26	199	0.54	0.53	1452	1.12	1.12
	0.3	528	0.19	0.19	11912	1.52	1.51	1283	0.22	0.22	4170	2.24	2.23
	0.2	2450	0.07	0.07	41906	0.57	0.57	5560	0.06	0.06	13459	0.74	0.74
	0.1	8788	0.06	0.06	110056	0.44	0.44	31366	0.08	0.08	43632	0.50	0.50
Mean		2410	0.13	0.13	33742	1.28	1.26	7713	0.19	0.19	12581	0.93	0.93
2.0	0.5	244	0.07	0.07	1414	0.44	0.44	226	0.04	0.04	258	0.05	0.05
	0.4	227	0.22	0.22	9298	5.50	5.35	407	0.53	0.50	2295	1.76	1.76
	0.3	2392	0.63	0.60	61294	0.84	0.84	8266	0.33	0.32	16426	1.60	1.59
	0.2	15560	0.09	0.09	316534	0.62	0.62	43544	0.09	0.09	79358	0.69	0.69
	0.1	76290	0.09	0.09	1132759	0.53	0.53	260455	0.08	0.08	338192	0.57	0.57
Mean		18943	0.22	0.21	304260	1.59	1.56	62580	0.21	0.21	87306	0.94	0.93
Grand Mean		7374	0.13	0.13	115808	1.44	1.43	23910	0.15	0.15	35033	0.93	0.93

gap does not increase as much. This makes the uniform gap a fairer measure of the quality of a solution when z^* is very close to zero.

Tables 6.4 and 6.5 show the mean values of the absolute, relative and uniform gaps (ϵ, η, ψ) over the 52 MIPs, for each parameter and overtime probability estimation method combination, for the Total and CVaR objectives respectively for General Surgery.

Since the absolute gaps are never close to zero for the Total objective the relative and uniform gaps are always very close. The absolute gap increases as υ decreases and also increases as λ increases. An increasing absolute gap as υ decreases and as λ increases is to be expected as the objective function values get larger. In general the DVED method has the largest absolute gaps, then the SND method, then GND, and the DED method has the smallest.

For the DVED and SND methods there are no clear trends in the relative or uniform gaps as υ decreases or as λ increases. For the DED and GND methods, however, both the relative and uniform gaps increase as λ increases. In addition the relative and uniform gaps are much smaller for the DED and GND methods than the DVED and SND methods.

Table 6.5: General Surgery Optimality Gap Results for CVaR Objective

β	υ	DED			DVED			GND			SND		
		ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ
0.8	0.5	4	0.64	0.53	13	13.07	2.99	4	0.49	0.41	7	2.82	1.56
	0.4	10	1.47	1.14	22	0.83	0.79	14	0.94	0.76	16	1.13	1.04
	0.3	20	0.57	0.54	34	0.66	0.65	27	0.49	0.48	24	0.72	0.69
	0.2	35	0.46	0.45	55	0.60	0.59	50	0.45	0.45	35	0.62	0.60
	0.1	60	0.45	0.44	91	0.57	0.56	93	0.46	0.46	55	0.58	0.57
Mean		26	0.72	0.62	43	3.14	1.12	38	0.57	0.51	27	1.17	0.89
0.9	0.5	5	3.80	1.48	15	9.21	2.04	5	2.07	1.16	9	13.13	2.81
	0.4	14	1.28	0.96	26	0.77	0.75	18	0.76	0.70	20	0.89	0.84
	0.3	25	0.58	0.56	41	0.66	0.64	33	0.52	0.51	30	0.69	0.67
	0.2	43	0.49	0.48	66	0.62	0.61	56	0.48	0.47	42	0.63	0.62
	0.1	69	0.47	0.47	105	0.60	0.59	107	0.49	0.48	65	0.60	0.60
Mean		31	1.33	0.79	51	2.37	0.93	44	0.86	0.66	33	3.19	1.11
0.95	0.5	6	3.02	1.40	15	2.35	1.20	5	3.45	1.45	9	7.72	2.56
	0.4	16	0.91	0.80	28	0.70	0.68	21	0.70	0.64	21	0.77	0.74
	0.3	27	0.56	0.55	45	0.65	0.64	37	0.51	0.50	34	0.66	0.64
	0.2	47	0.49	0.49	72	0.62	0.62	61	0.48	0.48	47	0.63	0.62
	0.1	74	0.48	0.47	115	0.61	0.61	113	0.49	0.48	71	0.62	0.61
Mean		34	1.09	0.74	55	0.99	0.75	47	1.12	0.71	36	2.08	1.04
Grand Mean		30	1.05	0.72	50	2.17	0.93	43	0.85	0.63	32	2.15	1.01

For the CVaR objective the absolute gap once again increases as υ decreases and also increases as β increases. Once again the DVED method had the largest absolute gaps, then the GND method, then SND, and the DED method has the smallest. The difference in the absolute gaps between the methods are smaller for the CVaR objective than the Total objective.

When considering the CVaR objective, there is a difference between the relative and uniform gap for some of the parameter combinations, particularly when υ is 0.5, when the absolute gaps are smallest. In addition, both the relative and uniform gap decrease as υ decreases, however the effect is larger for the relative gap, as it is generally larger for the larger values of υ. Also the relative gaps are smaller for the DED and GND methods than the DVED and SND methods. Further, the uniform gaps are also smaller for the DED and GND methods than the DVED and SND methods but have not decreased as much as the relative gaps. Overall the absolute gaps are smaller for the CVaR objective than the Total objective, whereas the relative and uniform gaps are larger. These trends also hold for Ophthalmology, the optimality gaps for which are given in tables F.1 and F.2. For Urology, however, none of the trends in relative or uniform gaps are observed because nearly every single MIP was solved to within the 5% optimality gap

Table 6.6: General Surgery First Fit Heuristic Gap Results for Total Objective

λ	DED			DVED			GND			SND		
	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ
1	1210.94	0.25	0.25	146.98	0.08	0.08	1449.71	0.24	0.24	242.91	0.20	0.20
1.5	1376.01	0.30	0.29	216.23	0.08	0.08	2093.42	0.49	0.48	251.55	0.12	0.12
2	7278.34	1.70	1.63	1306.07	0.10	0.10	13292.76	1.96	1.84	1190.89	0.17	0.17
Mean	3288.43	0.75	0.72	556.43	0.09	0.09	5611.96	0.89	0.85	561.79	0.16	0.16

Table 6.7: General Surgery First Fit Heuristic Gap Results for CVaR Objective

β	DED			DVED			GND			SND		
	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ
0.8	2.43	0.33	0.23	0.68	0.55	0.13	2.19	0.24	0.16	1.14	0.18	0.13
0.9	4.48	0.77	0.43	0.87	1.06	0.16	4.13	0.54	0.36	1.81	1.06	0.33
0.95	7.64	1.44	0.77	0.91	0.49	0.14	7.05	1.30	0.66	2.70	2.95	0.55
Mean	4.85	0.85	0.48	0.82	0.70	0.14	4.46	0.69	0.39	1.88	1.40	0.34

and all objective, estimation method, and parameter combinations have mean relative and uniform gaps of 0.05 or less as seen in tables F.3 and F.4.

To investigate the quality of solutions, the first fit heuristic solutions are compared to those from Gurobi solving the MIPs. Tables 6.6 and 6.7 show mean of the absolute, relative and uniform gaps between the objective function value of the solution found using the first fit heuristic, and the best solution returned by Gurobi, for the Total and CVaR objective functions respectively.

Table 6.6 shows that for all four estimation methods the absolute, relative, and uniform gaps all increase as λ increases. In addition all three gaps are much larger for the DED and GND methods than for the DVED and SND methods. Similar trends are seen for the CVaR objective in table 6.7, with increasing gaps as β increases. The relationships between the methods are not as obvious, however, with no clear trends in the absolute and relative gaps. The uniform gap does have the same trend as the Total objective with larger values for the DED and GND methods and smaller values for the DVED and SND methods, although the differences are smaller for the CVaR objective than the Total.

The relative (and uniform) gaps are much smaller for the other two specialties than for General Surgery, as shown in tables F.5 to F.8. The mean relative gap for Ophthalmology with the Total objective is 0.09, and 0.05 with the CVaR objective, for Urology these relative gaps are 0.01 and 0.26 respectively.

6.4 Discussion

We now discuss the results presented in the previous section and further investigate some of the relationships that were seen.

6.4.1 Constraint Methods as Classifiers

For General Surgery and Ophthalmology the accuracy of the four methods on the historical sample is similar to the accuracy on the future sample (out of sample). For Urology, however, the accuracy on the future sample is much worse than for the historical sample across all four methods.

When evaluating the quality of the four methods of overtime probability estimation as binary classifiers figures 6.7 and 6.8 showed that the SND method, which assumes that a session's duration is normally distributed with mean and variance equal to the sum of the means and variances of the operations assigned to it, performed as well or better than the empirical distribution methods DED and DVED. This improved performance is most noticeable for General Surgery when classifying sessions with a low probability of overtime. This suggests the sessions that the methods are tested on are different than those that they were trained on. The empirical methods were trained using a combined sample of historical and randomly generated sessions, whereas this test is on the sessions that are created by the optimisation models. The difference in the distribution of the assigned duration and variance between the sample that the empirical methods were trained on (training sample), and a random subset of the sessions from the optimisation solutions (solution sample) can be seen in figure 6.22.

Figure 6.22 shows that sessions generated by the optimisation models, referred to as 'solution sessions', have a much smaller range in both assigned session duration and variance, than the training sessions. The fact that there are very few solution sessions with a large assigned session duration or variance means that there are not many sessions that have a high probability of running overtime (>0.6). This explains why in figures 6.7 and 6.8 the accuracy of all the methods for General Surgery and Ophthalmology is very close to 1 when classifying sessions with a probability of overtime above 0.6.

Instead of evaluating the accuracy of the methods as classifiers on the sessions from the optimisation solutions we can evaluate the accuracy on just the historical sessions. This is shown in figure 6.23, where the historical sample of surgeries is also used to calculate the probability of overtime for a session.

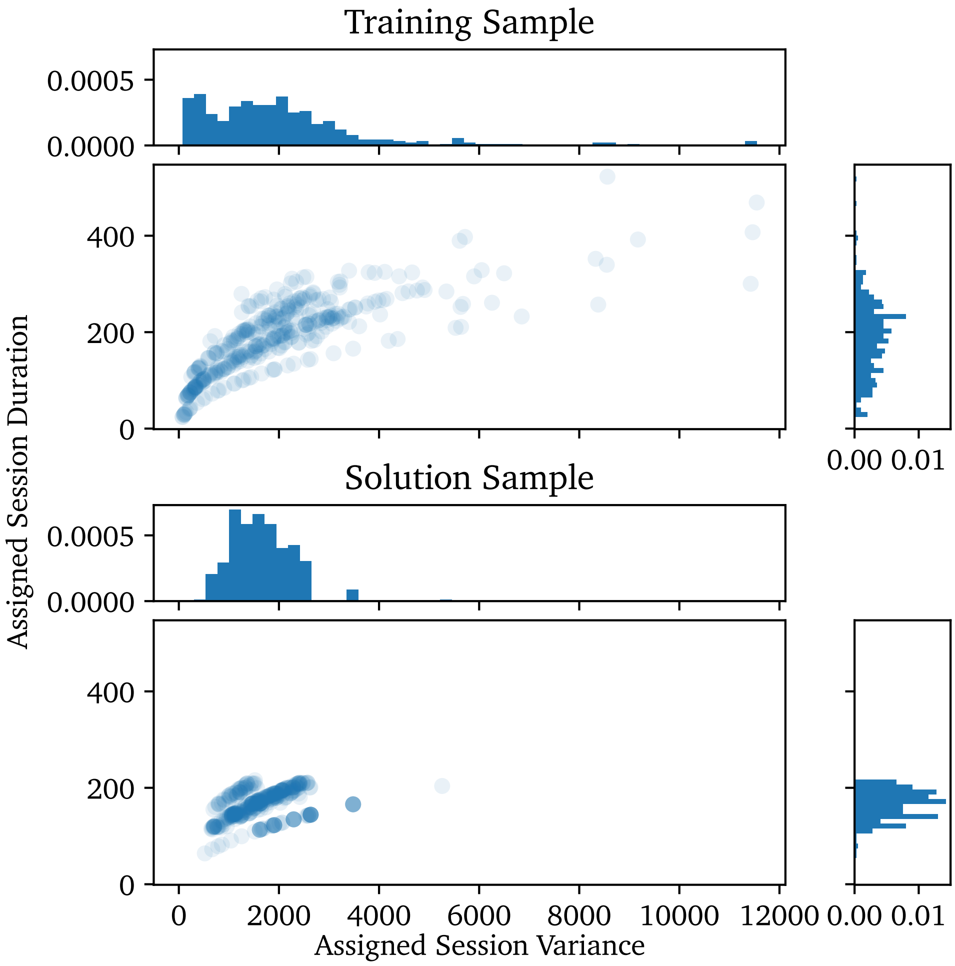


Figure 6.22: Distribution of Assigned Duration and Variance for Training and Solution Sessions

All of the methods show a higher accuracy on just the historical sessions compared to all of the sessions from solving the optimisation problems, for all three specialties. In addition the differences between the four methods are much smaller, with no one method clearly performing better than the others. The fact that the empirical methods have higher accuracies on the historical sessions suggests that if they were trained on a sample of sessions from optimisation solutions then their accuracy on these sessions could be improved. The SND method, on the other hand, does not suffer from as great a decrease in accuracy on the sessions from optimisation solutions, in comparison to the historical sessions. This suggests that the SND method may be more robust to systematic changes in the way that the sessions are scheduled, and therefore the types of sessions that are created. The SND method, therefore, may be useful when there is very little his-

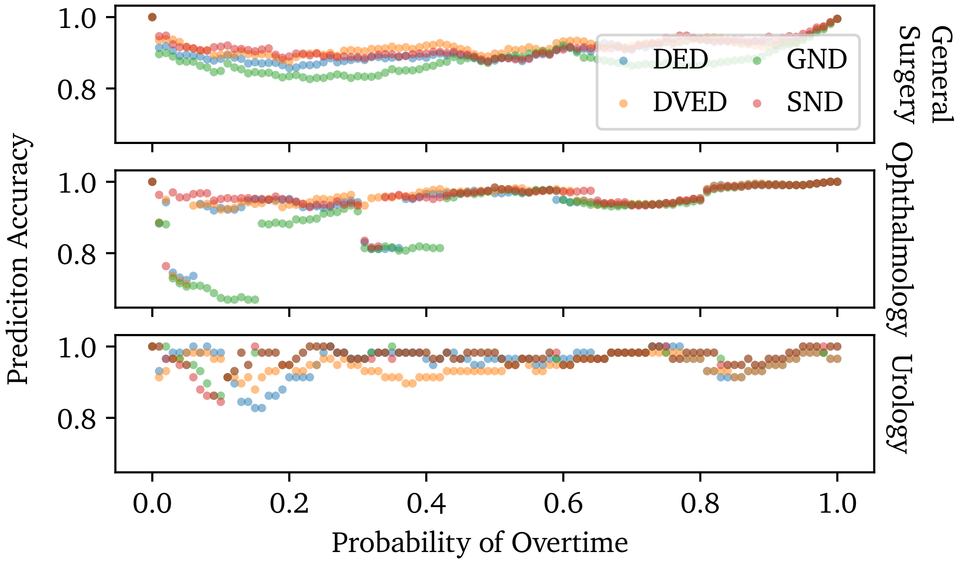


Figure 6.23: Accuracy of Overtime Probability Estimation on Historical Sessions

torical data to work with, or if there is going to be a known systematic change in the way that operations are performed or scheduled.

The consistency of the accuracies of all the methods between the historical surgery sample and the future sample for General Surgery and Ophthalmology, seen in figures 6.7 and 6.8, shows that the methods are, in general, not over fit to the operations duration distribution. Conversely, the large decrease in accuracy of all the methods for the Urology sessions between the historical surgery sample and the future surgery sample is not surprising considering that the distribution of the operations durations is very different in the historical and future samples, as seen in figure 6.2. In particular the fact that the probability of overtime is underestimated is to be expected since the average duration of operations increases in the future sample. As mentioned earlier the decrease in accuracy for Urology could be lessened by periodically re-training the overtime estimation methods. As new data becomes available it can be added to the training data, perhaps with more weight than the older data. This would allow the methods to slowly adapt to changes in the distribution of operation durations.

Finally the results from figures 6.9 and 6.10 show that the methods over estimate the probability of overtime on average for low probability sessions, and under estimate it for high probability sessions. This means that, on average, all of the methods are more restrictive than necessary on sessions with a low simulated probability of overtime. It is these sessions that are the most important as, in practice, OR managers are unlikely to be trying to schedule sessions with more than a 50% probability of running overtime.

6.4.2 2016 Schedule

As noted earlier, in practice, instead of solving the problem described in chapter 5, it may be desirable to solve a multi-objective problem with objectives such as: mean lateness, throughput, risk-adjusted (CVaR) lateness, or mean overtime probability. Figure 6.11 shows that, for mean lateness and throughput, this is unnecessary as they are highly linearly correlated with throughput increasing as mean lateness decreases.

If, however, the three objectives are mean lateness, CVaR at 95% lateness, and mean overtime probability, then figure 6.24 shows the efficient solutions that were found using the various combinations of parameters (the efficient solutions for Ophthalmology and Urology are given in appendix E).

From figure 6.24 it is clear that there is a large range of CVaR at 95% lateness between about 100 and 500 where no efficient solutions were found, and similar gaps are seen in figures E.1 and E.2. It is possible that there are no solutions in these regions for any of the specialties, however, it is also possible that the combinations of parameters used simply did not find any of the solutions in these regions. If a λ between 1.0 and 1.5 was used for the Total objective, or a β less than 0.8 was used for the CVaR objective it may be possible to find some solutions in this area.

Further exploring the λ values between 1.0 and 1.5, and β values less than 0.8, would also be of interest when considering the distribution of lateness as shown in figures 6.16 and 6.17. For the Total objective the distributions of lateness were very similar for $\lambda = 1.5$ and 2, showing that increasing the exponent on lateness past 1.5 does not improve the latest operations. For the CVaR objective the distributions were similar for all values of β, and notably the distributions were similar to those for the Total objective with $\lambda = 1.5$ or 2. This is notable because it suggests that it may be possible to achieve the risk aversion to lateness sought through the CVaR objective, by using an appropriate λ with the Total objective.

The other major observation from these distributions, that median , interquartile range, and range of the distributions decreases as υ increases, is to be expected because as υ increases more operations will be able to be performed in each session, so the operations will be performed earlier overall. The distributions of

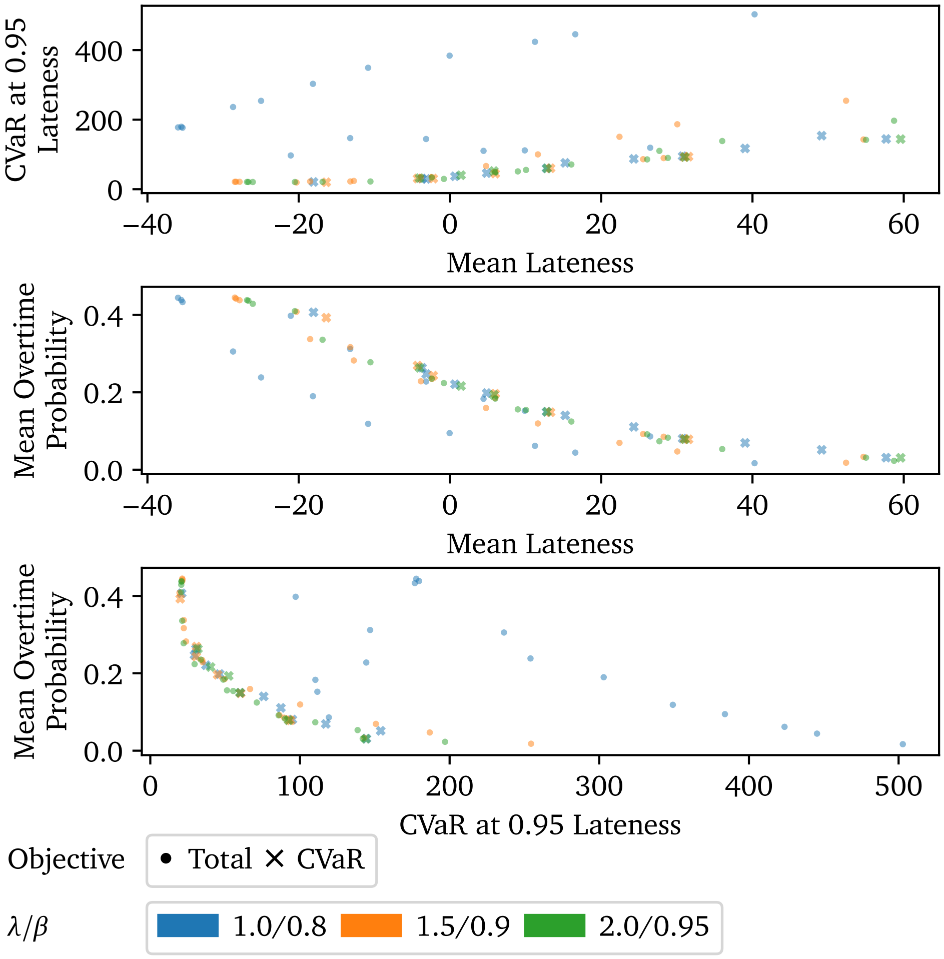


Figure 6.24: Efficient Solutions for General Surgery

overtime probability shown in figures 6.18 and 6.19 show that the only parameter that has a significant effect on this distribution is υ. Once again this is to be expected as υ limits the estimated probability of overtime of the sessions that are scheduled.

The way in which the parameters influence how the lateness of operations changes over time is seen in figures 6.20 and 6.21. By far the most important factor in determining the long term lateness of operations is υ, which again makes sense as it directly affects the number of operations that can be performed in each session. Interestingly, however, λ only appears to influence the short term lateness of operations, and eventually all three values converge to similar smoothed scatter plots, and β has almost no affect at all. Even the methods of estimating the probability of overtime have little effect on the short or long term trends. It can also be seen from these figures that, for General Surgery, a value of around 0.3

or higher for υ is required to prevent the lateness of the operations continually increasing in the long term.

Even though the smoothed lateness has the same long term trends we investigated how the scatter plots themselves differed for three solutions and the actual schedule. The three solutions all used the DED method with $\upsilon = 0.3$: the first two are the Total objective with $\lambda = 1$ and 1.5, the third is the CVaR objective with $\beta = 0.8$. The scatter plots of arrival date of the operations against the lateness of the operations for these three solutions and the actual schedule are shown in figure 6.25, in addition the colour of the points represents the duration of the operation.

When $\lambda = 1$ for the Total objective many operations are allowed to be more than 100 days late, whereas no operations are this late in any of the other schedules. The benefit of allowing these operations to be performed very late, is that the remaining operations are performed earlier. In addition the operations that are more than 100 days late tend to be the longer operations, as seen by the lighter coloured points.

When $\lambda = 1.5$ for the Total objective these very late operations are performed earlier, and there is a high concentration of operations performed just on time (lateness = 0). This is likely because the λ only applies to the cost of an operation being late, and not to it being early where the exponent is still 1. The CVaR objective with $\beta = 0.8$ has a similar profile as the Total objective with $\lambda = 1.5$, however the operations are more spread out around the on time mark. The actual schedule is once again similar to these two but with an even larger spread in operation lateness around 0.

The scatter plots in figure 6.25 suggest that using a λ of 1.5 or greater with the Total objective, or a β of 0.8 or greater with the CVaR objective, not only reduces the lateness of the latest operations, but also reduces the preference for shorter operations to be performed earlier than longer ones.

To further investigate the relationship between operation duration and lateness scatter plots of these two properties, for the three solutions described above and the actual schedule, are shown in figure 6.26. This shows that there is a clear positive relationship between operation duration and lateness when $\lambda = 1$ for the Total objective, but not for the three other schedules.

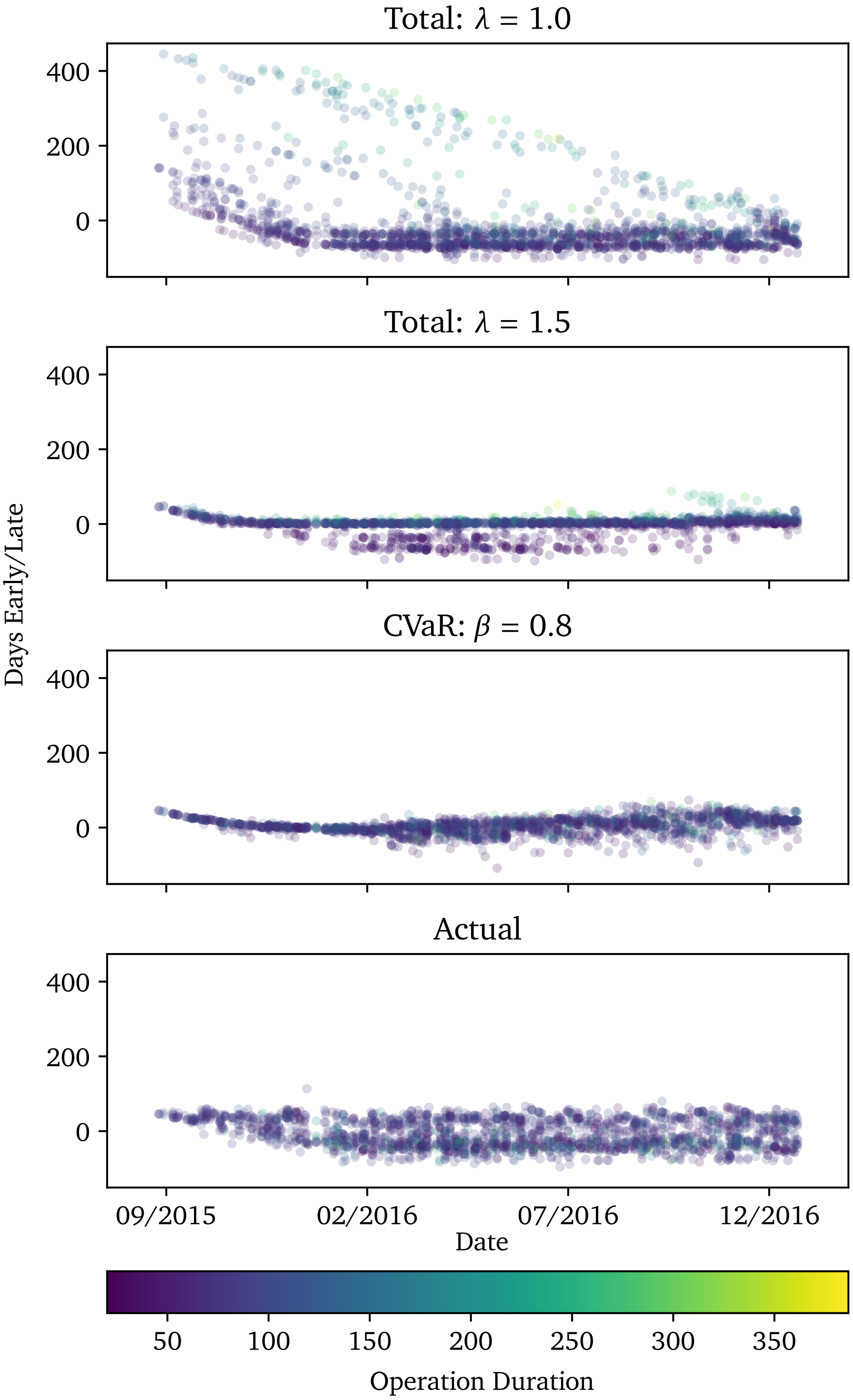


Figure 6.25: Scatter Plots of Arrival Date Against Lateness of Three General Surgery Solutions and the Actual Schedule

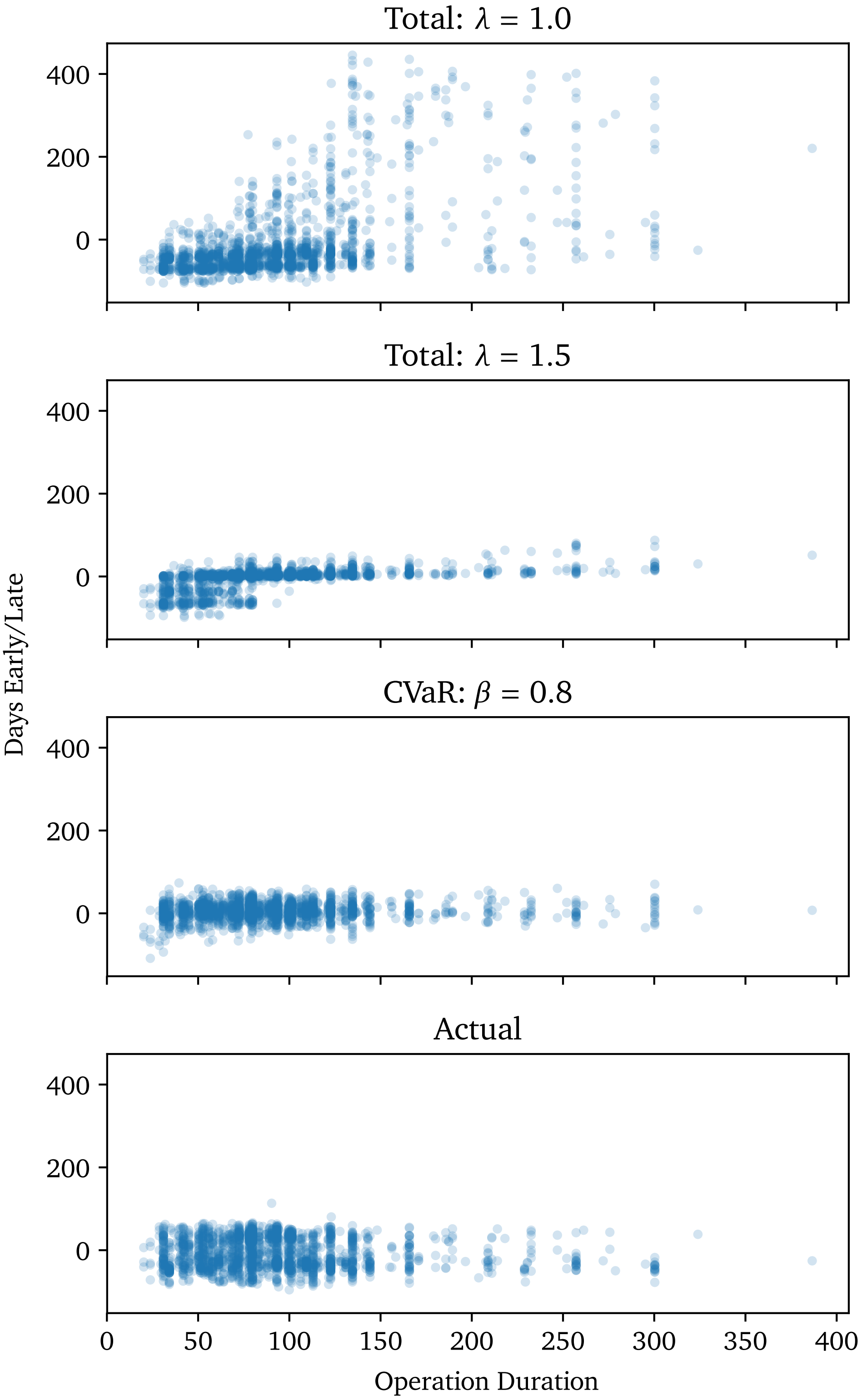


Figure 6.26: Scatter Plots of Operation Duration Against Lateness of Three General Surgery Solutions and the Actual Schedule

6.4.3 Optimality Gaps

In general the DED and GND methods had smaller relative and uniform gaps than the DVED and SND methods. This is to be expected as the DVED and SND methods required SOS2 constraints to model a piecewise linear function of assigned session duration and variance that constrained the possible sessions. These SOS2 constraints are handled internally by Gurobi in the branch and bound process, but nevertheless make the problems more difficult to solve. Using an alternative formulation for the piecewise linear models, for example one of the formulations suggested by Vielma, Ahmed, and Nemhauser (2010), could make these problems easier to solve.

The problems with the CVaR objective have larger relative and uniform gaps, than those with the Total objective. The larger relative gaps are, in part, due to problems where the lower bound is negative while the upper bound is small and positive. For these problems the uniform gap is much smaller than the relative gap. It also to be expected, however, that the problems with the CVaR objective are more difficult to solve than those with the Total objective since the CVaR objective requires additional variables and constraints to be modelled. Further since only $(1-\beta)$% of the operations contribute directly to the objective function, some effort in searching the branch and bound tree is spent deciding on the assignment of operations that do not directly influence the objective function.

The exception to both of these observations is the Urology specialty, where almost every single problem was solved to within the 5% relative gap within the 5 minute time limit. This is because Urology has much fewer operations to schedule and sessions in which to schedule them. As shown in table 6.1, there are only 269 Urology operations compared to 1516 for Ophthalmology and 1773 for General Surgery, and there are only 35 Urology sessions compared to 350 for Ophthalmology and 503 for General Surgery.

In order to further investigate the difficulty of some of the problems that did not reach the 5% relative gap within the 5 minute time limit we re-solved three General Surgery problems with a two hour (7200 seconds) time limit. The first problem was with the DED method, the CVaR objective, and $\beta = 0.8$, and had a relative gap of 18.9% in the original 5 minute solve. The other two problems used the DVED method with the Total and CVaR objective functions, and $\lambda = 1$ and $\beta = 0.8$ respectively. These two problems had relative gaps of 88.38% and 26.54% respectively at the end of the 5 minute time limits. These three problems all had an overtime probability limit of 0.5. Figure 6.27 shows the progression of the upper and lower bounds of the three problems over two hours.

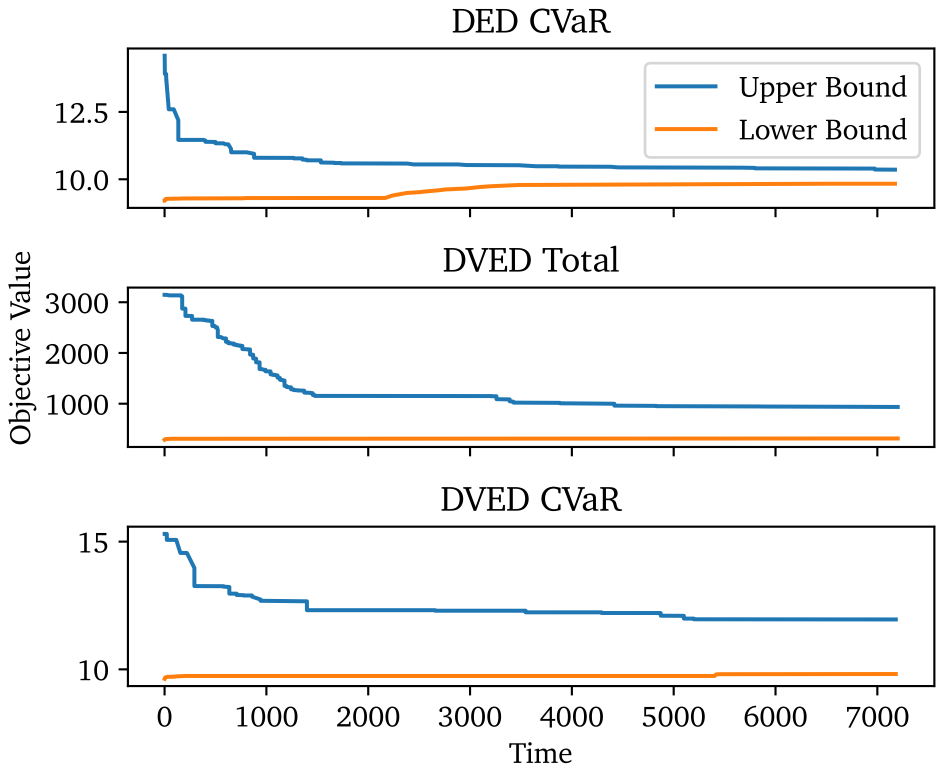


Figure 6.27: Upper and Lower Bounds of Three Problems with a Time Limit of 2 Hours

All three of the problems showed significant improvement in the upper bound in the first 1500 seconds of solving, and relatively little improvement from then on. The lower bounds, on the other hand, did not improve very much over the whole two hours, particularly for the two problems that used the DVED method, and therefore had SOS2 constraints. None of the problems reached the 5% relative gap threshold in two hours, however the DED CVaR problem came close with a finishing gap of 5.04%, while the other two problems finished with gaps of 66.38% and 17.87% respectively.

While the results presented above are only for three problems, and may not be representative of all of the problems that did not reach 5%, they suggest that approaches that focus on improving the lower bound could improve the solution process. Such approaches include reformulating the MIP so the linear programme (LP) relaxation provides a tighter lower bound, and custom branching methods.

Finally, the first fit heuristic, when compared to the MIPs, performed poorly on the General Surgery problems with the Total objective and the DED and GND methods (relative gaps of 75% and 89%), but better on those with the DVED and SND methods (9% and 16%). The performance of the first fit heuristic was more even across the four methods for the CVaR objective problems, with relative gaps between 69% and 140%. This reflects the MIP results where the relative gaps are smaller for the DED and GND methods than the DVED and SND methods for the Total objective, and are larger and more even for the CVaR objective. Once again this suggests that the DED and GND MIPs with the Total objective are easier to solve and can therefore find solutions that are much better than the first fit heuristic. The remaining MIPs are more difficult and it is harder to find any solution let alone good solutions. Hence, the first fit heuristic solutions compare more favourably to the MIP solution because the MIPs can not improve the first fit solution very quickly.

For the other two specialties the first fit heuristic gaps are smaller, between 3% and 11%. This performance on the Ophthalmology problems is somewhat surprising given that the MIPs are not much easier than those for General Surgery, judging by the optimality gaps of the MIPs and the number of operations and sessions to schedule. One possible cause for the improved performance is that the average duration of Ophthalmology operations is much smaller than for General Surgery (36 minutes compared to 92), and about two thirds of Ophthalmology operations are the same type of operation. Both of these factors could make it easier for the first fit heuristic to schedule operations into sessions as efficiently as the MIPs, however further investigation is required to determine if this is the case. On the other hand, the effectiveness of the first fit heuristic on the Urology problems is not as surprising, given that there very few operations and sessions to be scheduled, and therefore little room for improvement by the MIPs.

While discussing the first fit heuristic it should be noted that it was designed simply to provide a feasible solution as an incumbent to the MIPs, and as such is very rudimentary. Further work into incorporating the type of objective function being considered and the value of λ or β used would likely be able to improve the heuristic's performance.

Conclusion

This chapter presented a case study of the application of the surgery scheduling model developed in chapter 5. The model was used to schedule operations for three surgical specialties at the Manukau Surgery Centre (MSC). The incorporation of historical data into the estimations of operation durations and session overtime probability enabled better solutions to be found.

Analysis of the four methods of estimating the probability that a session runs overtime, presented in section 5.2.2, as binary classifiers showed that in sample accuracy of over 90% can be achieved by all the methods. Testing on a different operation sample showed that the accuracy is not greatly affected if the new sample is similar to the training sample. On the other hand, testing on sessions from the optimisation models (as opposed to the historical sessions used for training), showed that the way sessions are created affects the accuracy. Adding some sessions from the optimisation models into the set of training sessions for empirical methods could improve their accuracy on the these sessions (ones from the optimisation models), although this would first require solving an optimisation model to obtain the sessions.

When scheduling operations in 2016, schedules were found that outperform the actual schedule in terms of all three metrics of: mean lateness; CVaR at 95% lateness; and mean overtime probability. It was also shown that a λ of 1.5 is enough to greatly reduce the lateness of the latest operations, and reduce the bias towards short operations. Using a λ of 1.5 or 2 was shown to have very similar results to using the CVaR objective with all the values of β tested. In addition the Total objective problems were easier to solve and therefore present an attractive alternative to the CVaR objective, particularly in practical settings.

Investigation of the optimality gaps achieved when solving the various problems showed that the overtime probability estimation methods that required SOS2 constraints were more difficult to solve than those that did not, and the CVaR objective problems were, in general, more difficult than the Total objective ones. With the added difficulty of solving the respective MIPs in mind, the marginal additional accuracy when classifying overtime sessions given by the DVED and SND methods does not seem worthwhile to use in practice especially compared to the DED method.

The DED method appears to give the best trade off between accuracy of classifying overtime sessions and difficulty in solving the resulting optimisation models. In addition the Total objective with $\lambda = 1.5$ prevents operations from being performed very late, without making the models more difficult to solve. This completes our investigation of surgery scheduling and we present a concluding discussion of the whole thesis in the following chapter.

— Chapter 7 —

Concluding Discussion

Throughout this thesis we have discussed the utilisation of historical data in optimisation models for planning in healthcare. First we considered personnel scheduling in the context of rostering general medicine (GM) physicians at two hospitals. Second the scheduling of surgeries was examined at an elective surgical facility. This chapter summarises the research carried out in these two areas, highlights the major contributions to scheduling research, and compares the two areas.

7.1 General Medicine Rostering

In chapter 3 we introduced a mixed integer programme (MIP) for creating cyclic rosters for physicians in GM departments in hospitals. The rostering model was then applied to two case studies in chapter 4. The first case study, at Waitakere Hospital (WTH) in Waitemata District Health Board (WDHB), shows the full implementation of the model, whereas the second, at Auckland City Hospital (ACH) in Auckland District Health Board (ADHB), demonstrates the definition of parameters for some features of the model that are not required for the first case study.

7.1.1 Contributions

Embedded within the rostering MIP is a model of the patient pathways through the GM department. This patient flow model creates a link between the admission and discharge of patients, and the rosters of the physicians, which allows the number of patients that the physicians are caring for to be calculated. This novel use of patient data in physician rostering enables rosters to be created that balance the patient workloads between the physicians, and represents progress towards the objective of incorporating patient data into healthcare planning models.

Uncertainty in the number of arrivals and discharges throughout the planning horizon was taken into account by generating potential scenarios from the historical data. The scenarios are aggregated in the MIP either by averaging the cost of the roster in all of the scenarios, or using a risk measure, called conditional value at risk (CVaR), that only considers the worst scenarios. Both of these contributions are advances of the objective of incorporating uncertainty and risk into healthcare planning models. Further, a Benders' decomposition was presented where each scenario is represented in a sub-problem (SP) and the solutions, the physicians' workloads, can be found through straightforward calculations.

In order to achieve the objective of determining the value of the developed models a roster was created using this model and historical patient data from WDHB. The rosters created using this model performed 8% better on average, and 10% better in the worst scenarios, than a traditionally constructed roster. Further, one of the rosters created was put in place in the GM department at WTH.

7.1.2 Insights

The results of the first case study showed that using the optimisation model to create a roster for the physicians was a significant improvement over a hand made roster. On the other hand, including uncertainty in admissions and discharges in the optimisation model, at least in the scenarios that we generated, was only a marginal improvement over using the expected value. In addition the risk averse objective did not significantly improve the performance in the worst case scenarios. Future application of the model to additional cases where there is greater flexibility in feasible rosters and variation in admissions and discharges, may show that solving these more difficult problems is worthwhile.

7.1.3 Further Research

The rostering model assumes both that there is a fixed ordering of physicians, and that they work a cyclic roster. Both of these assumptions can be relaxed by extending the decision variables of the MIP. It is also assumed that the pathways of the patients are independent of the current workloads of the physicians. This is not always the case, as some departments have rules that patients will be transferred between physicians if one of them is caring for too many patients. The types of rostering rules (linked, following, exclusive, etc.), and therefore constraints, considered are not exhaustive and therefore do not cover all possibilities. A more general approach that allows any regular expression of shifts to be enforced or banned could be investigated. Other generalisations of the model are also possible, for example normalisation of the workloads of the workload groups by the number of physicians in each group. This normalisation has not yet been included as, in the settings we have encountered, all workload groups have the same number of physicians, however in other settings this may not be the case.

Although the Benders' decomposition reduced the time required to solve the MIP, some of the larger problems, particularly when there were many scenarios, were still difficult to solve. As the SPs are already solved very efficiently, additional effort into solving the master problem (MP), such as using a network representation, could prove fruitful.

7.2 Surgery Scheduling

Chapter 5 presented another MIP, this one for scheduling surgical operations into sessions. Historical data is used throughout the model, to estimate both the duration, and variance of the duration, of the operations, and the probability that a session, with operations assigned to it, runs overtime. The application of this model to scheduling operations at the Manukau Surgery Centre (MSC) in Counties Manukau District Health Board (CMDHB) is demonstrated in chapter 6.

7.2.1 Contributions

Two objective functions were considered for the MIP. The first minimises the total earliness/lateness of the operations, subject to an exponent on the lateness. The second, novel, objective applies the same risk measure used in the physician rostering problem, CVaR, to the latenesses of the operations, in order to focus the objective function on just the latest operations, and is a contribution to the thesis' objective of incorporating risk into healthcare planning models.

The uncertainty of the operations' durations is taken into account in chance constraints in the MIP. The chance constraints prevent sessions from being scheduled that have an estimated probability of running overtime greater than a prescribed value. Four methods for estimating the probability that a session runs overtime are presented, given the sum of the durations (and variances) of the operations assigned to the session. Of the four methods two – grand normal distribution (GND) and session normal distribution (SND) – rely on normal distribution assumptions, and the other two novel methods – duration empirical distribution (DED) and duration variance empirical distribution (DVED) – use empirical distributions estimated from historical sessions and operations. The two empirical distribution methods represent progress towards the objectives of utilising patient data and incorporating uncertainty into healthcare planning models.

The results of using the optimisation model to schedule operations throughout 2016 showed that schedules could be created that outperformed the actual schedule in terms of throughput, probability of sessions running overtime, and the lateness of the latest 5% of operations. The four methods of estimating the probability that a session runs overtime were all able to achieve accuracies of over 90% as binary classifiers, however the SND method performed the best on out of sample tests. These are both examples of achieving the objective of determining the value of the developed models in a practical setting.

The uptake of these modelling techniques in practical settings has been difficult to achieve, as surgical service operators are reluctant to relinquish scheduling decisions to a mathematical model. Simpler tools, however, that allow operators to maintain more control have had more success in their implementation. Further, encouraging initial discussions with operators at other facilities indicate that not all surgical services are opposed to the adoption of these methods.

7.2.2 Insights

In the case study considered we found that using an exponent applied to the lateness of operations produced similar results to a risk measure. This is useful as the application of an exponent to lateness is both simpler to understand and easier to implement. In addition we found that in the long term the lateness of operations depended mainly on the allowed probability of sessions running overtime, rather than the objective function used. This is because the long term lateness depends on the overall capacity or throughput of the system which is reduced if the accepted probability of overtime is lowered.

We also found that, when the variance of operations' durations is taken into account, using normal approximations performed very slightly better than an empirical distribution based method. When only the mean of operations' durations is accounted for, however, the empirical method is superior. Given that the difference in accuracy between the empirical method without variance and both the methods with variance is very small, but the additional effort to solve the variance-inclusive methods is significant, the empirical method without variance appears to be best suited for use in practice, in this instance.

7.2.3 Further Research

It was assumed that the master surgery schedule, which is the allocation of sessions to surgical specialties, had been determined prior to the scheduling of the operations. This meant that the overall scheduling problem could be decomposed by specialty, as there is no interaction between them. Instead we could allow the model to decide the specialty of each session, effectively creating the master surgery schedule at the same time as scheduling the operations. This would mean that the problem could no longer be decomposed by specialty, and would likely make it much more difficult to solve. Further it was assumed that no other factors, in particular the availability of surgeons, limited the assignment of operations to sessions. Surgeon availability could be taken into account in a similar manner to how we currently deal with the master surgery schedule. That is, each session could be allocated to not only a specialty but also a surgeon(s), and operations could be restricted to only those sessions that have been allocated to an appropriate surgeon.

Sources of uncertainty other than the duration of operations, such as the number of new operations that will arrive in the future, are not taken into account. To take operations that are not currently known about into account, space in future sessions could be reserved so that they are not completely filled, and the new arrivals could then be assigned to these spaces.

The chance constraints included in the model only limit the probability of sessions running overtime. Other constraints on overtime may be of interest, such as limiting the probability that a session runs more than 30 minutes overtime, or restricting a risk measure applied to overtime. For example multiple constraints on the probability of a session's duration exceeding various levels could be included, such as a constraint that limits the probability of any overtime to 10% and one that limits the probability of 30 minutes or more of overtime to 5%. In this type of case, combined with one of the methods that only uses the assigned session duration, one of the constraints will always be more restrictive than the remaining ones, making them redundant. For the methods that use the assigned session variance as well, however, including more than one of these constraints may be useful. Another approach to constraining overtime that may be of interest would be to consider all of the sessions scheduled as a 'portfolio' and restrict some properties of this portfolio. This may be attractive to operating room (OR) management if, for example, they are willing to sacrifice some sessions as definitely running overtime if it means that the remaining sessions all run on time.

Using historical session and operation data we defined empirical conditional probability distributions that a session runs overtime given the duration and variance of the operations assigned to it. Combining these empirical methods with the methods that assume normal distributions could improve the accuracy further, if, for example, the empirical methods are used when there is plenty of historical data, and the normal assumptions are used when the data is sparse, i.e., a hybrid approach to estimating session overtime probability.

Instead of using this approach, constraints could be formed directly from the historical data using machine learning techniques such as support vector machines. Given an acceptable probability of overtime, the historical sessions can be classified as either feasible or infeasible. A hyperplane can then be found that best separates the feasible sessions from the infeasible and further conditions can be placed on this hyperplane so that it can be easily incorporated into a MIP as a constraint.

7.3 Comparison

While the focus of this thesis has been on the application of historical data in scheduling models, it has also provided an opportunity to consider the similarities and differences between the two models, and problems, considered. In chapter 1 we briefly compared the overall approaches in the domains of personnel and surgery scheduling, here we compare and contrast the specific models we developed.

Both problems are modelled with MIPs, which means they both benefit from, and fall victim to, the strengths and weaknesses of this framework. The main benefits of modelling using a MIP are: the guarantee of optimality given long enough solution time; the capability to model many different types of decision processes; the availability of high quality solvers, for example Gurobi and CPLEX; and the quantity of research into integer programming techniques. On the other hand, a limitation of using MIPs, as seen in the surgery scheduling problem, is that it can be very difficult to solve them to optimality, and a lot of effort can be spent attempting to improve the solution process.

While both problems are modelled with MIPs the ways that uncertainty is taken into account in each problem presents a point of difference. The physician rostering model represents the uncertainty in admissions and discharges using scenarios, whereas the surgery scheduling model uses probability density functions.

Scenarios are used to represent the uncertainty in the physician rostering model because defining the required parametric or empirical distribution functions to link the admissions and discharges to the workloads of the physicians would require finding expressions for the behaviour of a complex queue system that represents the patients' pathways. The definition of such expressions is itself a difficult task, and there is no guarantee, given that the task is achievable, that they can be implemented in a MIP. In addition the scenarios are able to be represented as sub-problems in a Benders' decomposition, as they do not require any integer variables. This means that the problem can still be solved efficiently even when many scenarios are included.

Conversely, in the surgery scheduling model both parametric and empirical probability distribution functions can be constructed that link the operations assigned to a session to the probability that the session runs overtime. It is then straightforward to use these functions in chance constraints in the MIP. If scenarios of operation duration were to be utilised instead, it would require the addition of a constraint on the duration of a session for each scenario. Each constraint would use a different duration for the operations, coming from the different scenarios. A further constraint would be required to ensure that the session does not run overtime in more than the allowed fraction of scenarios. This formulation presents more difficulty than the corresponding physician rostering one, as there are integer variables in the scenario sub-problems, although a column generation approach could be utilised to overcome this difficulty. Also of concern is the feasibility of the MIP. Given a realisation of operation duration scenarios, it may not be possible to assign them to sessions in a way that the sessions do not run overtime in many scenarios. This is particularly likely if a small number of scenarios are used, as just a few large operation durations could make the model infeasible. In contrast, the physician rostering model does not have this problem as the SPs are evaluating the performance of a roster in a scenario, and do not effect the feasibility of the overall model.

One similarity between the models when considering uncertainty is that both use a resampling procedure applied to historical data. For the physician rostering model the resampling is used to create scenarios of patient admissions and discharges that are used directly in the model to represent the uncertainty in these processes. On the other hand, the surgery scheduling model uses resampling of historical operation durations to create the empirical distributions of the probability that a session runs overtime.

The difference in the approaches to risk are not as stark. In both instances a risk averse objective function, CVaR, is used to target the worst outcomes. The difference is that in the physician rostering model the worst outcomes relate to the scenarios in which the maximum difference in workloads is the greatest, whereas, in the surgery scheduling model the worst outcomes are the latest operations, and are not directly related to the uncertainty being modelled.

The final similarity, and perhaps most important in terms of practical usefulness, is that both models provide a test bed in which one can experiment with policy and procedural changes. This ability was touched on previously for both models. First the physician rostering model was used to compare the performance of rosters with different numbers of physicians and structures of the department. Second the surgery scheduling model showed the effects of using different penalties for lateness and values for the allowed probability of overtime on the surgery schedules. These parameters can be chosen by OR management to reflect their priorities and create a schedule that they find acceptable.

Many more opportunities for experimentation are offered by the two models, however, and both models also produce secondary output that may be valuable. In the physician rostering model, for instance, the effect of new roster rules, which may be the result of law changes or a new collective agreement with the employees, can be investigated. Also the scenarios could be used to represent the future increases in admissions due to a growing population. In this situation, in particular, the fact that the model not only calculates the largest difference in workloads, but also the workloads of all the physicians throughout the planning horizon would be useful for the department management. The workloads of all the physicians can aid in planning for other resources, particularly the bed capacity of the wards.

The surgery scheduling model also provides further scope for analysing changes to the OR scheduling process. More sessions can easily be added to the model, for example, or the current sessions can be made longer. One adjustment that may be of particular interest is the use of a morning and afternoon session, as compared to an all day session. In addition the schedules of operations assigned to sessions produced by the model can once again be used when planning other long term resources, such as staff and equipment levels, as they give a precise estimate of how long it will take to perform the currently waiting operations.

Both models make use of the data that is growing in volume and increasing in granularity to provide rostering and scheduling decision making environments. By embedding actual data into the decision making process these models provide data-informed insights into the effect of decisions and enable decision makers to have improved confidence in key decisions within the healthcare sector.

Appendices

— Appendix A —

Benders’ Decomposition Results

This appendix provides the dual transformation of the primal sub-problem (SP) of the Benders’ decomposition proposed in 3.3, and shows that it is always both feasible and bounded. Given the primal form of the SP below, we first show how the dual of this problem is formed.

$$
\begin{aligned}
\min \quad & \sum_{z \in Z} (\bar{M}^{(z)}), & & \\
\text{s.t.} \quad N^{(z)}_{l,k,c,p} &= \mathcal{B}^{(z)}_{l,k,c,p} \bar{x}, & & \forall l \in L, k \in K, c \in C, p \in P, z \in Z, \\
& & & \qquad (3.10') \\
M^{+(z)}_{c,p} - \sum_{k \in K} (N^{(z)}_{l,k,c,p}) &\geq 0, & & \forall l \in L, c \in C, p \in P, z \in Z, \qquad (3.11) \\
\sum_{k \in K} (N^{(z)}_{l,k,c,p}) - M^{-(z)}_{c,p} &\geq 0, & & \forall l \in L, c \in C, p \in P, z \in Z, \qquad (3.12) \\
\bar{M}^{(z)} - (M^{+(z)}_{c,p} - M^{-(z)}_{c,p}) &\geq 0, & & \forall c \in C, p \in P, z \in Z, \qquad (3.13) \\
N, M^{+}, M^{-}, \bar{M} &\geq 0. & & \qquad (3.17)
\end{aligned}
$$

We assume that the dual variables $\pi^1, \pi^2, \pi^3, \pi^4$ correspond to constraints (3.10′)–(3.13), and note that they are indexed by the sets that are used to enumerate the constraints. The objective of the dual is the sum of these new variables multiplied by the right hand sides of the corresponding constraints. The only right hand side that is non-zero is for constraint (3.10′), which is $\mathcal{B}^{(z)}_{l,k,c,p}\bar{x}$, and the corresponding dual variable is π^1. The dual objective is therefore

$$
\max \sum_{\forall l \in L, k \in K, c \in C, p \in P, z \in Z} \left(\pi^{(1,z)}_{l,k,c,p} (\mathcal{B}^{(z)}_{l,k,c,p} \bar{x}) \right),
$$

or equivalently using vector notation

$$\max \sum_{z\in Z} \left((\pi^{(1,z)})^\intercal (\mathcal{B}^{(z)}\bar{x}) \right).$$

To transform the constraints we start by noting that the all of the variables in the primal problem are ≥ 0 which means that all of the dual constraints are $\leq$, additionally we should be able to form four constraints corresponding to the four primal variables. In addition because the right hand sides of the constraints correspond to the objective coefficients in the primal problem, all of the right hand sides are zero except for the fourth constraint relating to the primal variable $\bar{M}^{(z)}$.

The first of the constraints relates to the variable $N^{(z)}_{l,k,c,p}$ which is involved in constraints (3.10′), (3.11), and (3.12), and as the second two constraints are not indexed by K but the first is we need to sum the dual variables for the first constraint over K, which gives the constraints below:

$$\sum_{k\in K}(\pi^{(1,z)}_{l,k,c,p}) - \pi^{(2,z)}_{l,c,p} + \pi^{(3,z)}_{l,c,p} \leq 0, \qquad \forall l \in L, c \in C, p \in P, z \in Z.$$

The second of the dual constraints is related to the primal variable $M^{+(z)}_{c,p}$, which is used the the constraints (3.11) and (3.13), once again the indices are different so a summation is required, this time over L.

$$\sum_{l\in L}(\pi^{(2,z)}_{l,c,p}) - \pi^{(4,z)}_{c,p} \leq 0, \forall c \in C, p \in P, z \in Z.$$

The third constraint is very similar to the second, with the only changes the signs on the variables and $\pi^{(3)}$ instead of $\pi^{(2)}$.

$$\pi^{(4,z)}_{c,p} - \sum_{l\in L}(\pi^{(3,z)}_{l,c,p}) \leq 0, \forall c \in C, p \in P, z \in Z.$$

The last constraint relates to the primal variable $\bar{M}^{(z)}$, and has a right hand side of 1 and a summation over C and P because of the difference in indices:

$$\sum_{c\in C, p\in P}(\pi^{(4,z)}_{c,p}) \leq 1, \forall z \in Z.$$

Finally the dual variables π^2, π^3, π^4 are ≥ 0 and π^1 is free, because of the relations of the primal constraints. Combining all of these constraints together gives the full dual SP:

$$
\begin{aligned}
\max \quad & \sum_{z \in Z} \left((\pi^{(1,z)})^\intercal (\mathcal{B}^{(z)} \bar{x}) \right), && && (3.21)\\
\text{s.t.} \quad & \sum_{k \in K} (\pi^{(1,z)}_{l,k,c,p}) - \pi^{(2,z)}_{l,c,p} + \pi^{(3,z)}_{l,c,p} \leq 0, && \forall l \in L, c \in C, p \in P, z \in Z, && (3.22)\\
& \sum_{l \in L} (\pi^{(2,z)}_{l,c,p}) - \pi^{(4,z)}_{c,p} \leq 0, && \forall c \in C, p \in P, z \in Z, && (3.23)\\
& \pi^{(4,z)}_{c,p} - \sum_{l \in L} (\pi^{(3,z)}_{l,c,p}) \leq 0, && \forall c \in C, p \in P, z \in Z, && (3.24)\\
& \sum_{c \in C, p \in P} (\pi^{(4,z)}_{c,p}) \leq 1, && \forall z \in Z, && (3.25)\\
& \pi^1 \text{ free}, \pi^2, \pi^3, \pi^4 \geq 0.
\end{aligned}
$$

The solution $\pi^1 = \pi^2 = \pi^3 = \pi^4 = 0$ is always feasible. In addition because of the assumption that $\mathcal{B}^{(z)}\bar{x}$ is always positive, combined with the maximisation objective mean that π^1 will always be positive or 0. This means that the problem is bounded as π^4 is bounded below by 0 and above by 1, π^2 is bounded below by 0 and above by π^4, and π^1 and π^3 are both bounded below by 0 and above by π^2. This means that the dual SP is always feasible and bounded and therefore there is always an optimal solution.

— Appendix B —

Surgery Scheduling Results Tables

This appendix provides tables of results from scheduling the operations of General Surgery, Ophthalmology, and Urology at Manukau Surgery Centre (MSC) throughout 2016. The results presented are: the mean lateness of operations scheduled in days, the conditional value at risk (CVaR) @ 95% lateness of operations scheduled in days, the mean overtime probability of the sessions scheduled in 2016, the CVaR @ 95% of overtime probability of the sessions scheduled in 2016, the number of operations scheduled to be performed in 2016, and the total duration of operations waiting to be performed in 2017 in 1000s of minutes.

Each table includes the results for one of the four methods for estimating the probability of overtime — duration empirical distribution (DED), duration variance empirical distribution (DVED), grand normal distribution (GND), and session normal distribution (SND) — and one of the two objective functions — Total, and CVaR. For each combination of these two factors all combinations of λ (or β for the CVaR objective) and υ are given in each table.

Table B.1: General Surgery Results for Total Objective and DED Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-36	180	0.44	0.82	1900	18
	0.4	-29	236	0.31	0.67	1840	27
	0.3	-18	303	0.19	0.62	1760	36
	0.2	0	384	0.09	0.50	1620	49
	0.1	17	445	0.04	0.31	1520	61
1.5	0.5	-28	21	0.44	0.84	1860	18
	0.4	-18	22	0.34	0.75	1780	25
	0.3	-4	35	0.23	0.68	1670	34
	0.2	12	100	0.12	0.54	1580	46
	0.1	30	187	0.05	0.33	1480	60
2	0.5	-26	21	0.43	0.83	1850	19
	0.4	-17	21	0.34	0.75	1770	25
	0.3	-1	30	0.22	0.66	1650	34
	0.2	16	71	0.12	0.55	1560	46
	0.1	36	139	0.05	0.36	1450	59

Table B.2: General Surgery Results for Total Objective and DVED Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-21	97	0.40	0.79	1820	22
	0.4	-8	94	0.28	0.66	1710	30
	0.3	4	110	0.18	0.50	1620	39
	0.2	26	119	0.09	0.31	1490	51
	0.1	55	187	0.03	0.20	1340	66
1.5	0.5	-20	19	0.41	0.79	1800	22
	0.4	-7	29	0.28	0.68	1710	30
	0.3	6	50	0.19	0.50	1610	39
	0.2	26	86	0.09	0.33	1500	51
	0.1	55	143	0.03	0.19	1340	66
2	0.5	-21	21	0.41	0.81	1800	21
	0.4	-7	28	0.28	0.66	1700	30
	0.3	6	49	0.18	0.49	1610	39
	0.2	26	86	0.09	0.31	1500	51
	0.1	55	142	0.03	0.18	1340	66

Table B.3: General Surgery Results for Total Objective and GND Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-36	178	0.44	0.84	1900	18
	0.4	-25	254	0.24	0.64	1810	31
	0.3	-11	349	0.12	0.51	1710	43
	0.2	11	424	0.06	0.40	1550	56
	0.1	40	503	0.02	0.16	1430	73
1.5	0.5	-28	21	0.44	0.85	1860	17
	0.4	-13	24	0.28	0.74	1730	29
	0.3	5	67	0.16	0.59	1620	41
	0.2	22	151	0.07	0.44	1530	55
	0.1	52	254	0.02	0.16	1380	72
2	0.5	-27	21	0.44	0.84	1850	18
	0.4	-11	22	0.28	0.73	1720	29
	0.3	9	51	0.16	0.59	1600	41
	0.2	28	110	0.07	0.43	1490	54
	0.1	59	197	0.02	0.20	1340	71

Table B.4: General Surgery Results for Total Objective and SND Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-35	177	0.43	0.83	1900	18
	0.4	-13	147	0.31	0.73	1750	27
	0.3	-3	144	0.23	0.65	1680	34
	0.2	10	112	0.15	0.57	1590	42
	0.1	28	122	0.08	0.41	1480	52
1.5	0.5	-28	21	0.44	0.85	1860	18
	0.4	-13	22	0.32	0.76	1740	26
	0.3	-2	35	0.23	0.65	1670	33
	0.2	10	55	0.16	0.57	1590	41
	0.1	28	89	0.09	0.43	1490	52
2	0.5	-27	21	0.44	0.86	1840	18
	0.4	-13	23	0.32	0.76	1740	26
	0.3	-2	34	0.24	0.65	1670	33
	0.2	10	55	0.15	0.56	1590	41
	0.1	29	90	0.08	0.41	1480	52

Table B.5: General Surgery Results for CVaR Objective and DED Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-19	20	0.43	0.84	1810	19
	0.4	-8	23	0.30	0.76	1720	27
	0.3	5	46	0.20	0.66	1630	37
	0.2	24	87	0.11	0.54	1510	48
	0.1	49	154	0.05	0.35	1380	61
1.5	0.5	-19	20	0.42	0.85	1800	19
	0.4	-8	23	0.30	0.76	1720	27
	0.3	6	45	0.20	0.63	1620	37
	0.2	26	95	0.11	0.52	1490	49
	0.1	53	205	0.05	0.36	1340	61
2	0.5	-19	20	0.42	0.83	1800	19
	0.4	-8	24	0.30	0.76	1720	28
	0.3	6	52	0.19	0.65	1620	37
	0.2	26	108	0.11	0.51	1480	49
	0.1	51	206	0.05	0.34	1360	61

Table B.6: General Surgery Results for CVaR Objective and DVED Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-16	22	0.39	0.79	1780	22
	0.4	-4	31	0.26	0.64	1690	32
	0.3	9	55	0.17	0.48	1600	41
	0.2	29	91	0.08	0.30	1480	52
	0.1	58	144	0.03	0.19	1330	66
1.5	0.5	-16	20	0.39	0.78	1780	22
	0.4	-4	31	0.27	0.66	1690	31
	0.3	11	56	0.16	0.50	1590	41
	0.2	31	93	0.08	0.28	1470	53
	0.1	58	143	0.03	0.19	1320	66
2	0.5	-16	23	0.39	0.76	1770	23
	0.4	-4	32	0.26	0.65	1690	32
	0.3	10	55	0.17	0.48	1590	41
	0.2	31	92	0.08	0.28	1470	53
	0.1	60	144	0.03	0.19	1310	67

Table B.7: General Surgery Results for CVaR Objective and GND Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-21	21	0.44	0.84	1820	18
	0.4	-3	29	0.25	0.70	1680	31
	0.3	15	76	0.14	0.61	1560	43
	0.2	39	117	0.07	0.45	1430	56
	0.1	76	232	0.02	0.21	1260	73
1.5	0.5	-19	21	0.42	0.84	1810	19
	0.4	-2	31	0.24	0.70	1680	32
	0.3	16	82	0.14	0.61	1560	44
	0.2	41	157	0.07	0.43	1400	56
	0.1	77	239	0.02	0.20	1240	73
2	0.5	-20	21	0.43	0.84	1810	18
	0.4	-1	32	0.25	0.70	1670	32
	0.3	16	75	0.14	0.59	1560	44
	0.2	41	162	0.07	0.42	1410	56
	0.1	82	258	0.02	0.20	1210	73

Table B.8: General Surgery Results for CVaR Objective and SND Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-18	21	0.41	0.85	1800	20
	0.4	-10	25	0.31	0.76	1730	27
	0.3	1	37	0.22	0.65	1650	35
	0.2	13	60	0.15	0.57	1570	43
	0.1	31	95	0.08	0.44	1470	53
1.5	0.5	-19	21	0.41	0.84	1800	19
	0.4	-9	24	0.30	0.75	1720	28
	0.3	1	37	0.22	0.64	1650	35
	0.2	13	60	0.15	0.57	1570	43
	0.1	32	94	0.08	0.41	1470	53
2	0.5	-18	21	0.41	0.83	1800	19
	0.4	-9	25	0.30	0.76	1730	27
	0.3	1	40	0.22	0.67	1650	35
	0.2	13	60	0.15	0.56	1570	42
	0.1	31	92	0.08	0.42	1470	53

Table B.9: Ophthalmology Results for Total Objective and DED Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-16	145	0.45	0.82	1800	2
	0.4	-15	148	0.41	0.75	1780	3
	0.3	-1	239	0.18	0.56	1700	10
	0.2	2	252	0.13	0.47	1680	11
	0.1	6	284	0.08	0.31	1650	13
1.5	0.5	-12	46	0.46	0.82	1800	2
	0.4	-9	49	0.41	0.76	1780	3
	0.3	9	57	0.19	0.62	1640	8
	0.2	14	68	0.15	0.60	1600	10
	0.1	18	83	0.10	0.42	1570	12
2	0.5	-12	38	0.47	0.82	1810	2
	0.4	-8	41	0.41	0.76	1770	3
	0.3	10	48	0.20	0.64	1630	8
	0.2	16	51	0.15	0.59	1590	10
	0.1	21	61	0.10	0.45	1550	12

Table B.10: Ophthalmology Results for Total Objective and DVED Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	6	246	0.12	0.50	1640	12
	0.4	7	258	0.11	0.42	1650	12
	0.3	10	253	0.10	0.40	1600	12
	0.2	14	243	0.08	0.32	1580	13
	0.1	29	234	0.05	0.17	1490	16
1.5	0.5	16	76	0.13	0.58	1580	11
	0.4	17	78	0.12	0.56	1570	11
	0.3	19	82	0.11	0.47	1570	12
	0.2	21	89	0.08	0.36	1550	13
	0.1	34	114	0.05	0.22	1470	16
2	0.5	19	58	0.12	0.56	1560	11
	0.4	20	59	0.12	0.51	1560	11
	0.3	21	62	0.10	0.42	1540	12
	0.2	24	65	0.09	0.39	1530	13
	0.1	37	87	0.05	0.26	1460	16

Table B.11: Ophthalmology Results for Total Objective and GND Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-15	149	0.42	0.81	1780	3
	0.4	-1	233	0.18	0.59	1700	9
	0.3	3	258	0.13	0.56	1670	12
	0.2	8	298	0.07	0.26	1630	14
	0.1	43	454	0.02	0.13	1440	22
1.5	0.5	-10	49	0.43	0.78	1790	2
	0.4	8	56	0.20	0.65	1640	8
	0.3	14	68	0.15	0.57	1590	10
	0.2	21	93	0.08	0.39	1550	13
	0.1	50	212	0.02	0.16	1370	21
2	0.5	-9	40	0.42	0.82	1780	2
	0.4	9	47	0.20	0.64	1640	8
	0.3	16	52	0.15	0.61	1590	10
	0.2	24	67	0.08	0.39	1530	13
	0.1	56	153	0.02	0.16	1340	20

Table B.12: Ophthalmology Results for Total Objective and SND Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-15	149	0.42	0.81	1790	3
	0.4	-13	164	0.38	0.73	1780	4
	0.3	1	237	0.19	0.57	1700	9
	0.2	3	259	0.14	0.48	1660	11
	0.1	14	246	0.09	0.34	1590	13
1.5	0.5	-10	48	0.43	0.80	1780	2
	0.4	-7	49	0.38	0.71	1760	4
	0.3	9	53	0.20	0.64	1640	8
	0.2	13	67	0.15	0.59	1600	10
	0.1	20	87	0.09	0.41	1550	12
2	0.5	-9	40	0.42	0.78	1780	3
	0.4	-6	41	0.38	0.72	1760	4
	0.3	10	47	0.20	0.65	1640	8
	0.2	15	52	0.15	0.60	1590	10
	0.1	23	67	0.09	0.43	1530	12

Table B.13: Ophthalmology Results for CVaR Objective and DED Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-11	37	0.48	0.80	1810	1
	0.4	-6	39	0.40	0.73	1770	3
	0.3	12	45	0.19	0.62	1630	8
	0.2	18	48	0.14	0.58	1580	10
	0.1	27	54	0.09	0.36	1520	12
1.5	0.5	-11	37	0.48	0.83	1810	1
	0.4	-5	39	0.40	0.72	1770	3
	0.3	12	47	0.19	0.63	1620	8
	0.2	17	48	0.15	0.60	1590	10
	0.1	27	52	0.09	0.36	1520	12
2	0.5	-11	38	0.48	0.81	1810	1
	0.4	-5	40	0.39	0.73	1760	3
	0.3	12	48	0.19	0.64	1620	8
	0.2	18	50	0.15	0.57	1580	10
	0.1	27	53	0.09	0.36	1530	12

Table B.14: Ophthalmology Results for CVaR Objective and DVED Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-9	37	0.45	0.78	1800	2
	0.4	-3	38	0.38	0.72	1750	4
	0.3	17	45	0.16	0.62	1590	10
	0.2	23	48	0.11	0.45	1540	11
	0.1	36	60	0.07	0.31	1460	14
1.5	0.5	-9	37	0.45	0.80	1790	2
	0.4	-3	38	0.37	0.71	1740	4
	0.3	17	46	0.16	0.60	1590	10
	0.2	22	48	0.12	0.46	1540	11
	0.1	35	62	0.06	0.24	1460	14
2	0.5	-8	38	0.45	0.82	1790	2
	0.4	-3	40	0.37	0.74	1740	4
	0.3	17	46	0.16	0.61	1590	10
	0.2	22	49	0.12	0.45	1540	11
	0.1	35	62	0.07	0.34	1460	14

Table B.15: Ophthalmology Results for CVaR Objective and GND Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-8	38	0.43	0.80	1790	2
	0.4	9	45	0.22	0.64	1650	8
	0.3	17	48	0.15	0.60	1590	10
	0.2	27	53	0.09	0.41	1520	12
	0.1	63	136	0.03	0.18	1310	20
1.5	0.5	-7	39	0.42	0.78	1780	3
	0.4	10	46	0.22	0.66	1640	8
	0.3	17	50	0.15	0.59	1580	10
	0.2	27	52	0.09	0.39	1520	12
	0.1	63	135	0.03	0.18	1310	20
2	0.5	-7	39	0.42	0.81	1780	3
	0.4	10	47	0.21	0.65	1640	8
	0.3	17	50	0.15	0.59	1590	10
	0.2	27	54	0.09	0.41	1520	12
	0.1	63	139	0.03	0.19	1300	20

Table B.16: Ophthalmology Results for CVaR Objective and SND Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-7	39	0.43	0.77	1780	3
	0.4	-2	40	0.36	0.66	1740	4
	0.3	11	44	0.20	0.59	1630	8
	0.2	20	47	0.13	0.56	1560	10
	0.1	27	50	0.09	0.32	1510	12
1.5	0.5	-7	38	0.42	0.76	1780	3
	0.4	-2	39	0.36	0.71	1740	4
	0.3	12	44	0.20	0.60	1630	8
	0.2	20	47	0.14	0.58	1560	11
	0.1	27	50	0.09	0.32	1510	12
2	0.5	-7	40	0.42	0.78	1780	3
	0.4	-3	40	0.37	0.69	1750	4
	0.3	11	45	0.20	0.61	1630	8
	0.2	20	48	0.14	0.57	1560	11
	0.1	27	52	0.09	0.33	1510	12

Table B.17: Urology Results for Total Objective and DED Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	2	267	0.65	0.99	229	2.8
	0.4	7	262	0.53	0.95	221	3.1
	0.3	12	282	0.46	0.84	215	3.4
	0.2	20	304	0.29	0.83	204	3.8
	0.1	35	347	0.11	0.46	184	4.6
1.5	0.5	4	38	0.72	0.99	230	2.6
	0.4	7	39	0.65	0.93	224	2.9
	0.3	13	42	0.52	0.85	215	3.2
	0.2	21	56	0.33	0.64	204	3.6
	0.1	35	68	0.13	0.55	186	4.3
2	0.5	5	39	0.69	0.99	228	2.7
	0.4	8	39	0.62	0.94	223	2.9
	0.3	13	39	0.49	0.83	213	3.3
	0.2	21	52	0.33	0.72	203	3.7
	0.1	35	67	0.13	0.52	186	4.3

Table B.18: Urology Results for Total Objective and DVED Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	-2	250	0.74	0.98	234	2.6
	0.4	1	268	0.70	0.97	231	2.7
	0.3	8	265	0.57	0.83	220	3.1
	0.2	15	312	0.38	0.83	210	3.6
	0.1	30	362	0.15	0.38	190	4.3
1.5	0.5	1	39	0.77	0.97	233	2.6
	0.4	4	41	0.72	0.95	230	2.7
	0.3	11	43	0.58	0.83	219	3.1
	0.2	19	52	0.39	0.67	207	3.5
	0.1	32	74	0.17	0.53	190	4.2
2	0.5	3	37	0.73	0.96	230	2.6
	0.4	5	39	0.70	0.92	227	2.8
	0.3	11	40	0.58	0.89	218	3.1
	0.2	20	46	0.37	0.69	206	3.5
	0.1	34	69	0.15	0.62	188	4.2

Table B.19: Urology Results for Total Objective and GND Method

λ	v	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	3	291	0.65	0.96	228	2.9
	0.4	7	277	0.58	0.89	222	3.1
	0.3	12	287	0.45	0.83	214	3.4
	0.2	16	309	0.35	0.73	208	3.6
	0.1	21	319	0.24	0.59	200	3.9
1.5	0.5	3	39	0.76	0.97	230	2.6
	0.4	7	39	0.62	0.95	224	2.9
	0.3	13	42	0.53	0.83	216	3.2
	0.2	17	43	0.43	0.76	210	3.4
	0.1	23	58	0.29	0.66	200	3.8
2	0.5	4	39	0.72	0.96	229	2.7
	0.4	9	40	0.61	0.94	222	2.9
	0.3	13	39	0.52	0.89	214	3.2
	0.2	17	41	0.42	0.76	208	3.5
	0.1	23	53	0.29	0.66	200	3.8

Table B.20: Urology Results for Total Objective and SND Method

λ	v	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	2	266	0.67	0.96	229	2.8
	0.4	6	259	0.58	0.94	223	3.0
	0.3	11	275	0.53	0.96	217	3.2
	0.2	14	256	0.44	0.75	213	3.4
	0.1	19	301	0.31	0.72	203	3.7
1.5	0.5	3	39	0.75	0.96	230	2.6
	0.4	6	43	0.68	0.95	225	2.8
	0.3	11	40	0.57	0.88	218	3.1
	0.2	15	46	0.47	0.76	212	3.3
	0.1	21	56	0.35	0.58	204	3.6
2	0.5	4	39	0.73	0.98	229	2.7
	0.4	7	40	0.64	0.95	224	2.9
	0.3	12	40	0.56	0.91	217	3.1
	0.2	17	40	0.46	0.76	210	3.4
	0.1	22	52	0.34	0.58	203	3.7

Table B.21: Urology Results for CVaR Objective and DED Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	7	37	0.67	0.97	225	2.8
	0.4	10	38	0.60	0.96	220	3.0
	0.3	15	40	0.50	0.84	212	3.3
	0.2	24	50	0.29	0.72	200	3.8
	0.1	37	64	0.12	0.51	182	4.4
1.5	0.5	8	37	0.64	0.96	222	2.9
	0.4	12	39	0.57	0.96	217	3.1
	0.3	17	40	0.44	0.72	210	3.4
	0.2	25	51	0.28	0.74	199	3.8
	0.1	37	64	0.11	0.50	183	4.4
2	0.5	9	40	0.63	0.97	222	2.9
	0.4	12	41	0.55	0.92	216	3.1
	0.3	16	40	0.45	0.78	210	3.4
	0.2	24	50	0.29	0.77	200	3.8
	0.1	37	64	0.11	0.50	182	4.4

Table B.22: Urology Results for CVaR Objective and DVED Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	6	37	0.69	0.98	227	2.7
	0.4	10	38	0.61	0.93	221	3.0
	0.3	15	40	0.50	0.80	213	3.3
	0.2	22	46	0.32	0.70	203	3.7
	0.1	37	65	0.12	0.48	183	4.4
1.5	0.5	7	36	0.67	0.95	225	2.8
	0.4	10	38	0.60	0.92	220	3.0
	0.3	16	40	0.45	0.82	211	3.4
	0.2	22	47	0.32	0.75	202	3.7
	0.1	37	64	0.12	0.47	183	4.4
2	0.5	7	37	0.68	0.95	225	2.8
	0.4	11	38	0.59	0.92	220	3.0
	0.3	16	40	0.48	0.78	211	3.3
	0.2	22	47	0.36	0.77	205	3.6
	0.1	38	64	0.12	0.48	182	4.4

Table B.23: Urology Results for CVaR Objective and GND Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	6	37	0.67	0.96	225	2.8
	0.4	11	38	0.58	0.95	219	3.0
	0.3	13	39	0.50	0.90	213	3.2
	0.2	17	40	0.43	0.75	209	3.4
	0.1	25	51	0.28	0.61	199	3.8
1.5	0.5	6	37	0.69	0.96	226	2.8
	0.4	12	39	0.54	0.86	215	3.2
	0.3	15	40	0.48	0.81	212	3.3
	0.2	17	40	0.41	0.74	208	3.5
	0.1	27	52	0.25	0.69	196	3.9
2	0.5	8	37	0.63	0.94	221	2.9
	0.4	11	39	0.56	0.90	219	3.1
	0.3	16	40	0.47	0.85	211	3.3
	0.2	17	42	0.43	0.75	209	3.4
	0.1	26	53	0.27	0.67	198	3.8

Table B.24: Urology Results for CVaR Objective and SND Method

λ	υ	Mean Lateness	CVaR @ 0.95 Lateness	Mean Overtime Probability	CVaR @ 0.95 Overtime	Number Performed	Duration Waiting
1	0.5	7	37	0.67	0.98	225	2.8
	0.4	11	38	0.59	0.93	219	3.0
	0.3	14	39	0.49	0.89	212	3.3
	0.2	17	40	0.43	0.73	209	3.4
	0.1	24	50	0.30	0.60	200	3.8
1.5	0.5	7	37	0.66	0.95	224	2.8
	0.4	12	38	0.55	0.94	217	3.1
	0.3	16	40	0.46	0.82	212	3.3
	0.2	17	40	0.42	0.72	208	3.5
	0.1	24	49	0.29	0.62	200	3.8
2	0.5	8	37	0.64	0.97	222	2.9
	0.4	13	39	0.53	0.86	215	3.2
	0.3	16	40	0.47	0.77	210	3.4
	0.2	17	40	0.42	0.73	209	3.4
	0.1	24	50	0.30	0.59	201	3.7

— Appendix C —

Surgery Scheduling Lateness and Overtime Box Plots

This appendix includes box plots of both the lateness of operations and the overtime probability of sessions for Ophthalmology and Urology. There is one figure for each combination of: specialty, objective, and distribution (lateness of overtime). Within each figure there is a sub-plot for each combination of overtime probability estimation method, and λ/β.

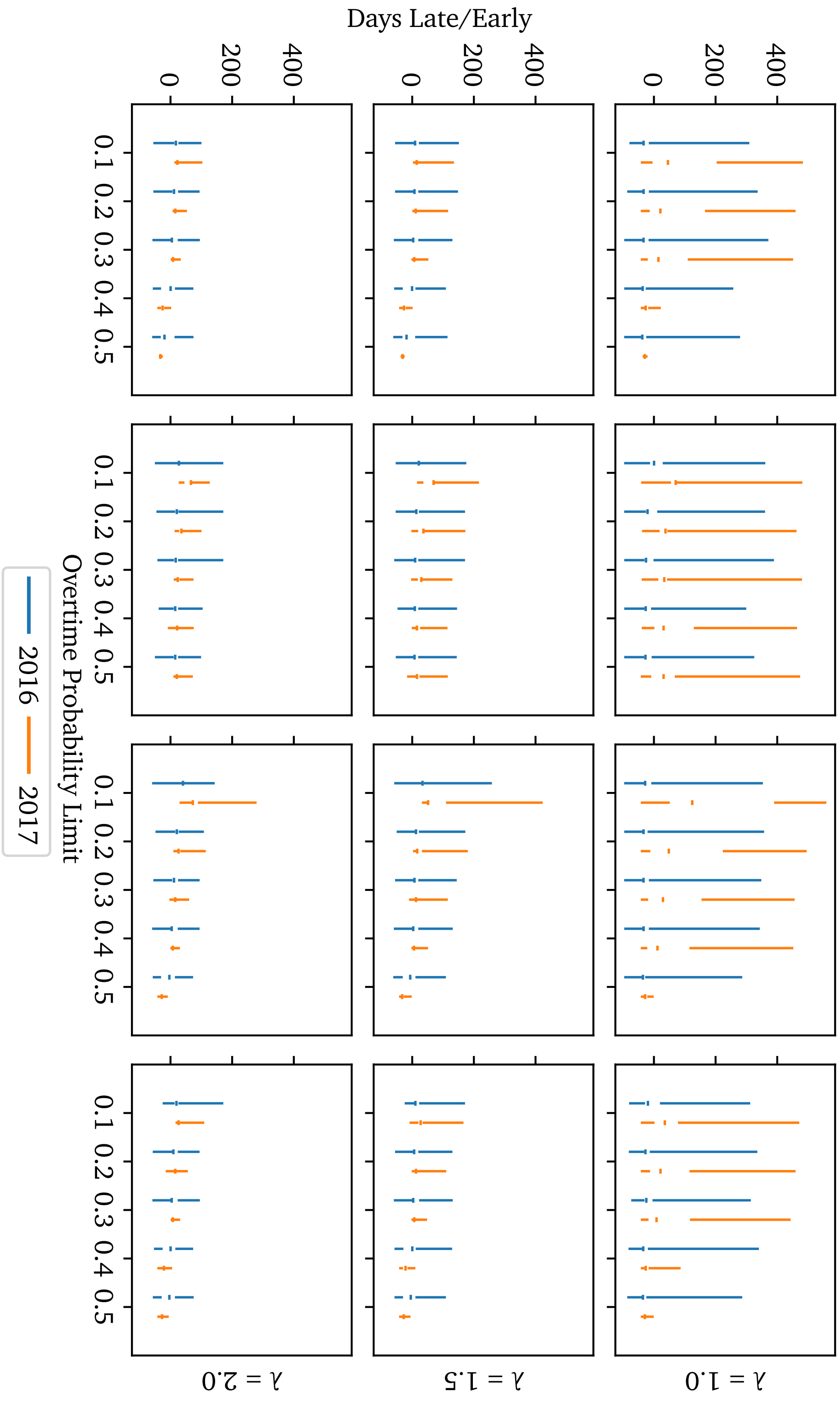


Figure C.1: Box Plots of Lateness for Ophthalmology with the Total Objective

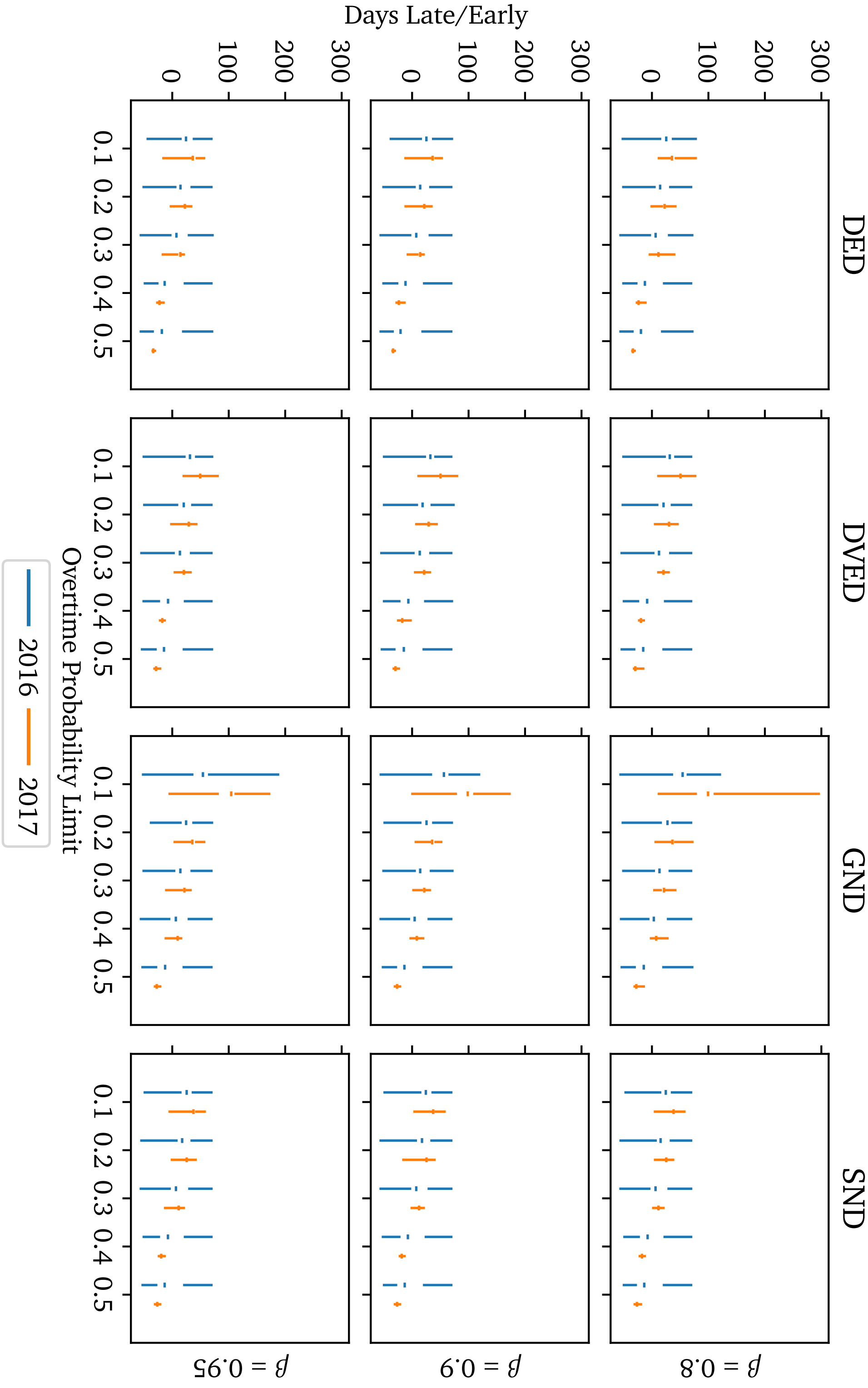


Figure C.2: Box Plots of Lateness for Ophthalmology with the CVaR Objective

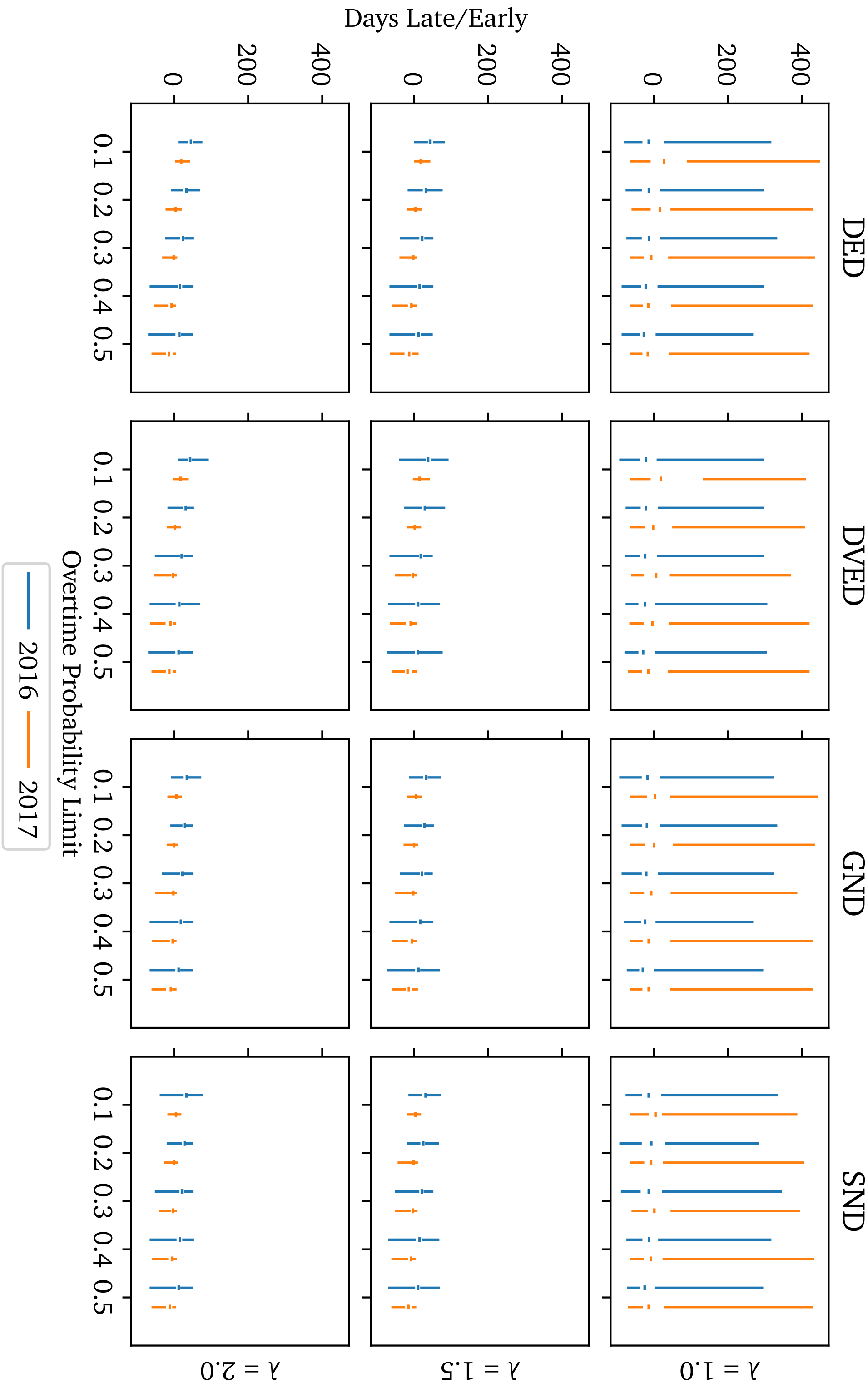


Figure C.3: Box Plots of Lateness for Urology with the Total Objective

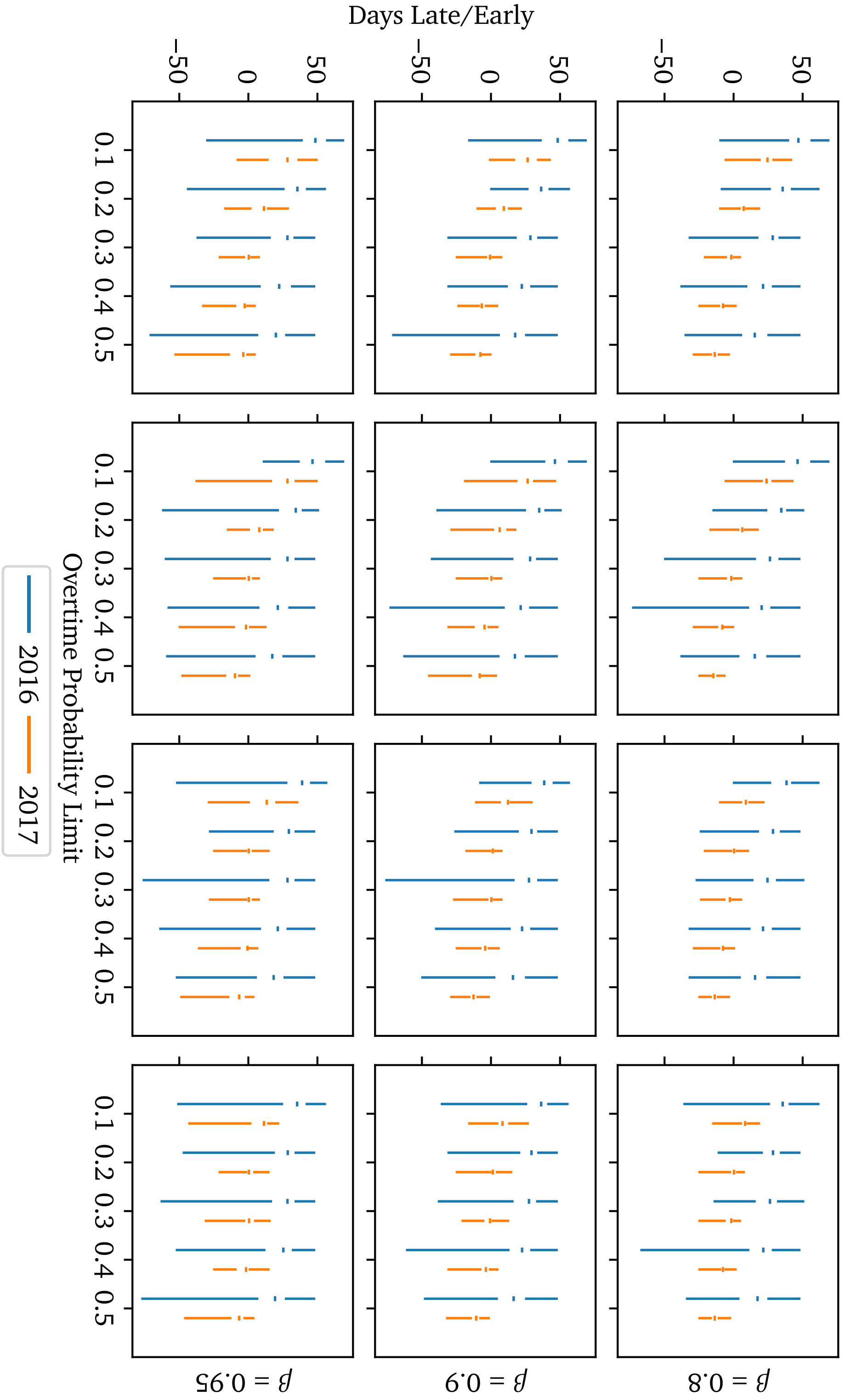


Figure C.4: Box Plots of Lateness for Urology with the CVaR Objective

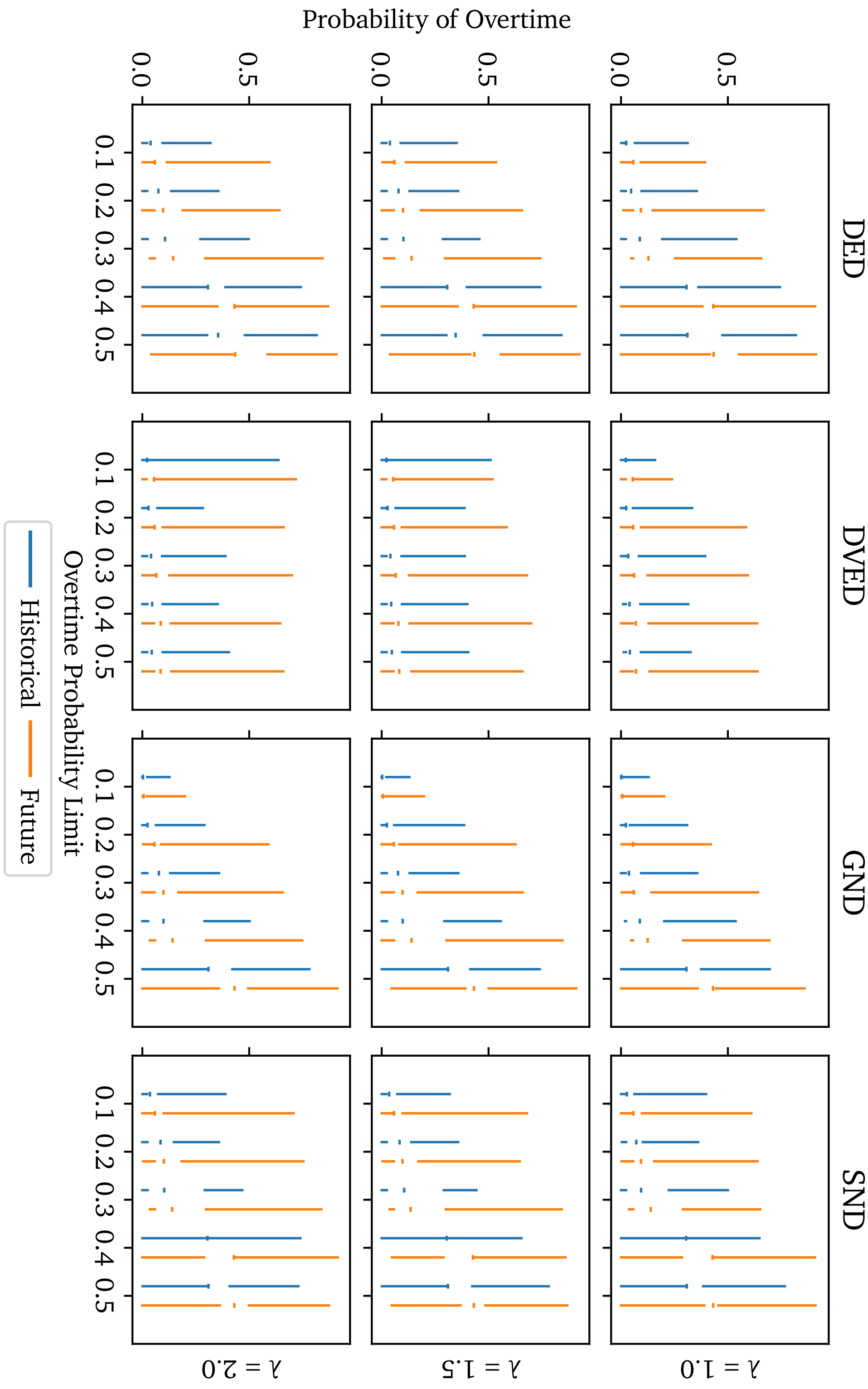


Figure C.5: Box Plots of Overtime Probability for Ophthalmology with the Total Objective

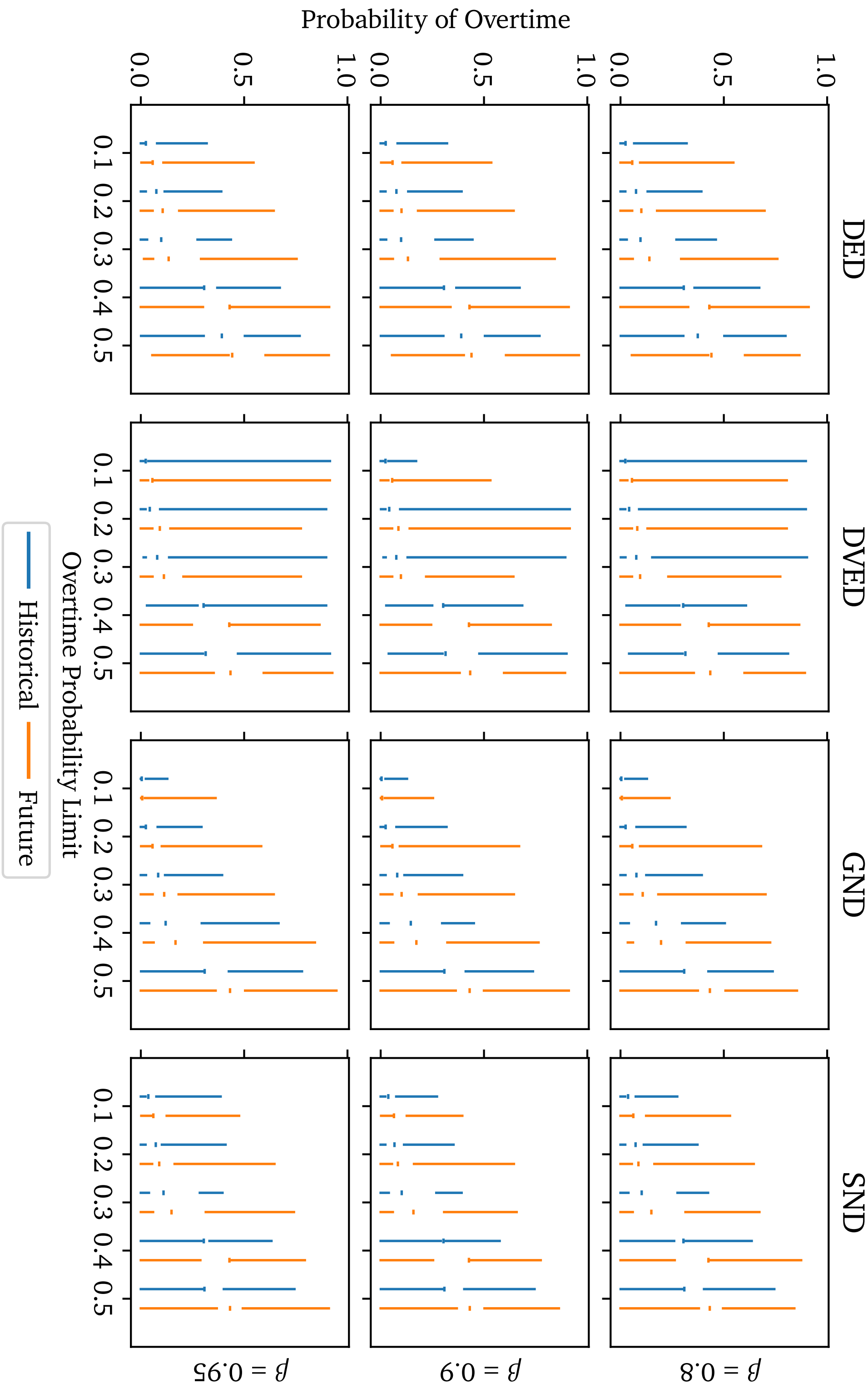


Figure C.6: Box Plots of Overtime Probability for Ophthalmology with the CVaR Objective

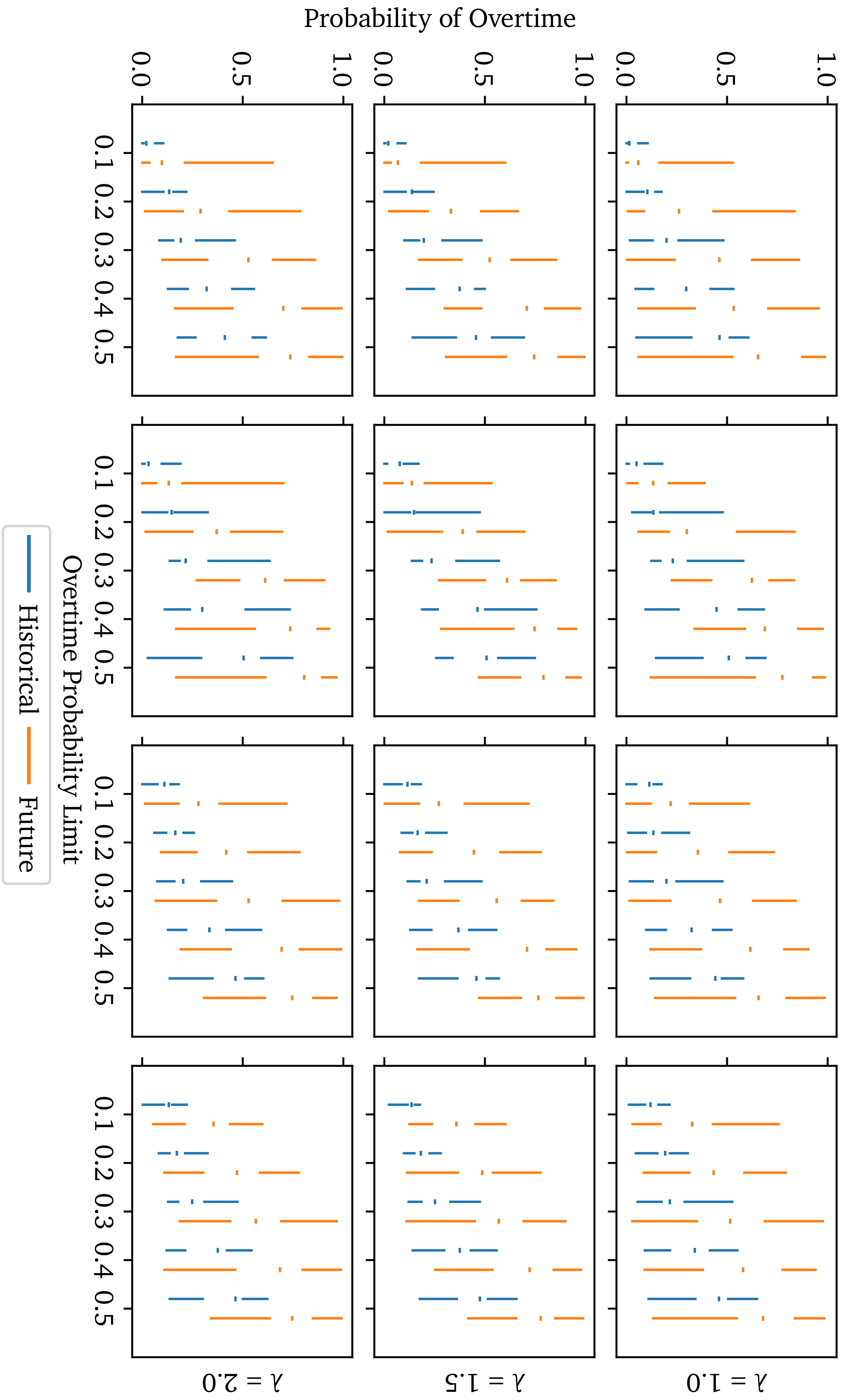


Figure C.7: Box Plots of Overtime Probability for Urology with the Total Objective

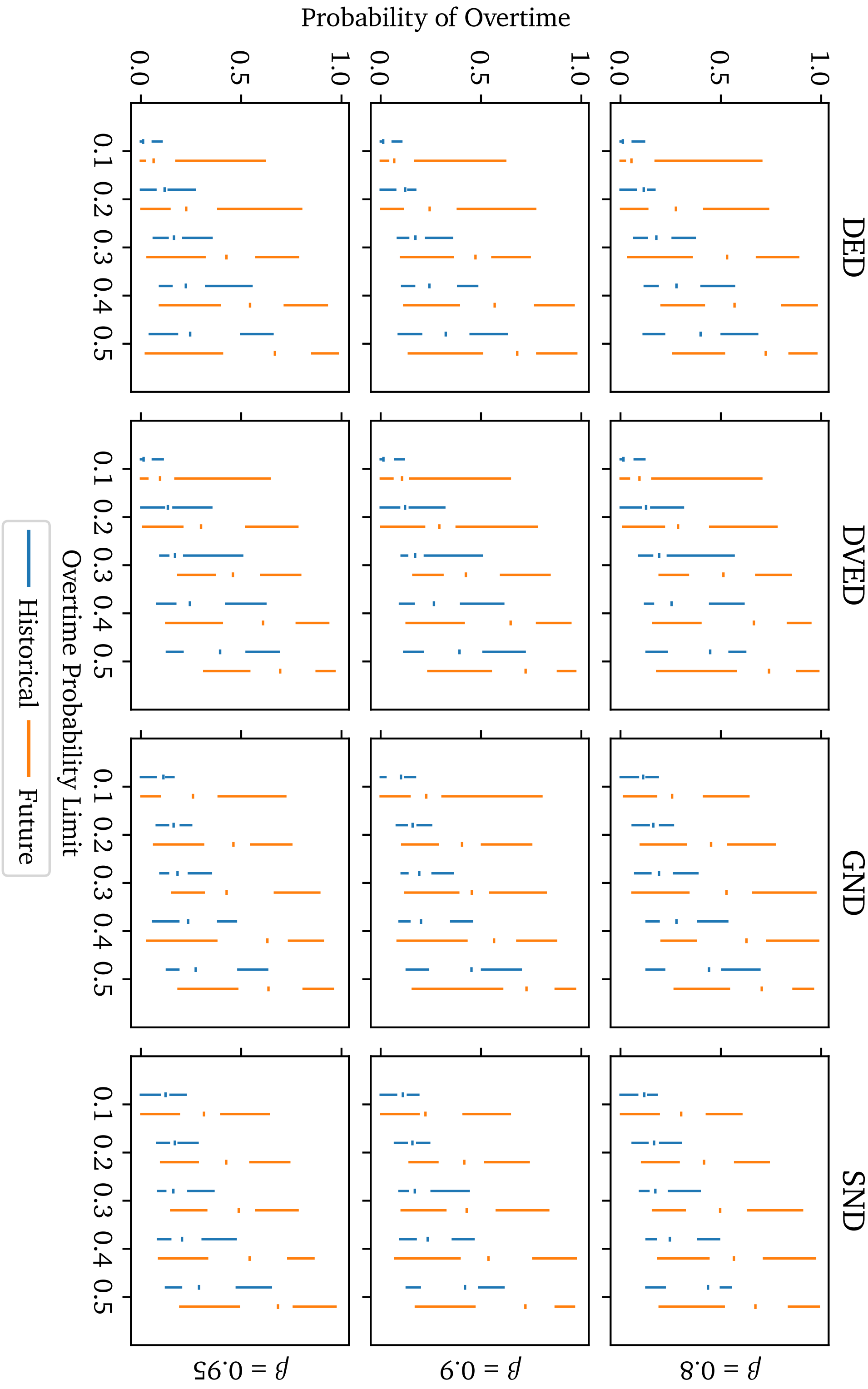


Figure C.8: Box Plots of Overtime Probability for Urology with the CVaR Objective

— Appendix D —

Surgery Scheduling Smoothed Scatter Plots

This appendix contains smoothed scatter plots of arrival date on the x-axis and lateness on the y-axis, for the Ophthalmology and Urology specialties. One figure is given for each combination of specialty and objective function. Within each figure there is a sub-plot for each type of overtime probability estimation method.

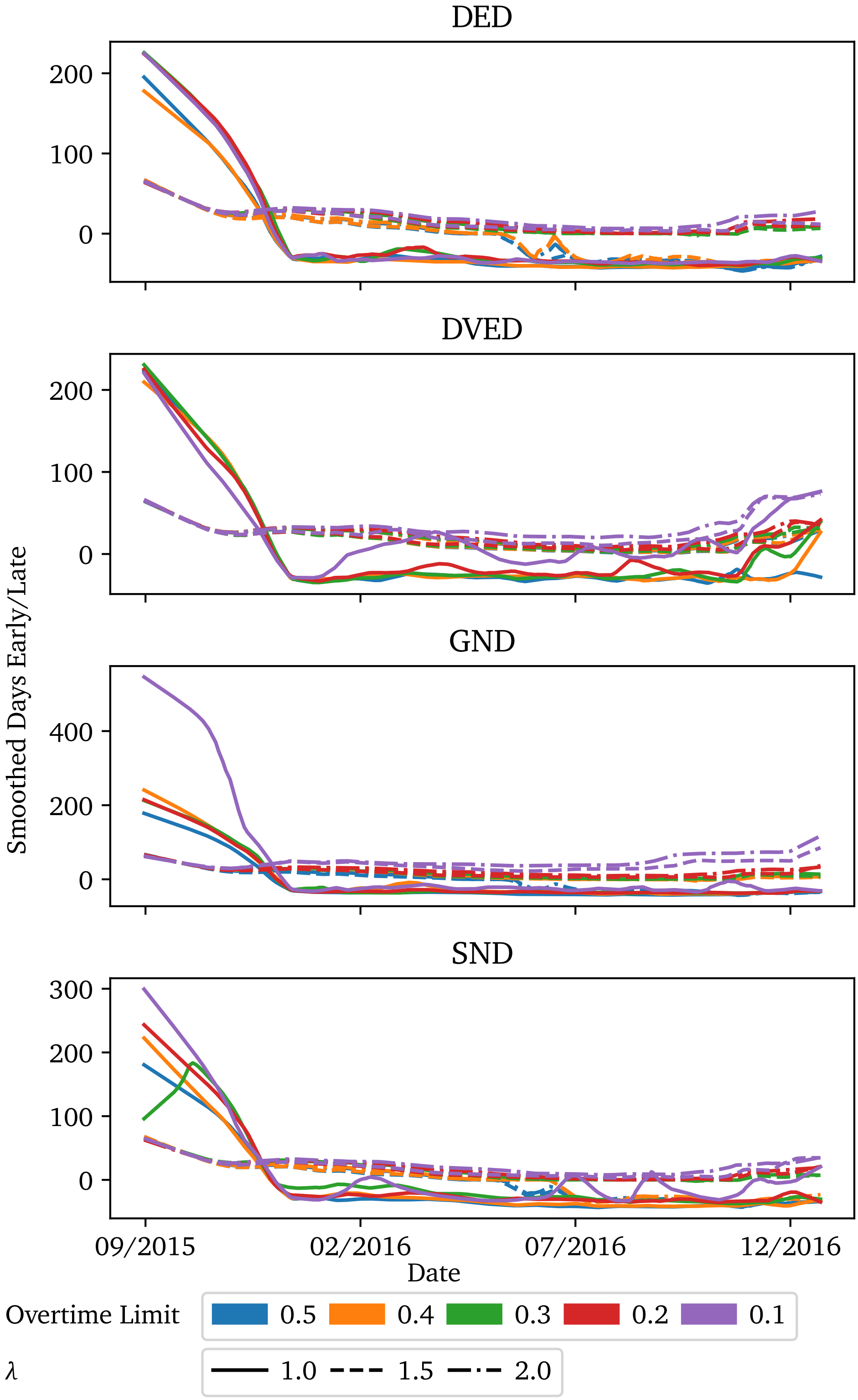


Figure D.1: Smoothed Scatter Plots of Arrival Date and Lateness for Ophthalmology and the Total Objective

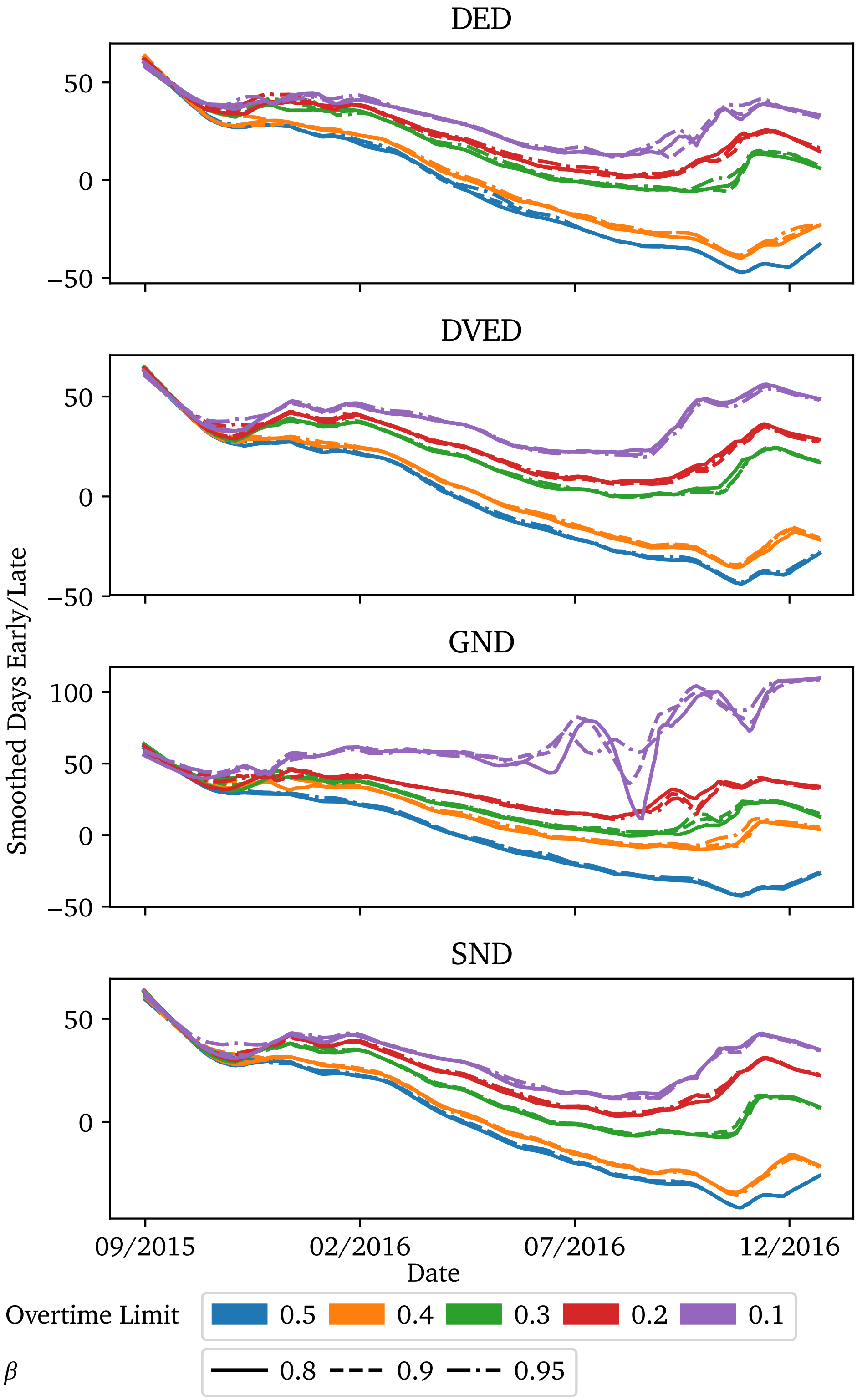


Figure D.2: Smoothed Scatter Plots of Arrival Date and Lateness for Ophthalmology and the CVaR Objective

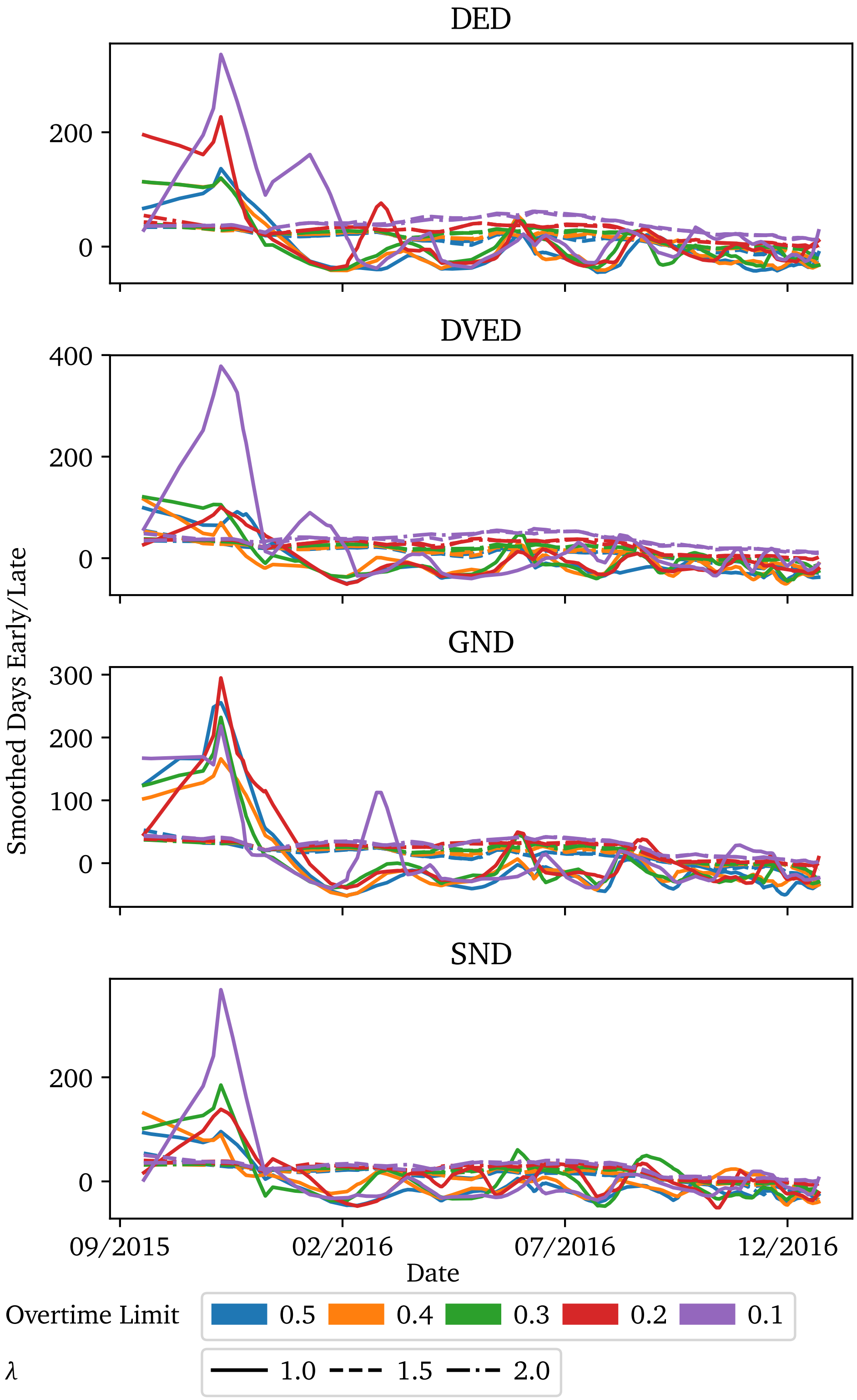


Figure D.3: Smoothed Scatter Plots of Arrival Date and Lateness for Urology and the Total Objective

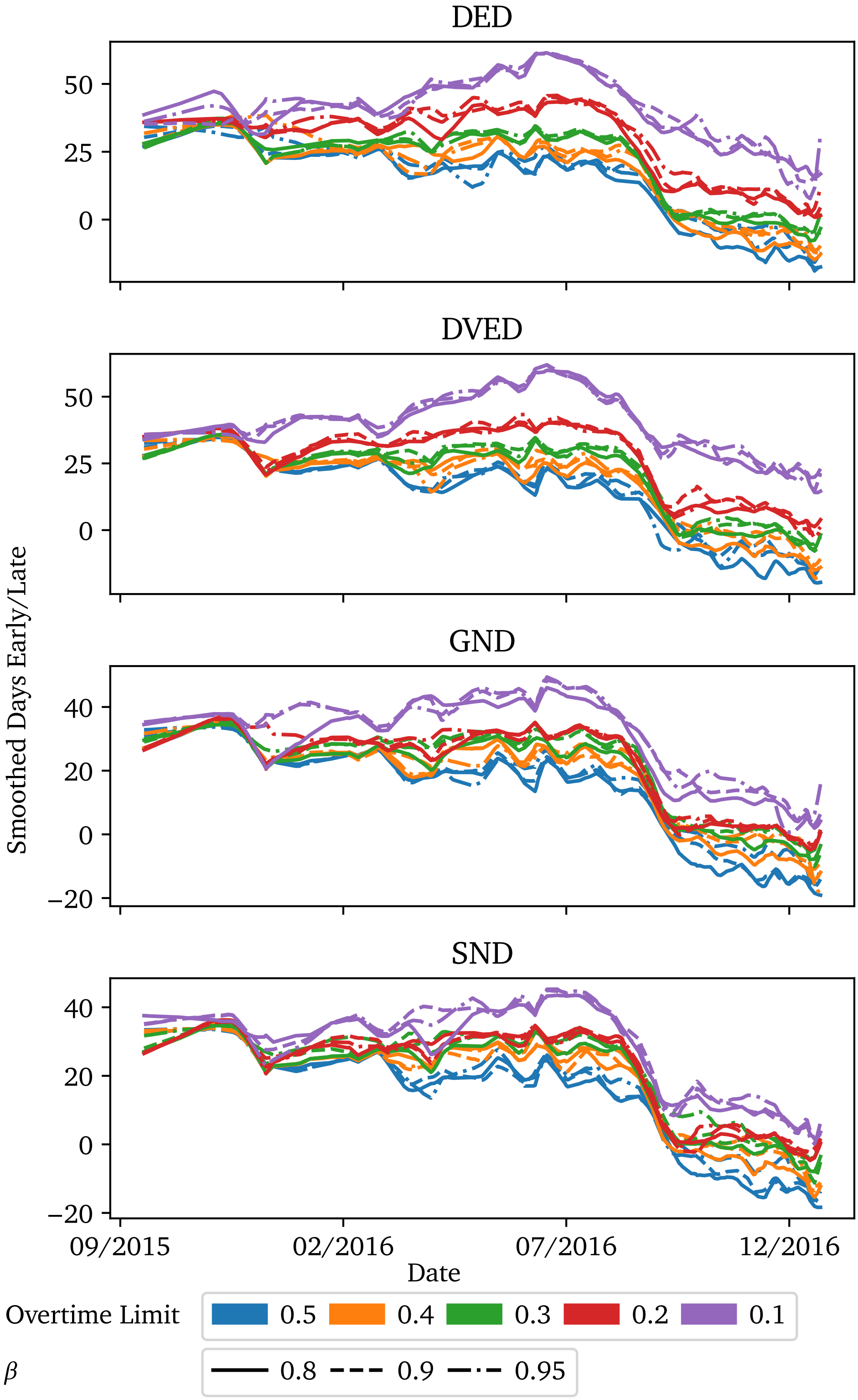


Figure D.4: Smoothed Scatter Plots of Arrival Date and Lateness for Urology and the CVaR Objective

— Appendix E —

Scatter Plots of Efficient Solutions

This appendix provides three two-dimensional scatter plots of the efficient solutions for both Ophthalmology and Urology. The objectives that the solutions are compared using to determine efficiency are: mean lateness, conditional value at risk (CVaR) at 95% lateness, and mean overtime probability.

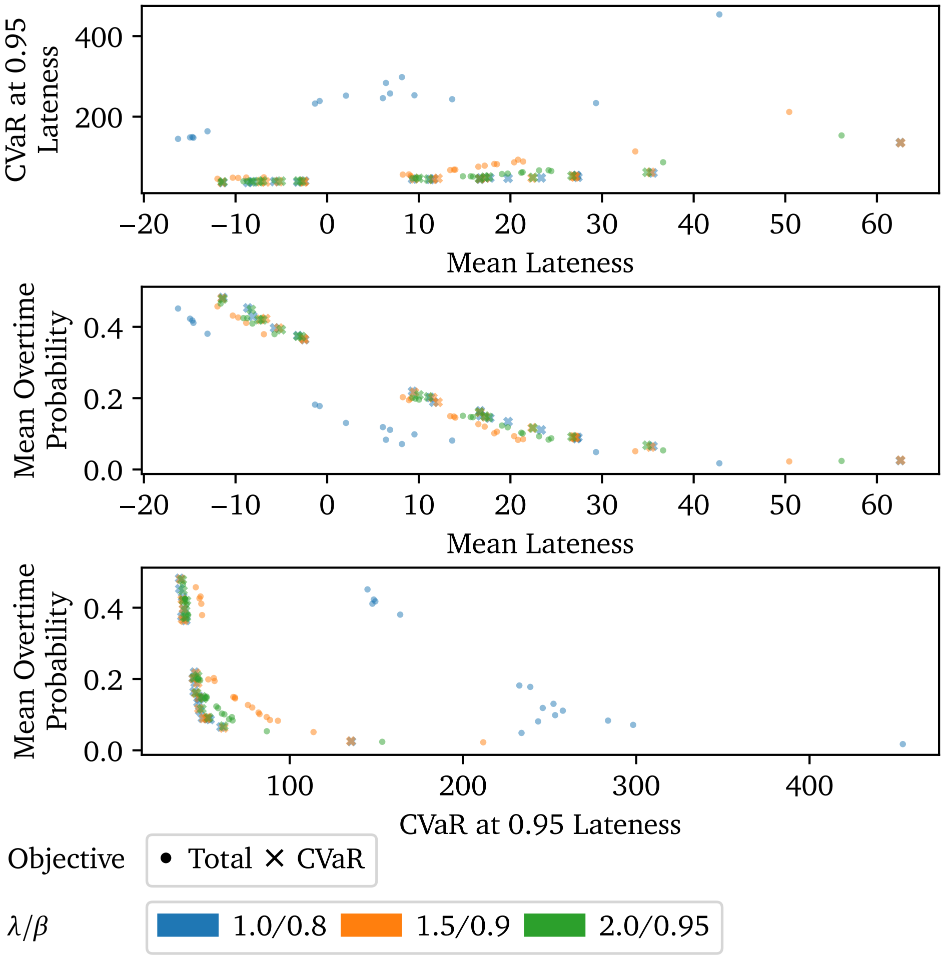


Figure E.1: Efficient Solutions for Ophthalmology

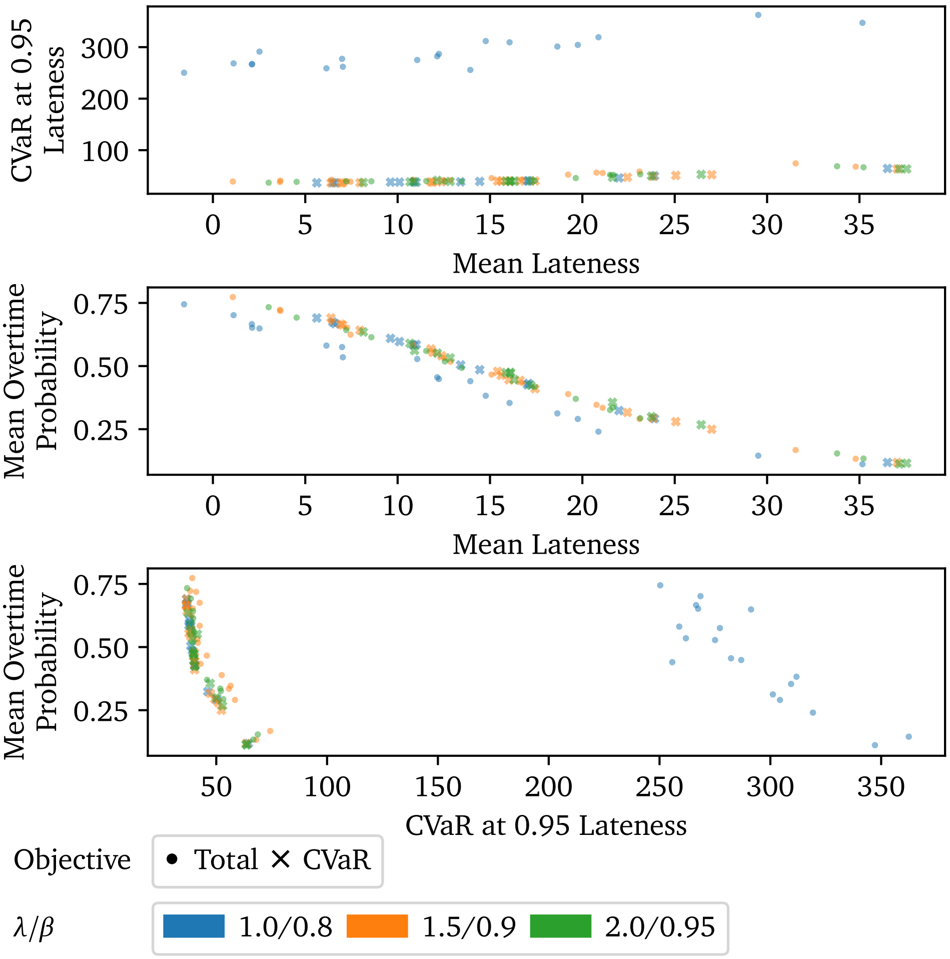


Figure E.2: Efficient Solutions for Urology

— Appendix F —

Surgery Scheduling Optimality Gaps

This appendix contains tables of the optimality and first fit heuristic gap results for both Ophthalmology and Urology. The optimality gap tables are separated by specialty and objective function, and show the mean values of the absolute, relative and uniform gaps (ϵ, η, ψ) over the 52 mixed integer programmes (MIPs), for each parameter and overtime probability estimation method combination. The first fit heuristic gap tables are also separated by specialty and objective function, and show the mean values of the absolute, relative and uniform gaps (ϵ, η, ψ) over the 52 MIPs and the five overtime limits, υ, for each objective parameter, λ/β.

Table F.1: Ophthalmology Optimality Gap Results for Total Objective

λ	υ	DED			DVED			GND			SND		
		ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ
1	0.5	244	0.03	0.03	663	0.09	0.09	216	0.03	0.03	229	0.03	0.03
	0.4	220	0.04	0.04	847	0.19	0.19	531	0.04	0.04	621	0.08	0.08
	0.3	600	0.04	0.04	2203	0.16	0.16	705	0.04	0.04	1802	0.11	0.11
	0.2	727	0.04	0.04	2951	0.15	0.15	745	0.04	0.04	2529	0.13	0.13
	0.1	678	0.04	0.04	3860	0.11	0.11	1937	0.04	0.04	3015	0.12	0.12
Mean		494	0.04	0.04	2105	0.14	0.14	827	0.04	0.04	1639	0.09	0.09
1.5	0.5	487	0.03	0.03	2221	0.15	0.15	551	0.03	0.03	542	0.03	0.03
	0.4	590	0.03	0.03	2918	0.16	0.16	797	0.03	0.03	1421	0.07	0.07
	0.3	984	0.04	0.04	8370	0.56	0.56	1319	0.04	0.04	5358	1.10	1.08
	0.2	1355	0.04	0.04	11655	0.38	0.38	1968	0.04	0.04	8297	0.30	0.30
	0.1	1946	0.04	0.04	23288	0.30	0.30	9587	0.04	0.04	12088	0.27	0.27
Mean		1072	0.04	0.04	9691	0.31	0.31	2844	0.04	0.04	5541	0.35	0.35
2	0.5	2150	0.03	0.03	12285	0.12	0.12	2441	0.03	0.03	2335	0.03	0.03
	0.4	2763	0.03	0.03	16088	0.25	0.25	4376	0.04	0.04	5900	0.07	0.07
	0.3	5747	0.04	0.04	53316	0.56	0.56	7234	0.05	0.05	27920	1.92	1.87
	0.2	7607	0.05	0.05	72551	0.43	0.43	12229	0.05	0.05	47749	0.35	0.35
	0.1	11693	0.05	0.05	157209	0.35	0.35	79802	0.04	0.04	77669	0.33	0.33
Mean		5992	0.04	0.04	62290	0.34	0.34	21217	0.04	0.04	32315	0.54	0.53
Grand Mean		2520	0.04	0.04	24695	0.27	0.27	8296	0.04	0.04	13165	0.33	0.32

Table F.2: Ophthalmology Optimality Gap Results for CVaR Objective

β	υ	DED			DVED			GND			SND		
		ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ
0.8	0.5	1.66	0.05	0.05	3.33	0.09	0.09	1.99	0.06	0.06	2.38	0.07	0.06
	0.4	2.50	0.07	0.07	4.45	0.15	0.14	4.87	0.48	0.37	3.45	0.12	0.11
	0.3	5.82	0.34	0.30	14.16	0.44	0.42	7.77	0.25	0.24	9.16	0.69	0.53
	0.2	8.21	0.25	0.24	17.15	0.38	0.37	13.23	0.25	0.25	13.59	0.36	0.35
	0.1	13.40	0.26	0.25	26.44	0.39	0.38	35.28	0.29	0.29	18.24	0.35	0.34
Mean		6.32	0.19	0.18	13.10	0.29	0.28	12.63	0.27	0.24	9.36	0.32	0.28
0.9	0.5	1.76	0.05	0.04	3.11	0.08	0.08	2.30	0.06	0.06	2.65	0.07	0.07
	0.4	2.53	0.07	0.06	4.54	0.15	0.14	5.50	0.43	0.34	3.60	0.12	0.11
	0.3	6.69	0.32	0.29	14.19	0.40	0.39	8.48	0.25	0.24	9.52	0.46	0.42
	0.2	8.99	0.26	0.25	17.31	0.37	0.37	14.74	0.27	0.27	14.17	0.34	0.33
	0.1	14.18	0.26	0.26	28.02	0.39	0.38	39.71	0.31	0.31	19.28	0.35	0.34
Mean		6.83	0.19	0.18	13.43	0.28	0.27	14.15	0.27	0.24	9.84	0.27	0.25
0.95	0.5	1.50	0.04	0.04	2.35	0.06	0.06	1.89	0.05	0.05	2.05	0.05	0.05
	0.4	2.10	0.06	0.05	3.64	0.11	0.11	5.25	0.26	0.23	3.07	0.08	0.08
	0.3	5.63	0.22	0.21	12.80	0.35	0.33	8.18	0.23	0.22	8.57	0.42	0.37
	0.2	8.36	0.22	0.22	16.04	0.33	0.32	14.01	0.25	0.25	14.04	0.32	0.31
	0.1	14.02	0.26	0.25	26.83	0.36	0.36	42.75	0.32	0.32	18.58	0.32	0.31
Mean		6.32	0.16	0.15	12.33	0.24	0.24	14.42	0.22	0.21	9.27	0.24	0.23
Grand Mean		6.49	0.18	0.17	12.96	0.27	0.26	13.73	0.25	0.23	9.49	0.27	0.25

Table F.3: Urology Optimality Gap Results for Total Objective

λ	υ	DED			DVED			GND			SND		
		ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ
1	0.5	79	0.03	0.03	79	0.04	0.04	91	0.03	0.03	77	0.03	0.03
	0.4	97	0.04	0.04	92	0.04	0.04	106	0.04	0.04	92	0.04	0.04
	0.3	120	0.04	0.04	111	0.04	0.04	120	0.04	0.04	121	0.05	0.05
	0.2	151	0.04	0.04	158	0.04	0.04	142	0.04	0.04	128	0.04	0.04
	0.1	230	0.04	0.04	270	0.05	0.05	171	0.04	0.04	176	0.05	0.05
Mean		135	0.04	0.04	142	0.04	0.04	126	0.04	0.04	119	0.04	0.04
1.5	0.5	123	0.04	0.04	143	0.06	0.06	126	0.04	0.04	130	0.04	0.04
	0.4	171	0.04	0.04	178	0.05	0.05	170	0.04	0.04	172	0.07	0.06
	0.3	227	0.04	0.04	239	0.05	0.05	217	0.04	0.04	260	0.06	0.06
	0.2	404	0.04	0.04	401	0.05	0.05	286	0.04	0.04	355	0.06	0.06
	0.1	857	0.04	0.04	1020	0.06	0.06	454	0.04	0.04	459	0.05	0.05
Mean		356	0.04	0.04	396	0.06	0.06	251	0.04	0.04	275	0.06	0.05
2	0.5	597	0.04	0.04	865	0.06	0.06	615	0.04	0.04	593	0.04	0.04
	0.4	817	0.04	0.04	881	0.05	0.05	841	0.04	0.04	872	0.04	0.04
	0.3	1156	0.04	0.04	1223	0.05	0.05	1002	0.04	0.04	1361	0.05	0.05
	0.2	2650	0.05	0.05	2466	0.05	0.05	1594	0.04	0.04	2382	0.09	0.09
	0.1	6032	0.04	0.04	8192	0.07	0.07	2826	0.05	0.05	3584	0.06	0.06
Mean		2250	0.04	0.04	2725	0.06	0.06	1376	0.04	0.04	1758	0.06	0.06
Grand Mean		914	0.04	0.04	1088	0.05	0.05	584	0.04	0.04	718	0.05	0.05

Table F.4: Urology Optimality Gap Results for CVaR Objective

β	υ	DED			DVED			GND			SND		
		ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ
0.8	0.5	1.11	0.04	0.04	1.13	0.04	0.04	1.08	0.04	0.04	1.09	0.04	0.04
	0.4	1.20	0.04	0.04	1.31	0.04	0.04	1.28	0.04	0.04	1.29	0.04	0.04
	0.3	1.45	0.05	0.04	1.43	0.04	0.04	1.33	0.04	0.04	1.33	0.04	0.04
	0.2	1.89	0.05	0.04	1.92	0.05	0.05	1.47	0.04	0.04	1.49	0.05	0.04
	0.1	2.45	0.04	0.04	4.52	0.08	0.08	2.00	0.05	0.05	2.45	0.06	0.06
Mean		1.62	0.04	0.04	2.06	0.05	0.05	1.44	0.04	0.04	1.53	0.05	0.04
0.9	0.5	0.85	0.03	0.03	0.96	0.03	0.03	0.82	0.03	0.03	0.90	0.03	0.03
	0.4	1.14	0.03	0.03	1.15	0.03	0.03	1.18	0.04	0.03	1.16	0.04	0.03
	0.3	1.43	0.04	0.04	1.30	0.04	0.04	1.36	0.04	0.04	1.31	0.04	0.04
	0.2	1.94	0.04	0.04	1.94	0.05	0.04	1.47	0.04	0.04	1.31	0.04	0.04
	0.1	2.37	0.04	0.04	4.12	0.07	0.07	1.84	0.04	0.04	2.37	0.05	0.05
Mean		1.55	0.04	0.04	1.89	0.04	0.04	1.33	0.04	0.04	1.41	0.04	0.04
0.95	0.5	0.62	0.02	0.02	0.71	0.02	0.02	0.66	0.02	0.02	0.78	0.03	0.02
	0.4	0.74	0.02	0.02	1.08	0.03	0.03	0.86	0.03	0.02	0.96	0.03	0.03
	0.3	0.95	0.03	0.03	1.19	0.03	0.03	0.92	0.03	0.02	0.96	0.03	0.03
	0.2	1.58	0.03	0.03	1.67	0.04	0.04	1.02	0.03	0.03	1.12	0.03	0.03
	0.1	2.14	0.04	0.03	3.09	0.05	0.05	1.61	0.03	0.03	2.55	0.05	0.05
Mean		1.21	0.03	0.03	1.55	0.03	0.03	1.01	0.03	0.03	1.27	0.03	0.03
Grand Mean		1.46	0.04	0.03	1.83	0.04	0.04	1.26	0.04	0.03	1.40	0.04	0.04

Table F.5: Ophthalmology First Fit Heuristic Gap Results for Total Objective

λ	DED			DVED			GND			SND		
	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ
1	658.42	0.08	0.08	384.06	0.05	0.04	855.87	0.18	0.17	378.89	0.06	0.06
1.5	611.64	0.06	0.06	559.84	0.03	0.03	1187.35	0.06	0.06	550.42	0.06	0.06
2	2594.66	0.04	0.04	2948.61	0.04	0.04	5355.58	0.06	0.05	2677.63	0.07	0.07
Mean	1288.24	0.06	0.06	1297.50	0.04	0.04	2466.27	0.10	0.09	1202.31	0.06	0.06

Table F.6: Ophthalmology First Fit Heuristic Gap Results for CVaR Objective

β	DED			DVED			GND			SND		
	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ
0.8	0.71	0.03	0.03	0.25	0.01	0.01	0.64	0.11	0.04	0.24	0.05	0.01
0.9	0.70	0.03	0.03	0.28	0.02	0.01	0.65	0.03	0.02	0.32	0.02	0.02
0.95	0.77	0.03	0.03	0.36	0.04	0.02	0.58	0.06	0.03	0.41	0.02	0.02
Mean	0.73	0.03	0.03	0.30	0.02	0.01	0.62	0.06	0.03	0.33	0.03	0.02

Table F.7: Urology First Fit Heuristic Gap Results for Total Objective

λ	DED			DVED			GND			SND		
	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ
1	22.02	0.01	0.01	30.92	0.01	0.01	22.05	0.01	0.01	19.33	0.01	0.01
1.5	96.11	0.03	0.03	107.12	0.04	0.04	90.33	0.05	0.05	86.54	0.08	0.08
2	811.34	0.05	0.05	809.59	0.06	0.06	696.03	0.09	0.09	687.60	0.06	0.06
Mean	309.82	0.03	0.03	315.87	0.04	0.04	269.47	0.05	0.05	264.49	0.05	0.05

Table F.8: Urology First Fit Heuristic Gap Results for CVaR Objective

β	DED			DVED			GND			SND		
	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ	ϵ	η	ψ
0.8	1.05	0.10	0.06	1.05	0.21	0.07	0.97	0.65	0.07	1.03	0.08	0.06
0.9	1.49	0.06	0.05	1.50	0.05	0.04	1.34	0.07	0.06	1.55	0.07	0.06
0.95	2.28	0.05	0.05	2.33	0.06	0.06	1.78	0.05	0.05	2.27	0.06	0.06
Mean	1.60	0.07	0.05	1.63	0.11	0.06	1.36	0.26	0.06	1.62	0.07	0.06

References

Adams, T., M. O'Sullivan, and C. Walker (2019). "Physician rostering for workload balance". In: *Operations Research for Health Care* 20, pp. 1–10.

Addis, B., G. Carello, A. Grosso, and E. Tànfani (2016). "Operating room scheduling and rescheduling: a rolling horizon approach". In: *Flexible Services and Manufacturing Journal* 28.1, pp. 206–232.

Bailey, J. (1985). "Integrated days off and shift personnel scheduling". In: *Computers & Industrial Engineering* 9.4, pp. 395–404.

Balakrishnan, N. and R. T. Wong (1990). "A network model for the rotating workforce scheduling problem". In: *Networks* 20.1, pp. 25–42.

Ballestín, F., Á. Pérez, and S. Quintanilla (2018). "Scheduling and rescheduling elective patients in operating rooms to minimise the percentage of tardy patients". In: *Journal of Scheduling*.

Bard, J. F., Z. Shu, and L. Leykum (2014). "A network-based approach for monthly scheduling of residents in primary care clinics". In: *Operations Research for Health Care* 3.4, pp. 200–214.

Barrera, J., R. A. Carrasco, S. Mondschein, G. Canessa, and D. Rojas-Zalazar (2018). "Operating room scheduling under waiting time constraints: the Chilean GES plan". In: *Annals of Operations Research*.

Beale, E. and J. Tomlin (1969). "Special facilities in a general mathematical programming system for nonconvex problems using ordered sets of variables". In: *Operational Research* 69, pp. 447–454.

Beaulieu, H., J. A. Ferland, B. Gendron, and P. Michelon (2000). "A mathematical programming approach for scheduling physicians in the emergency room". In: *Health care management science* 3.3, pp. 193–200.

Benders, J. F. (1962). "Partitioning procedures for solving mixed-variables programming problems". In: *Numerische mathematik* 4.1, pp. 238–252.

Birge, J. R. and F. V. Louveaux (1988). “A multicut algorithm for two-stage stochastic linear programs”. In: *European Journal of Operational Research* 34.3, pp. 384–392.

Board, C. M. D. H. (2015). *Our district*. URL: `https://www.countiesmanukau.health.nz/about-us/our-district/` (visited on 08/16/2019).

Bruni, R. and P. Detti (2014). “A flexible discrete optimization approach to the physician scheduling problem”. In: *Operations Research for Health Care* 3.4, pp. 191–199.

Brunner, J. O. and G. M. Edenharter (2011). “Long term staff scheduling of physicians with different experience levels in hospitals using column generation”. In: *Health Care Management Science* 14.2, pp. 189–202.

Burke, E. K. and T. Curtois (2014). “New approaches to nurse rostering benchmark instances”. In: *European Journal of Operational Research* 237.1, pp. 71–81.

Cardoen, B., E. Demeulemeester, and J. Beliën (2009a). “Optimizing a multiple objective surgical case sequencing problem”. In: *International Journal of Production Economics* 119.2, pp. 354–366.

— (2009b). “Sequencing surgical cases in a day-care environment: An exact branch-and-price approach”. In: *Computers & Operations Research* 36.9, pp. 2660–2669.

— (2010). “Operating room planning and scheduling: A literature review”. In: *European Journal of Operational Research* 201.3, pp. 921–932.

De Causmaecker, P. and G. Vanden Berghe (2011). “A categorisation of nurse rostering problems”. In: *Journal of Scheduling* 14.1, pp. 3–16.

Denton, B. T., A. J. Miller, H. J. Balasubramanian, and T. R. Huschka (2010). “Optimal Allocation of Surgery Blocks to Operating Rooms Under Uncertainty”. In: *Operations Research* 58.4, pp. 802–816.

Denton, B. T., J. Viapiano, and A. Vogl (2007). “Optimization of surgery sequencing and scheduling decisions under uncertainty”. In: *Health Care Management Science* 10.1, pp. 13–24.

Epstein, K., E. Juarez, A. Epstein, K. Loya, and A. Singer (2010). “The impact of fragmentation of hospitalist care on length of stay”. In: *Journal of Hospital Medicine* 5.6, pp. 335–338.

Ernst, A., H. Jiang, M. Krishnamoorthy, and D. Sier (2004). “Staff scheduling and rostering: A review of applications, methods and models”. In: *European Journal of Operational Research* 153.1, pp. 3–27.

Fei, H., C. Chu, and N. Meskens (2009). “Solving a tactical operating room planning problem by a column-generation-based heuristic procedure with four criteria”. In: *Annals of Operations Research* 166.1, pp. 91–108.

Fei, H., C. Chu, N. Meskens, and A. Artiba (2008). “Solving surgical cases assignment problem by a branch-and-price approach”. In: *International Journal of Production Economics* 112.1, pp. 96–108.

Ferrand, Y., M. Magazine, U. S. Rao, and T. F. Glass (2011). “Building Cyclic Schedules for Emergency Department Physicians”. In: *Interfaces* 41.6, pp. 521–533.

Fischetti, M., D. Salvagnin, and A. Zanette (2010). “A note on the selection of Benders’ cuts”. In: *Mathematical Programming* 124.1, pp. 175–182.

Fortz, B. and M. Poss (2009). “An improved Benders decomposition applied to a multi-layer network design problem”. In: *Operations Research Letters* 37.5, pp. 359–364.

Ganguly, S., S. Lawrence, and M. Prather (2014). “Emergency Department Staff Planning to Improve Patient Care and Reduce Costs”. In: *Decision Sciences* 45.1, pp. 115–145.

Hans, E. W., G. Wullink, M. van Houdenhoven, and G. Kazemier (2008). “Robust surgery loading”. In: *European Journal of Operational Research* 185.3, pp. 1038–1050.

Hjortdahl, P. and E. Laerum (1992). “Continuity of care in general practice: effect on patient satisfaction.” In: *BMJ* 304.6837, pp. 1287–1290.

Jebali, A. and A. Diabat (2015). “A stochastic model for operating room planning under capacity constraints”. In: *International Journal of Production Research* 53.24, pp. 7252–7270.

Kamran, M. A., B. Karimi, and N. Dellaert (2018). “Uncertainty in advance scheduling problem in operating room planning”. In: *Computers & Industrial Engineering* 126, pp. 252–268.

Kazemian, P., Y. Dong, T. R. Rohleder, J. E. Helm, and M. P. Van Oyen (2014). “An IP-based healthcare provider shift design approach to minimize patient handoffs”. In: *Health Care Management Science* 17.1, pp. 1–14.

Kim, K. and S. Mehrotra (2015). “A Two-Stage Stochastic Integer Programming Approach to Integrated Staffing and Scheduling with Application to Nurse Management”. In: *Operations Research* 63.6, pp. 1431–1451.

Kleywegt, A. J., A. Shapiro, and T. Homem-de-Mello (2002). “The sample average approximation method for stochastic discrete optimization”. In: *SIAM Journal on Optimization* 12.2, pp. 479–502.

Kortbeek, N., A. Braaksma, C. Burger, P. Bakker, and R. J. Boucherie (2015). "Flexible nurse staffing based on hourly bed census predictions". In: *International Journal of Production Economics* 161, pp. 167–180.

Kroer, L. R., K. Foverskov, C. Vilhelmsen, A. S. Hansen, and J. Larsen (2018). "Planning and scheduling operating rooms for elective and emergency surgeries with uncertain duration". In: *Operations Research for Health Care*.

Lamiri, M., J. Dreo, and X. Xie (2007). "Operating Room Planning with Random Surgery Times". In: *2007 IEEE International Conference on Automation Science and Engineering*. 2007 IEEE International Conference on Automation Science and Engineering. Scottsdale, AZ, USA: IEEE, pp. 521–526.

Lamiri, M., F. Grimaud, and X. Xie (2009). "Optimization methods for a stochastic surgery planning problem". In: *International Journal of Production Economics* 120.2, pp. 400–410.

Landa, P., R. Aringhieri, P. Soriano, E. Tànfani, and A. Testi (2016). "A hybrid optimization algorithm for surgeries scheduling". In: *Operations Research for Health Care* 8, pp. 103–114.

Lanzarone, E. and A. Matta (2014). "Robust nurse-to-patient assignment in home care services to minimize overtimes under continuity of care". In: *Operations Research for Health Care* 3.2, pp. 48–58.

Li, Q. and J. S. Racine (2006). *Nonparametric Econometrics: Theory and Practice*. Economics Books 8355. Princeton University Press.

Liu, R. and L. Yang (2008). "Kernel estimation of multivariate cumulative distribution function". In: *Journal of Nonparametric Statistics* 20, pp. 661–677.

MacPherson, L. (2016). *National Population Projections: 2016(base)-2068*. Statistics New Zealand.

Magnanti, T. L. and R. T. Wong (1981). "Accelerating Benders Decomposition: Algorithmic Enhancement and Model Selection Criteria". In: *Operations Research* 29.3, pp. 464–484.

Marques, I. and M. E. Captivo (2017). "Different stakeholders' perspectives for a surgical case assignment problem: Deterministic and robust approaches". In: *European Journal of Operational Research* 261.1, pp. 260–278.

Marques, I., M. E. Captivo, and M. Vaz Pato (2012). "An integer programming approach to elective surgery scheduling: Analysis and comparison based on a real case". In: *OR Spectrum* 34.2, pp. 407–427.

— (2015). "A bicriteria heuristic for an elective surgery scheduling problem". In: *Health Care Management Science* 18.3, pp. 251–266.

Millar, H. H. and M. Kiragu (1998). "Cyclic and non-cyclic scheduling of 12 h shift nurses by network programming". In: *European Journal of Operational Research* 104.3, pp. 582–592.

Min, D. and Y. Yih (2010). "An elective surgery scheduling problem considering patient priority". In: *Computers & Operations Research* 37.6, pp. 1091–1099.

Molina-Pariente, J. M., E. W. Hans, and J. M. Framinan (2018). "A stochastic approach for solving the operating room scheduling problem". In: *Flexible Services and Manufacturing Journal* 30.1, pp. 224–251.

Naoum-Sawaya, J. and S. Elhedhli (2013). "An interior-point Benders based branch-and-cut algorithm for mixed integer programs". In: *Annals of Operations Research* 210.1, pp. 33–55.

Naudin, E., P. Y. C. Chan, M. Hiroux, T. Zemmouri, and G. Weil (2012). "Analysis of three mathematical models of the Staff Rostering Problem". In: *Journal of Scheduling* 15.1, pp. 23–38.

Oostrum, J. M. van, M. Van Houdenhoven, J. L. Hurink, E. W. Hans, G. Wullink, and G. Kazemier (2008). "A master surgical scheduling approach for cyclic scheduling in operating room departments". In: *OR Spectrum* 30.2, pp. 355–374.

Papadakos, N. (2008). "Practical enhancements to the Magnanti–Wong method". In: *Operations Research Letters* 36.4, pp. 444–449.

Pulido, R., A. M. Aguirre, M. Ortega-Mier, Á. García-Sánchez, and C. A. Méndez (2014). "Managing daily surgery schedules in a teaching hospital: a mixed-integer optimization approach". In: *BMC Health Services Research* 14.1.

Rachuba, S. and B. Werners (2014). "A robust approach for scheduling in hospitals using multiple objectives". In: *Journal of the Operational Research Society* 65.4, pp. 546–556.

Rahmaniani, R., T. G. Crainic, M. Gendreau, and W. Rei (2017). "The Benders decomposition algorithm: A literature review". In: *European Journal of Operational Research* 259.3, pp. 801–817.

El-Rifai, O., T. Garaix, V. Augusto, and X. Xie (2015). "A stochastic optimization model for shift scheduling in emergency departments". In: *Health Care Management Science* 18.3, pp. 289–302.

Rocha, M., J. F. Oliveira, and M. A. Carravilla (2013). "Cyclic staff scheduling: optimization models for some real-life problems". In: *Journal of Scheduling* 16.2, pp. 231–242.

Rockafellar, R. T. and S. Uryasev (2000). "Optimization of conditional value-at-risk". In: *The Journal of Risk* 2.3, pp. 21–41.

Sagnol, G., C. Barner, R. Borndörfer, M. Grima, M. Seeling, C. Spies, and K. Wernecke (2018). “Robust allocation of operating rooms: A cutting plane approach to handle lognormal case durations”. In: *European Journal of Operational Research* 271.2, pp. 420–435.

Santos, H. G., T. A. M. Toffolo, R. A. M. Gomes, and S. Ribas (2014). “Integer programming techniques for the nurse rostering problem”. In: *Annals of Operations Research*.

Shylo, O. V., O. A. Prokopyev, and A. J. Schaefer (2013). “Stochastic Operating Room Scheduling for High-Volume Specialties Under Block Booking”. In: *INFORMS Journal on Computing* 25.4, pp. 682–692.

Smalley, H. K., P. Keskinocak, and A. Vats (2015). “Physician Scheduling for Continuity: An Application in Pediatric Intensive Care”. In: *Interfaces* 45.2, pp. 133–148.

Stolletz, R. and J. O. Brunner (2012). “Fair optimization of fortnightly physician schedules with flexible shifts”. In: *European Journal of Operational Research* 219.3, pp. 622–629.

Strum M.D., D. P., J. H. May Ph.D, and L. G. Vargas Ph.D. (2000). “Modeling the Uncertainty of Surgical Procedure Times: Comparison of Log-normal and Normal Models”. In: *Anesthesiology: The Journal of the American Society of Anesthesiologists* 92.4, pp. 1160–1167.

The New Zealand Ministry of Health (2011). *Targeting more elective operations: improved access to elective surgery.* OCLC: 721312094. Wellington, N.Z.: Ministry of Health.

— (2018a). *Funding*. URL: https://www.health.govt.nz/new-zealand-health-system/overview-health-system/funding (visited on 10/09/2018).

— (2018b). *Overview of the Health System*. URL: https://www.health.govt.nz/new-zealand-health-system/overview-health-system (visited on 10/09/2018).

— (2018c). *The structure of the New Zealand health and disability sector*. URL: https://www.health.govt.nz/sites/default/files/images/nz-health-system/structure-nz-health-disability-sector-oct16.png (visited on 10/09/2018).

— (2019a). *Planned Care services*. URL: https://www.health.govt.nz/our-work/hospitals-and-specialist-care/planned-care-services (visited on 11/01/2019).

— (2019b). *Population of Counties Manukau DHB*. URL: https://www.health.govt.nz/new-zealand-health-system/my-dhb/counties-manukau-dhb/population-counties-manukau-dhb (visited on 08/16/2019).

Vali-Siar, M. M., S. Gholami, and R. Ramezanian (2018). “Multi-period and multi-resource operating room scheduling under uncertainty: A case study”. In: *Computers & Industrial Engineering* 126, pp. 549–568.

Van den Bergh, J., J. Beliën, P. De Bruecker, E. Demeulemeester, and L. De Boeck (2013). “Personnel scheduling: A literature review”. In: *European Journal of Operational Research* 226.3, pp. 367–385.

Van Huele, C. and M. Vanhoucke (2014). “Analysis of the Integration of the Physician Rostering Problem and the Surgery Scheduling Problem”. In: *Journal of Medical Systems* 38.6.

Vielma, J. P., S. Ahmed, and G. L. Nemhauser (2010). “Mixed-Integer Models for Nonseparable Piecewise-Linear Optimization: Unifying Framework and Extensions”. In: *Operations Research*, p. 14.

Wang, J., H. Guo, M. Bakker, and K.-L. Tsui (2018). “An integrated approach for surgery scheduling under uncertainty”. In: *Computers & Industrial Engineering* 118, pp. 1–8.

Wang, Y., J. Tang, and R. Y. Fung (2014). “A column-generation-based heuristic algorithm for solving operating theater planning problem under stochastic demand and surgery cancellation risk”. In: *International Journal of Production Economics* 158, pp. 28–36.

Wolfe, H. B. (1979). “Disaggregation planning, scheduling, and allocation of nursing staff”. In: *Disaggregation*. Springer, pp. 563–575.

Zhu, S., W. Fan, S. Yang, J. Pei, and P. M. Pardalos (2018). “Operating room planning and surgical case scheduling: a review of literature”. In: *Journal of Combinatorial Optimization*.