An Adjustable Robust Optimization Approach to Multi-objective Personnel Scheduling Under Uncertain Demand

A Case Study at a Pathology Department

Author:

Roshanak Mahdavi

Supervisors:

Dr. Jonathan Patrick

Dr. Sarah Ben Amor

A thesis presented for the degree of

Master of Science in Health Systems

Telfer School of Management

University of Ottawa

Abstract

In this thesis, we address a multi-objective personnel scheduling problem where personnel's workload is uncertain and propose a two-stage robust modelling approach with demand uncertainty. In the first stage, we model a multi-objective personnel scheduling problem without incorporating demand coverage and, in the second stage, we minimize over or under-staffing after the realization of the demand and the assignments from the first stage. Two solution approaches are introduced for this model. The first approach solves the proposed model through a cutting plane strategy known as Benders dual cutting plane method, and the second approach reformulates the problem based on the strong duality theory. As a case study, the proposed model and the first solution approach are applied to an existing scheduling problem in the pathology department at The Ottawa Hospital. It is shown that the proposed model is successful at reducing the unmet demand while maintaining the performance with respect to other metrics when compared against the deterministic alternative.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Many decision support tools are developed based on deterministic models. Although in some cases they may have satisfactory performance, depending on different factors, the derived solution can be inaccurate and even infeasible. This inaccuracy is usually due to a parameter's deviations from its assumed value in the theoretical model. These deviations can threaten the accuracy of the model and make the outcome undesirable [1]. In situations where uncertainty has a strong impact on the outcome of the problem, incorporating the uncertainty in the model could be very beneficial. Over the past few decades, different problems under uncertainty have been studied and different approaches have been developed. Here we focus on personnel scheduling problems under uncertainty.

Personnel scheduling is one of the most popular problems that has been studied since the 1950s. Although many researchers have ignored the uncertainty, the need to arrive at more reliable solutions cannot be ignored. Ever-increasing labor costs is only one of the reasons why scheduling processes would benefit from a more realistic method of modelling. This need is intensified when the staff are skilled and when the quality of service is very important.

While researchers have defined different stages for personnel scheduling or what is referred to as "rostering" [2][3], they mostly outline two steps: first, determine the workload, and second, allocate staff to the required workload. Researchers have confronted three sources of uncertainty throughout these steps: uncertainty in service times, uncertainty in arrivals, and uncertainty in capacity (manpower) [4]. In the next chapter, two of the most frequent approaches for handling uncertainty are explained.

Chapter 2

Literature Review

Several strategies to deal with different types of uncertainty have been developed over the past few decades. In a conventional approach known as stochastic programming, the distribution of an uncertain variable is realized based on historical data, and the probabilistic values for the next planning horizon are forecasted. Then, an optimization model solves the problem over a set of scenarios derived from the probabilistic values. This results in a stochastic program which includes the parameter's uncertainty. Stochastic programming is a popular approach that has been applied to many problems. In spite of that, this approach is considered impractical in many applications. One of its main challenges is that stochastic programming can be intractable and computationally challenging as the number of scenarios increases. Another challenge with the application of this method is to find the exact distribution of the uncertain variable(s) when there is little information available about it. Even with enough historical data, the variable's behaviour may change over time and tracking its values may not be practical.

2.1 Stochastic Programming

Despite its limitations, stochastic programming has been successfully applied for many years to different problems. Here we briefly mention some of the applications involving personnel scheduling and assignment decisions. Morton et al. [5]

use stochastic programming in order to model an employee scheduling problem in a production environment and utilize a Bayesian forecasting model to describe machine availabilities and production rates. Parisio et al. [6] use a two-stage stochastic programming approach to generate multi-task employee schedules for retail outlets where the demand for sales is uncertain. In the first stage, an initial employee assignment is made and, in the second stage, the decision compensates for deficiencies in the schedule after observing demand realizations by recalculating under-staffing and over-staffing levels for each task. Punnakitikashem et al. [7] use a stochastic integer programming model to minimize the excess workload of nurses and staffing costs in an integrated nurse staffing and assignment problem. They apply three solution approaches which are Benders decomposition, Lagrangian relaxation with Benders decomposition, and a heuristic based on nested Benders decomposition.

2.2 Robust Optimization

In the early 1970s, another approach to handle uncertainty called Robust Optimization was introduced [8]. The goal was to protect the model against the worst-case scenario(s) arising from realizations of the variables. These realizations are assumed to have known bounds and belong to a known set called the uncertainty set. Therefore, a robust model is not stochastic but rather deterministic and set-based [9]. Depending on the problem and the uncertain parameters, different structures for modelling the uncertainty set have been proposed. In 1998, robust optimization was further discussed by Ben-Tal et al. [10]. They proposed ellipsoidal uncertainty sets that lead to more computationally tractable models. A few years later Bertsimas et al. [11] proposed a robust optimization approach where the uncertain parameters belong to polyhedral uncertainty sets, which has further advantages in terms of tractability for large-scale problems. Since then, many researchers have followed their work and applied robust optimization in different problem settings, including personnel scheduling and assignment problems.

For example, Chen et al. [12] describe a two-stage algorithm to determine minimum medical staff levels and staff schedule separately. In the first stage, they adopt a robust approach to determine the minimum number of staff under uncertain demand for service. In the second stage, they define hard and soft constraints for a monthly medical staff schedule and apply the Analytic Hierarchy Process (AHP) to determine the penalty for each soft constraint. Zhen [13] considers a task assignment problem under workload uncertainty and proposes both a stochastic programming model and a robust optimization model for this problem. He concludes that the robust optimization model can find a good solution, that is near to the optimal solution for the stochastic programming model, by appropriately adjusting the degree of conservatism (an adjustable parameter which allows decision-maker to change the level of conservatism). Carello et al. [14] apply a robust approach to a nurse-to-patient assignment problem involving continuity of care considerations. Their model assumes demand for care is uncertain and its values belongs to a polyhedral set. It is concluded that the robust model has a better performances in terms of overtime costs and fairness in nurses' workload in comparison to the deterministic version of the model. Liao et al. [15] formulate a staffing problem in a call center setting to minimize the total operating cost with uncertain and non-stationary arrival rates as both a stochastic model and a robust optimization model. They conduct a numerical study and use real data from a hospital in order to evaluate and analyze the results from the two approaches. Tang et al. [16] describe a robust model to solve a surgery capacity allocation problem with demand uncertainty, that seeks to minimize the revenue loss from a shortage of allocated OR resources. They also compare the robust approach with a scenario-based stochastic optimization approach and show that the robust optimization approach outperforms stochastic optimization on limiting the worst-case outcome of the surgery capacity allocation problem.

2.2.1 Adjustable Robust Optimization

Frequently, for many different problems, the decision-making process under uncertainty can be decomposed into multiple steps [17]. While robust optimization yields a static solution for problems, a multi-stage robust optimization approach or what is referred to as Adjustable Robust Optimization (ARO) can provide better solutions by allowing more flexibility. The adjustable robust optimization approach proposed by Ben-Tal et al. in 2004 [18] is a dynamic approach which divides variables into two types: adjustable (wait-and-see) and non-adjustable (here-and-now). The values for here-and-now variables should be specified at the first stage, whereas the model allows wait-and-see variables to adjust after the realization of at least some of the uncertainty. Furthermore, an Adjustable Robust Counterpart (ARC) of a problem is significantly less conservative than the usual robust counterpart. This is because allowing the wait-and-see decisions to change according to the realized data provides more flexibility to the decision making process. More precisely, "ARO yields optimal objective values that are at least as good as those obtained using the standard RO approach" [19].

Mattia et al. [20] describe a two-stage adjustable robust model for shift scheduling at a call center. In the first stage, the staff assignment cost is minimized and, in the second stage, they minimize the worst-case reallocation cost. The uncertainty set in this problem has a special structure that comprises the dependency between variations in consecutive time slots. They test the efficiency of their model and the solution approach on real data, from a large distributed call center. Neyshabouri et al. [21] propose a two-stage robust optimization model for surgery scheduling to address the uncertainty in surgery durations and patients' length-of-stay. In the first stage, their model assigns patients to surgery blocks and in the second stage, the model minimizes the worst-case scenario in terms of overtime and denied admission costs.

2.3 Solution approaches

Several approaches have been developed to solve robust or adjustable robust optimization problems. According to Zeng et al. [22], these solution approaches can be classified into two main groups: approximations and exact reformulations.

2.3.1 Approximations

In general, adjustable robust models are intractable and hard to solve. To deal with this issue, Ben Tal et al. [18] propose a special form of ARC where the second-stage decision variable (i.e., the wait-and-see variable) can be written as an affine function of the uncertain parameter(s). The adjustable robust counterparts containing affine functions in their second-stage variables are called Affinely Adjustable Robust Counterpart (AARC). In [19], Yanikoglu et al. describe a small example of an AARC model for an inventory problem where the production in a time period depends affinely on the demand from the previous time periods.

2.3.2 Exact reformulations

An exact reformulation of a robust counterpart can be obtained using approaches such as vertex enumeration algorithm introduced by Bienstock and Ozbay [23], cutting plane algorithms, or reformulation of each constraint.

The vertex enumeration algorithm is applicable only when the uncertainty set is a finite set [24]. This method, which is also called the implementer-adversary algorithm, is a variant of the Benders decomposition. As an example, Tang et al. [16] solve a robust surgery capacity allocation problem with the implementer-adversary algorithm.

Also, applying the reformulation per constraint approach is applicable when the second stage of the problem contains small number of extreme points in its convex hull. An instance of an application of this approach can be found in [25].

Cutting Plane methods

When an approximation or a reformulation approach is not applicable due to the structure of the problem, solving adjustable robust models with a cutting plane algorithm is a popular strategy. Several cutting-plane methods have been proposed over the past few years to deal with two-stage robust problems. For example, Gabrel et al. [26] propose a mixed integer programming formulation to solve a problem that involves a quadratic second stage. Their cutting plane method is based on Kelley's cutting plane approach [27] and defines a tight bound for their model. As a numerical experiment, they present a two-stage robust location-transportation problem.

Bertimas et al. [28] propose a two-level solution methodology based on a combination of Benders decomposition and the outer approximation technique to decompose the overall two-stage problem into a master problem corresponding to the first-stage decisions at the outer level and a bilinear sub-problem corresponding to the second-stage actions at the inner level, which is solved by an outer approximation approach.

Jiang et al. [29] study a two-stage robust integer programming model which addresses the unit commitment (UC) problem under supply and demand uncertainty. They solve this problem with both an exact solution approach and a heuristic approach that provides a tight lower bound.

Zeng et al. [13] propose a column-and-constraint generation method which dynamically generates cuts that are defined by a set of created recourse decision variables in the form of constraints in the recourse problem. This procedure is implemented in a master-subproblem framework. They implement and test their method on a location-transportation problem.

Due et al. [30] propose a two-stage adjustable robust model for a facility location problem that minimizes the maximum transportation cost between an opened facility and a client under different scenarios. They provide three solution methods for this problem: a linear integer reformulation, a benders dual algorithm, and a column-and-constraint generation method. They conclude that the performance of the Benders dual method is the best for most of the instances, whereas the column-and-constraint generation algorithm is competitive and faster for the larger ones.

Terry et al. [31] propose three cutting plane methods to solve robust models with recourse and right-hand-side uncertainty. The first method is a cutting-plane method based on Kelley's cutting plane algorithm [27], the second one is an Analytic Center Cutting-Plane Method (ACCPM), and the third is a subgradient algorithm. They show that the performance of Kelley's cutting-plane algorithm is far better than both the ACCP and the subgradient methods in solving a newsvendor problem with simple recourse.

2.4 Personnel scheduling problems and schedule preferences

For many companies, reducing labor costs is a major incentive that motivates them toward implementing a new staff schedule. Thus, in many staff scheduling problems, the main objective is to minimize costs. However, this needs to be balanced with meeting the employee needs and preferences and providing quality service.

A comprehensive literature review on personnel scheduling problems by Van den Bergh et al. [4] outlines model constraints that cover staff preferences, schedule flexibility, and performance. They categorize these constraints as coverage constraints, time-related constraints, fairness constraints, and balance constraints.

The coverage constraints ensure that there are enough staff for every shift, task, or time slot during the entire planning period [32]. These constraints can be incorporated into the schedule as soft or hard constraints depending on the problem's requirements. Hard constraints are the regulations or conditions that cannot be changed whereas soft constraints may be violated but at a cost [12]. While in most studies the coverage constraints are modelled as hard constraints [4], there are problems where under-staffing is allowed and the model penalizes deviations from the forcasted demand for service. In what follows, we mention some of many studies that focus on department or staff preferences and incorporate these conditions in multi-objective programming approaches in personnel scheduling problems.

Topaloglu et al. [33] present a goal programming approach for a tour scheduling problem that models coverage as a soft constraint. In addition to coverage, the model takes into account personnel working hour preferences, shift type preferences, and employee's weekend-off request as soft constraints. Chen et al. [12] consider personnel preferences as soft constraints in a medical staff scheduling problem and apply the Analytic Hierarchy Process (AHP) to determine the penalty for every soft constraint. Topaloglu in [34] proposes a multi-objective programming model to schedule emergency medicine residents which penalizes the deviations from each goal in the objective function. The goals include the residents' working days, weekendsoff and the fairness in the distribution of positions. The weights in the objective function are determined by the AHP method using pairwise comparisons. In [35], Li et al. propose a hybrid approach consisting of goal programming and a meta-heuristic search for multi-objective scheduling problems such as a nurse rostering with many hard and soft constraints. The soft constraints include preferences for shifts, number of working days, and avoiding certain types of shift successions. Azaiez et al. [36] also propose a goal programming approach for nurse scheduling where the objective function penalizes the weighted deviations from the problem's preferences.

Chapter 3

Research Gap

Uncertainty is an inevitable factor in real-world problems. In a personnel scheduling problem, uncertainty can emerge from different sources such as service times (durations), arrivals(service requests), and capacity (manpower). In most cases, the uncertainty in arrivals is coupled with the uncertainty in service times [4]. In this research, we will refer to both as the uncertainty in demand.

In two-stage problems, the second stage allows the decision-maker to postpone some decisions until after the uncertainty is realized. Since this helps to compensate for any deficiencies caused by the uncertain data, the wait-and-see variable is also called the recourse variable. In the same way, the second-stage objective function is also called the recourse function.

In problems with demand uncertainty, usually the recourse function deals with the unmet demand. Since robust problems with recourse consider the worst case, covering the unmet demand while maintaining the same level of objective value may not be possible. Mostly in problems with recourse, the first-stage objective is to minimize cost. Therefore, when the level of robustness of the problem changes, the cost changes accordingly.

It frequently arises in the literature that authors study scheduling under demand uncertainty where the first stage objective is to minimize the cost associated with that schedule. Then, in the analysis section, they investigate the relationship between solution's robustness and the cost associated with that robust decision. To our knowledge, there have been only a few studies where, in a multi-objective context, the relationship between robustness and its effects not on schedule's costs, but to metrics representing schedule's goals is investigated. In this research, we consider a robust multi-objective staff scheduling problem with recourse and demonstrate the relationship between robustness of demand coverage and the performance of the schedule with respect to the other goals such as personnel preferences, utilizations, staff's workload, etc.

3.1 Research questions

To bridge the gap mentioned in the previous section, we need to answer the following questions:

1. How can one find an adjustable robust solution for a multi-objective staff scheduling problem under uncertain demand?
2. How different is the solution to the deterministic model from its robust counterpart?
3. How does robustness in demand coverage in a multi-objective staff scheduling problem affect other goals of that problem?

In order to answer the above questions, we formulate and solve a robust staff scheduling model with recourse. Here is an outline of the structure of this thesis: Chapter 4 depicts a general multi-objective staff scheduling problem and describes the steps required to formulate its adjustable robust counterpart. After that, it proposes solution approaches to solve the adjustable robust model. Chapter 5 introduces a case-study based on a personnel scheduling problem faced by the Pathology Department at The Ottawa Hospital and describes the application of the proposed model and solution approach to this problem. Chapter 6 presents results, analyses and conclusions obtained from solving the robust staff scheduling model.

Chapter 4

Methodology

Consider a multi-objective personnel scheduling problem with demand uncertainty. Let P be the set of personnel, T the set of tasks, and D the set of days in a planning horizon. The decision variable x_{ptd} represents the amount of demand from task t assigned to person p on day d. Assume the model should satisfy $|I|$ number of schedule goals, each represented by function F_i and target T_i, and the problem's objective function minimizes any deviation from $i^{th} \in I$ goal:

$$\begin{aligned} \min_{x,h} \quad & \sum_{i \in I} m_i \Big(\mu_i^1 h_i^+ + \mu_i^2 h_i^- \Big) \\ s.t. \quad & F_i(x) + h_i^+ - h_i^- = T_i \qquad \forall i \in I \\ & x \in S \quad h^+, h^- \in \mathbb{R}^+ \end{aligned} \tag{4.1}$$

where S is the region created by the problem's constraints, m_i is the weight corresponding to the i^{th} goal, and μ_i^1 and μ_i^2 are binary coefficients for h_i for including or excluding the variables. Assume one of the problem's constraints is the demand coverage constraint represented in (4.2), where its target is demand b_{td} for task t on day d:

$$\sum_{p \in P} x_{ptd} + h_{td}^+ - h_{td}^- = b_{td} \qquad \forall t, d \tag{4.2}$$

In this research, we model and solve an adjustable robust counterpart of the above problem when demand is uncertain. The solution of the proposed ARO model will not only be protected against changes in the uncertain demand, but it will also improve the solution after the realization of the demand using a recourse function.

In previous sections, some advantages of robust optimization over stochastic programming were mentioned. Once again, the reasons to choose robust optimization over stochastic programming for modeling the above problem can be summarized as follows:

- Robust optimization can result in more computationally tractable models in comparison with stochastic approaches. This is because robust models optimize over an uncertainty set rather than over the distribution of the uncertainty. This deficiency in stochastic models is magnified in large-scale problems due to the so-called "curse of dimensionality".
- The scheduling process can happen months in advance when there is little information available about the demand. Robust optimization allows modelling when there is little available information about the uncertain parameter.
- Uncertainty in a robust model can be introduced in the form of a polyhedron (polyhedral uncertainty) or an ellipsoid (ellipsoidal uncertainty). This corresponds to having infinite sets of scenarios [17] and thus providing stronger confidence in the robustness of the solution.

The reasons for modeling the problem as an adjustable robust model are as follows:

- ARO allows decision making in more than one stage. If an important decision depends on uncertain data, the recourse function or the second-stage objective function can help compensate for the unpredictable variations of the uncertainty.
- ARO is less conservative than the classic RO approach. This is also due to having a recourse function since a final decision is made after adjusting the recourse variable.
- A one-stage reformulation cannot be applied to the mentioned problem, since the uncertainty is placed in the RHS of an equality constraint, which causes the robust single stage problem to be infeasible [20].

- And finally, a static solution often does not correspond to what is done in practice since all the staffing levels that were computed before the realization of the demand may be adjusted when the real demand becomes available [37].

4.1 Definition of the first and second stages

This section outlines the steps required to develop a multi-objective robust staff scheduling model with recourse. First, we define the here-and-now and wait-and-see variables and then we state the formulation for each stage.

4.1.1 Adjustable and non-adjustable variables

One possible way to formulate the adjustable robust counterpart of model (4.1) is to initially assign tasks to staff in the first stage, without considering the demand and the coverage constraints. Then, as a recourse function, we can minimize the amount of over-staffing and under-staffing after the realization of the uncertainty. In that case, the first-stage variable is the assignment variable from model (4.1), denoted by x_{ptd}. It is assumed that the demand is the same every day, so the index d is deleted from variable x in formulations (It is possible to remove this assumption with only minimal changes to formulation). We define the second-stage variables U_t and $O_t \in \mathbb{R}^+$ as the amount of under-staffing and over-staffing after the realization of the demand for task t.

4.1.2 The first stage

Assume there is a number of goals for this scheduling problem and one of them is to meet the demand. For the sake of simplicity, we lump the penalties corresponding to the rest of the goals of the objective function into term $J(h)$ and represent the constraints related to them by set S. Thus, the two-stage robust model would be:

$$\begin{aligned} \min_{x,h} \quad & J(h) + Q(x,b) \\ s.t. \quad & x \in S \end{aligned} \tag{4.3}$$

where $Q(x,b)$ is the recourse function or the second-stage objective function and represents the minimum value for over-staffing or under-staffing over all tasks after the realization of the demand.

4.1.3 The second stage

Assume in a day, we realize that the demand for task t is $\boldsymbol{b_t}$. Our assignments from the first-stage model for that specific day are $\boldsymbol{x_{pt}}$, and the difference between demand and the total amount of demand assigned to staff for each task is $[\boldsymbol{b_t} - \sum_{p\in P} \boldsymbol{x_{pt}}]$. We can write the recourse function as:

$$\begin{aligned} Q(x,b) \;=\; \min_{U,O} \quad & \sum_{t\in T} \Big(w_1\, U_t + w_2\, O_t \Big) \\ s.t. \quad & U_t - O_t = \boldsymbol{b_t} - \sum_{p\in P} \boldsymbol{x_{pt}} \qquad \forall t \in T \\ & U\,,\, O \in \mathbb{R}^+ \end{aligned} \tag{4.4}$$

where w_1 and w_2 are the weights penalizing over-staffing and under-staffing, respectively. The recourse function $Q(x,b)$ minimizes the amount of over-staffing and under-staffing after observing the demand for different tasks. Recall that after the first stage both the assignments and the demand are realized. Therefore, in the above second-stage model, $\boldsymbol{x_{pt}}$ and $\boldsymbol{b_t}$ are parameters.

Before proceeding any further, we need to define the uncertainty set. As stated earlier, robust models introduce non-stochastic uncertainties confined to a set known as the uncertainty set. The structure of this set for the problem at hand is described in the following section.

4.1.4 Uncertainty set structure

Uncertainty sets can be defined in different structures. We follow Bertsimas and Sim's work in [38] and assume that the demand vector on each day belongs to a polyhedral uncertainty set B where:

$$B = \{b : \quad b_t = \bar{b}_t + \hat{b}_t e_t\,, \quad \forall t \in T \quad | \quad e \in E\} \tag{4.5}$$

and we define set E as:

$$E = \{e : \quad \sum_{t\in T} e_t \leq \Gamma\,, \quad 0 \leq e_t \leq 1\,, \quad \forall t \in T\} \tag{4.6}$$

where $\overline{b}_t$ is the demand's nominal value and $\hat{b}_t$ is the maximum deviation of $b_t \in [\overline{b}_t \ , \ \overline{b}_t + \hat{b}_t] \quad \forall t \in T$.

Γ is called the budget of uncertainty and determines the level of conservatism in a robust model. This parameter can take values between 0 and $|T|$. If $\Gamma = 0$, then all the e_t values will be zero. This will result in forcing all b_t to have their nominal values which means there will be no protection against uncertainty. By contrast if $\Gamma = |T|$, all of the e_t values and as a result, all of the b_t values are free within their corresponding bounds, which yields a very conservative solution [39].

Since the purpose of this research is to create a schedule with a more robust solution that can yield more coverage against higher demand realizations in the day of the practice, the proposed uncertainty set (which is a one-sided uncertainty set) protects model's solution against the realizations of the demand that need higher levels of staffing. If in a multi-objective model, the other parts of the objective function control not to over-utilize resources, covering higher realizations of the demand would be desirable for a robust model. After that, balancing between resource utilization and demand coverage can be achieved by adjusting the budget of uncertainty (Γ).

4.2 The robust personnel scheduling model with recourse

Recall that the recourse function is:

$$(\boldsymbol{P}) \qquad \begin{aligned} Q(x,b) \;=\; \min_{U,O} \quad & \sum_{t \in T} \Big(w_1\, U_t + w_2\, O_t \Big) \\ s.t. \quad & U_t - O_t = \boldsymbol{b_t} - \sum_{p \in P} \boldsymbol{x_{pt}} \qquad \forall t \in T \\ & U \,,\; O \in \mathbb{R}^+ \end{aligned}$$

In the above model, the realized demand $\boldsymbol{b} \in B$ takes a value in the uncertainty set described in the previous section. As a robust model, the solution must remain feasible for almost any value of the uncertain demand in set B. Since robust optimization is based on an adversarial approach which considers the worst-case scenario, the solution of this model must remain feasible for any realization of the demand, including when it takes its maximum values which makes the coverage the most difficult. In light of this, we need to maximize the demand in $Q(x, b)$ over set B:

$$\begin{aligned} \min_{U,O} \quad & \sum_{t \in T} \Big(w_1\, U_t + w_2\, O_t \Big) \\ s.t. \quad & U_t - O_t = \max_{\boldsymbol{b}} \boldsymbol{b_t} - \sum_{p \in P} \boldsymbol{x_{pt}} \qquad \forall t \in T \\ & U,\; O \in \mathbb{R}^+ \end{aligned} \tag{4.7}$$

In model (4.7), the source of uncertainty b_t is in the right-hand-side of the constraint. This type of uncertainty is referred to as column-wise uncertainty and was first introduced by Soyster [8]. Unlike the row-wise uncertainty discussed by Ben-Tal [10] and Bertsimas [38] that assumes that each row of the constraint matrix belongs to a known uncertainty set consisting of an ellipsoid or a polyhedron, Soyster assumes that each column of a constraint matrix is either exactly known or supposed to belong to a given uncertainty set [40]. As some considered Soyster's modelling approach too conservative, they tried to change the column-wise uncertainty to a row-wise uncertainty by writing the dual of the problem, which in our case, can bring the uncertainty from the right-hand-side of the constraint to the objective function

[40]. As a result, we can define the uncertainty set as an ellipsoid or a polyhedron and introduce the budget of uncertainty, which leads to a less conservative solution. Notice that $Q(x, b)$ is feasible and bounded. By strong duality, we can rewrite $Q(x, b)$ and write the dual of $(\boldsymbol{P})$ as:

$$(\boldsymbol{D}) \qquad \begin{aligned} Q(x, b) = \quad & \max_{y} \sum_{t \in T} (\boldsymbol{b_t} - \sum_{p \in P} \boldsymbol{x_{pt}})\, y_t \\ s.t. \quad & -w_2 \leq y_t \leq w_1 \qquad \forall t \in T \end{aligned}$$

where y_t is the dual variable. We can now maximize the demand over set B and write $Q(x) = \max_{b \in B} Q(x, b)$ as:

$$\begin{aligned} Q(x) = \quad & \max_{b \in B,\, y} \sum_{t \in T} (b_t - \sum_{p \in P} \boldsymbol{x_{pt}})\, y_t \\ s.t. \quad & -w_2 \leq y_t \leq w_1 \qquad \forall t \in T \end{aligned} \tag{4.8}$$

By placing $\max_{b \in B} Q(x, b)$ in the two-stage problem, the robust version of problem (4.3) is as follows:

$$\begin{aligned} \min_{x,h} \quad & J(h) + \sum_{d} \max_{b \in B} Q(x, b) \\ s.t. \quad & x \in S \end{aligned} \tag{4.9}$$

By introducing an auxiliary variable λ, we can write the recourse functions $\max_{b \in B} Q(x, b)$ as a constraints and rewrite the two-stage problem as:

$$\begin{aligned} \min_{x,h} \quad & J(h) + \lambda \\ s.t. \quad & \lambda \geq Q(x, b) \qquad \forall b \in B \\ & x \in S \quad \lambda \in \mathbb{R}^+ \end{aligned} \tag{4.10}$$

or alternatively:

$$\begin{aligned} \min_{x,h} \quad & J(h) + \lambda \\ s.t. \quad & \lambda \geq \max_{y} \sum_{t \in T} (b_t - \sum_{p \in P} x_{pt}) y_t \qquad \forall b \in B \\ & -w_2 \leq y_t \leq w_1 \qquad \forall t \in T \\ & x \in S \quad \lambda \in \mathbb{R}^+ \end{aligned} \tag{4.11}$$

where λ represents the (weighted) worst-case over-staffing or under-staffing after the first-stage decisions. The above model is intractable in its current form due to the infinite number of constraints ($\forall b \in B$). There are several strategies to reformulate and solve the above problem that were introduced in Chapter 2. In what follows, we refer to two approaches, one cutting plane and one reformulation method to solve problem (4.11).

4.3 First approach: Benders dual cutting plane

In this section, we explain an iterative row generation strategy known as Benders-dual cutting plane algorithm for solving robust models. This algorithm starts with a relaxed version of the problem with a finite number of constraints and iteratively, adds more cuts (i.e. constraints) until optimality is reached. This method is based on a master-subproblem framework and is a variant of the general Benders decomposition. In our case, the master problem is a relaxation of the two-stage robust model (4.11) starting with a finite number of its constraints, and the sub-problem is the inner-problem in model (4.11). Therefore, we can write the master problem (MP) as follows:

$$\begin{aligned} \boldsymbol{(MP):} \quad & \min_{x,\,\lambda} \quad J(h) + \lambda \\ & s.t. \quad \lambda \geq \sum_{t \in T} (\dot{\boldsymbol{b}}_{\boldsymbol{ti}} - \sum_{p \in P} x_{pt}) \dot{\boldsymbol{y}}_{\boldsymbol{ti}} \qquad \forall i \leq j \\ & \qquad\; x \in S \quad \lambda \in \mathbb{R}^+ \end{aligned} \tag{4.12}$$

where x, h, and λ are the master problem's variables. The above problem represents the MP at iteration $(j+1)$, which includes j cuts generated in previous iterations. The values of $\dot{\boldsymbol{b}}_{\boldsymbol{tdj}}$ and $\dot{\boldsymbol{y}}_{\boldsymbol{tdj}}$ are the solutions from the sub-problem (SP) in the j^{th} iteration which corresponds to:

$$\begin{aligned} \boldsymbol{(SP):} \quad Q(x) = & \max_{b \in B,\, y} \sum_{t \in T} (b_t - \sum_{p \in P} \dot{\boldsymbol{x}}_{\boldsymbol{pt(j)}}) y_t \\ & s.t. \quad -w_2 \leq y_t \leq w_1 \qquad \forall t \in T \end{aligned} \tag{4.13}$$

where b and y are the decision variables in the sub-problem and $\dot{\boldsymbol{x}}_{\boldsymbol{pt(j)}}$ is the solution obtained from the master problem in the j^{th} iteration. At each iteration, the sub-problem introduces a new realization of the demand to the master problem so that the problem incorporates that specific demand realization.

The sub-problem is a bilinear optimization model which is in general NP-hard [41]. There are several strategies to reformulate the above model. We consider the approach proposed by Thiele et al. [25] specific to problems with the same structure as the second-stage (simple recourse). This strategy is illustrated in Section 4.3.1.

While solving the $\boldsymbol{MP}$ and the $\boldsymbol{SP}$ iteratively, the value of the objective function in the master problem provides a lower bound for the objective function of the two-stage problem, and by replacing λ with the second-stage objective function, it provides an upper bound for the objective function of the two-stage problem. The process of generating cuts and solving the two problems continues until the gap between the lower bound and the upper bound is less than a predefined constant.

To summarize, we provide a step-by-step algorithm based on the algorithm mentioned by Zeng et al. in [22] where LB and UB denote the lower and upper bound of the two-stage problem's objective value, j is the iteration number, and ϵ denotes the stopping criteria value (tolerance gap):

Algorithm 1: Benders-Dual Cutting Plane Algorithm

Initialization: Set $LB = 0$, $UB = \infty$, and $j = 0$

Step 1: Solve the master problem (4.12) and obtain the optimal solution ($\dot{\boldsymbol{x}}_{\boldsymbol{j+1}}$, $\dot{\boldsymbol{\lambda}}_{\boldsymbol{j+1}}$, $\dot{\boldsymbol{h}}_{\boldsymbol{j+1}}$)

Step 2: Update the lower bound $LB = \boldsymbol{J(\dot{h}_{j+1})} + \dot{\boldsymbol{\lambda}}_{\boldsymbol{j+1}}$

Step 3: Solve the sub-problem $\boldsymbol{SP}$ and obtain the optimal solution ($\dot{\boldsymbol{b}}_{\boldsymbol{j+1}}, \dot{\boldsymbol{y}}_{\boldsymbol{j+1}}$).

$$\boldsymbol{SP}: \quad \max_{b\in B,\, y} \sum_{t\in T}(b_t - \sum_{p\in P} \dot{\boldsymbol{x}}_{\boldsymbol{pt(j+1)}})y_t$$

$$s.t. \quad -w_2 \leq y_t \leq w_1 \qquad \forall t \in T$$

Step 4: Update $UB = min\Big\{UB \ , \ \boldsymbol{J(\dot{h}_{j+1})} + Q(\dot{\boldsymbol{x}}_{\boldsymbol{j+1}})\Big\}$

Step 5: If $\frac{UB-LB}{LB} \leq \epsilon$, return $\dot{\boldsymbol{x}}_{\boldsymbol{j+1}}$ and terminate. Otherwise, update $j = j+1$ and go to step 1 with the addition of the following constraint to the MP.

$$\lambda \geq \sum_{t\in T} \big(\dot{\boldsymbol{b}}_{\boldsymbol{t(j+1)}} - \sum_{p\in P} x_{pt}\big)\dot{\mathbf{y}}_{\mathbf{t(j+1)}}$$

Here we have the complete recourse property that the sub-problem is always feasible for any given value of x and b. This is because the sub-problem is simply calculating the over or under-staffing for a given set of x and a realized set of b, and it does not specify any bounds for the variables.

4.3.1 Reformulating the sub-problem

In this section, we follow the method developed by Thiele et al. in [25] for computing the sub-problem $Q(x)$. Recall that the structure of the sub-problem is:

$$\begin{aligned} \boldsymbol{SP}: \quad Q(x) = & \max_{b\in B,\, y} \sum_{t\in T}(b_t - \sum_{p\in P} \boldsymbol{\widehat{x}_{pt(j-1)}})y_t \\ & s.t. \quad -w_2 \leq y_t \leq w_1 \qquad \forall t \in T \end{aligned} \tag{4.14}$$

The constraint in the above model implies that each y_t should take either its lower bound value $-w_2$, or its upper bound value w_1. In light of this, we can rewrite $Q(x)$ as:

$$Q(x) = \max_{b\in B} \sum_{t\in T} \Big[\max \big\{ w_1(b_t - \sum_{p\in P} x_{pt}) \ , \ w_2(\sum_{p\in P} x_{pt} - b_t) \big\} \Big] \tag{4.15}$$

Recall that the structure of set B is:

$$\begin{aligned} B &= \{b: \quad b_t = \bar{b}_t + \hat{b}_t e_t \ , \quad \forall t \in T \quad | \quad e \in E\} \\ E &= \{e: \quad \sum_{t\in T} e_t \leq \Gamma \ , \quad 0 \leq e_t \leq 1 \ , \quad \forall t \in T\} \end{aligned} \tag{4.16}$$

which is a convex set. Given an integer budget of uncertainty Γ, maximizing over the set B would result in having Γ number of $(b_t = \bar{b}_t + \hat{b}_t)$ and $|T| - \Gamma$ number of $(b_t = \bar{b}_t)$. We can define a new parameter Δ_t as:

$$\begin{aligned} \Delta_t = \max \Big\{ w_1(\bar{b}_t + \hat{b}_t - \sum_{p\in P} x_{pt}) \ , \ w_2(\sum_{p\in P} x_{pt} - \bar{b}_t) \Big\} \\ - \max \Big\{ w_1(\bar{b}_t - \sum_{p\in P} x_{pt}) \ , \ w_2(\sum_{p\in P} x_{pt} - \bar{b}_t) \Big\} \end{aligned} \tag{4.17}$$

The first maximum term in Δ_t represents the unmet demand for task t given the assignments from the first stage if b_t reaches its upper bound in the second stage. In the same way, the second maximum term represents the unmet demand for task t given the assignments from the first stage if b_t stays at its nominal value in the second stage.

Since all of the terms in the above equation are realized after the first-stage, we can use Δ_t and reformulate $Q(x)$ as:

$$\begin{aligned} Q(x) =& \max_{b\in B} \sum_{t\in T} \max\Big\{w_1(b_t - \sum_{p\in P} x_{pt}) \ , \ w_2(\sum_{p\in P} x_{pt} - b_t)\Big\} \\ =& \sum_{t\in T} \max\Big\{w_1(\bar{b}_t - \sum_{p\in P} x_{pt}) \ , \ w_2(\sum_{p\in P} x_{pt} - \bar{b}_t)\Big\} + \max_{e\in E} \sum_{t\in T} \Delta_t e_t \end{aligned} \tag{4.18}$$

We can rewrite equation (4.18) as:

$$\begin{aligned} Q(x) = \sum_{t\in T'} \max\Big\{w_1(\bar{b}_t + \hat{b}_t - \sum_{p\in P} x_{pt}) \ , \ w_2(\sum_{p\in P} x_{pt} - \bar{b}_t)\Big\} \\ + \sum_{t\notin T'} \max\Big\{w_1(\bar{b}_t - \sum_{p\in P} x_{pt}) \ , \ w_2(\sum_{p\in P} x_{pt} - \bar{b}_t)\Big\} \end{aligned} \tag{4.19}$$

where T' is the set of indices for the first Γ tasks, if we sort the tasks based on their Δ_t values in a descending order.

Equation (4.19) indicates that model $Q(x)$ takes the demand for tasks with larger Δ_t to their extreme values and leaves the rest of the demand at their nominal values. In equation (4.18), the variable e_t determines whether a task's demand should deviate from its nominal value.

Recall that variable x is determined in the first-stage. Thus, we can specify the values for the worst-case demand b_t in model (4.19) as follows:

$$b_t = \begin{cases} \bar{b}_t + \hat{b}_t & if \quad t \in T' \quad and \quad w_1(\bar{b}_t + \hat{b}_t - \sum_p x_{pt}) \geq w_2(\sum_p x_{pt} - \bar{b}_t) \\ \bar{b}_t & if \quad t \in T' \quad and \quad w_1(\bar{b}_t + \hat{b}_t - \sum_p x_{pt}) < w_2(\sum_p x_{pt} - \bar{b}_t) \\ \bar{b}_t & if \quad t \notin T' \end{cases} \tag{4.20}$$

and the values for the dual variable y_t can be summarized as:

$$y_t = \begin{cases} w_1 & if \quad t \in T' \quad and \quad w_1(\bar{b}_t + \hat{b}_t - \sum_p x_{pt}) \geq w_2(\sum_p x_{pt} - \bar{b}_t) \\ -w_2 & if \quad t \in T' \quad and \quad w_1(\bar{b}_t + \hat{b}_t - \sum_p x_{pt}) < w_2(\sum_p x_{pt} - \bar{b}_t) \\ w_1 & if \quad t \notin T' \quad and \quad w_1(\bar{b}_t - \sum_p x_{pt}) \geq w_2(\sum_p x_{pt} - \bar{b}_t) \\ -w_2 & if \quad t \notin T' \quad and \quad w_1(\bar{b}_t - \sum_p x_{pt}) < w_2(\sum_p x_{pt} - \bar{b}_t) \end{cases} \tag{4.21}$$

We can now implement the benders-dual cutting plane algorithm on our two-stage problem.

4.4 Second approach: Reformulation

If $Q(x)$ includes a relatively small number of extreme points in its feasible region, we can reformulate the two-stage problem as a linear optimization problem [42]. Recall that $Q(x)$ was:

$$\begin{aligned} Q(x) = \quad & \max_{b \in B,\, y} \sum_{t \in T} \Big((b_t - \sum_{p \in P} x_{pt}) y_t \Big) \\ s.t. \quad & -w_2 \leq y_t \leq w_1 \qquad \forall t \in T \end{aligned} \tag{4.22}$$

Considering the possible values of variable y, it is straightforward to see that the convex hull corresponding to the above model has $2^{|T|}$ extreme points. If we define set I as $I = \{1, 2, ..., 2^{|T|}\}$, we can introduce y_t^i $(i \in I)$ as the coefficients for $(b_t - \sum_p x_{pt})$ in the i^{th} scenario and we have:

$$Q(x) = \max_{b \in B} \max \Big\{ \sum_{t \in T} (b_t - \sum_{p \in P} x_{pt}) y_t^1 \;,\; \sum_{t \in T} (b_t - \sum_{p \in P} x_{pt}) y_t^2 \;,\; \ldots \;,\; \sum_{t \in T} (b_t - \sum_{p \in P} x_{pt}) y_t^{2^t} \Big\} \tag{4.23}$$

Or alternatively, we can write:

$$Q(x) = \max_{b \in B} \max_{i \in I} \Big\{ \sum_{t \in T} (b_t - \sum_{p \in P} x_{pt}) y_t^i \Big\} \tag{4.24}$$

If the number of tasks is small, we can enumerate all of the extreme points in the solution space. Placing $Q(x)$ into robust two-stage model, we have:

$$\begin{aligned} \min \quad & J(h) + \max_{b \in B} \max_{i \in I} \Big\{ \sum_{t \in T} (b_t - \sum_{p \in P} x_{pt}) y_t^i \Big\} \\ s.t. \quad & x \in S \end{aligned} \tag{4.25}$$

By introducing the dummy variable λ, we can bring the $2^{|T|}$ terms in the maximization to the constraints. Thus, we have:

$$\begin{aligned} \min \quad & J(h) + \lambda \\ s.t. \quad & \lambda \geq \max_{b \in B} \sum_{t \in T} (b_t - \sum_{p \in P} x_{pt}) y_t^i \qquad \forall i \in I \\ & x \in S \end{aligned} \tag{4.26}$$

Recall that the demand values belong to set B. We can rewrite the inner maximization in (4.26) as:

$$\max_{b\in B}\sum_{t\in T}(b_t-\sum_{p\in P}x_{pt})y_t^i \;=\; \sum_{t\in T}(\bar{b}_t-\sum_{p\in P}x_{pt})y_t^i \;+\; \max_{e\in E}\sum_{t\in T}\hat{b}_t|y_t^i|e_t \qquad \forall i\in I \tag{4.27}$$

Since the above problem is feasible and bounded, by strong duality, (4.27) is equivalent to:

$$\begin{aligned}
\max_{b\in B}\sum_{t\in T}(b_t-\sum_{p\in P}x_{pt})y_t^i \;=\; \sum_{t\in T}(\bar{b}_t-\sum_{p\in P}x_{pt})y_t^i \;+\; &\min_{r,\pi}\sum_{t\in T}\pi_t+\Gamma r \qquad \forall i\in I\\
&s.t.\quad \pi_t+r\geq \hat{b}_t|y_t^i| \quad \forall t,\ i\in I\\
&\qquad\quad \pi,r\geq 0 \qquad \forall t\in T
\end{aligned} \tag{4.28}$$

By substituting the above inner-model into (4.26), we have:

$$\begin{aligned}
\min_{x}\quad & J(h)+\lambda\\
s.t.\quad & \lambda\geq\sum_{t\in T}(\bar{b}_t-\sum_{p\in P}x_{pt})y_t^i \;+\; \min_{r,\pi}\sum_{t\in T}\pi_t+\Gamma r \qquad \forall i\in I\\
& \qquad\qquad\qquad\qquad \pi_t+r\geq\hat{b}_t|y_t^i| \qquad \forall t,\ i\in I\\
& \qquad\qquad\qquad\qquad x\in S \quad \pi,r\geq 0 \qquad \forall t\in T
\end{aligned} \tag{4.29}$$

We can remove the minimization term from the RHS of the above constraint since it is sufficient for the constraint to hold for at least one π and r [19].

$$\begin{aligned}
\min_{x,\,\pi\, r}\quad & J(h)+\lambda\\
s.t.\quad & \lambda\geq\sum_{t\in T}(\bar{b}_t-\sum_{p\in P}x_{pt})y_t^i \;+\; \sum_{t\in T}\pi_t+\Gamma r \qquad \forall i\in I\\
& \pi_t+r\geq\hat{b}_t|y_t^i| \qquad \forall t\in T,\ i\in I\\
& x\in S \quad \pi,r\geq 0 \qquad \forall t\in T
\end{aligned} \tag{4.30}$$

Now that the problem has been reformulated as an LP, we can simply call an optimization solver to find an optimal robust solution to the problem.

Chapter 5

A Case Study

To test our model, we apply the proposed robust model with recourse to a personnel scheduling problem faced by the Department of Pathology and Laboratory Medicine (DPLM) at The Ottawa Hospital. The DPLM, which employs 30 pathologists who cover 20 subspecialties (tasks) had relied on manual scheduling to assign pathologists to different tasks until 2015 when Montazeri et al. [43] developed an initial deterministic scheduling model. In 2019, Patrick et al. extended this model by incorporating six performance metrics that were recognized as "what comprises a good schedule for the DPLM [44].

In both studies, the demand for each task is considered deterministic throughout the planning period (which is 5 weeks long). The models, which are multi-objective, assign pathologists to different tasks based on the forecasted demand while seeking to satisfy the department's needs and preferences. However, due to possible variations in the demand for the different subspecialties, the actual number of required pathologists may differ from what is suggested by the deterministic model. The deterministic model is successful at reducing the time required to develop a staff schedule and at meeting the department's needs; however, it could be more useful to create a robust solution that is protected against possible variations in the demand.

According to Ernst et al. [2], developing a personnel scheduling tool requires three main steps. The first step is to model and forecast the demand for service based on the historical data, and to convert it to staffing levels required to meet the demand. The second step is to determine the methodology and the solution approach for satisfying all of the conditions and goals in the problem setting, and the last step is to specify the reporting tool that displays the final solution. In the following paragraphs, we will go through these three steps and compare them with the steps required in scheduling the pathologists at TOH.

According to Patrick et al. [44], the first steps in scheduling the pathologists at the DPLM was to forecast the expected daily arrivals of specimens requiring diagnosis for each subspecialty for the entire planning period. After that, they translated this information into a daily resource requirement, which was the number of required pathologists for each subspecialty. These steps are in accordance with the first step in a general personnel scheduling problem mentioned earlier. Because finding the demand for service at the DPLM and translating it has been studied in the previous works, we take it as predetermined in this study.

The next step in the pathology scheduling problem in [44] was to assign pathologists to different subspecialties based some conditions, which mainly were to meet the daily demand for service as well as the department's conditions. This step is also similar to the second step in the personnel scheduling problem explained by Ernst et al. in [2]. In this research, we develop our MIP based on the deterministic model introduced by Patrick et al. [44] for the pathology scheduling problem at the DPLM, and incorporate the demand uncertainty on the existing model.

As the interface of the decision support tool which is the final stage of a personnel scheduling problem, the pathologists' schedule was returned to MS Excel in a form familiar to the administrative assistant at the DPLM.

In what follows, we describe the main structure of the deterministic model and the changes applied to it to make it suitable for the proposed robust approach.

The scheduling model proposed by Patrick et al. [44] introduces a unique set of objectives or performance metrics which are added to the model as soft constraints along with the model's main constraints. These model's objectives are listed below:

1. **Demand coverage:** Tracks missed assignments or unmet demand

2. **Utilization:** Controls each pathologist's workload

3. **Back-to-back workload:** Tracks occasions where the same task (subspecialty) is given to the same pathologist two weeks in a row

4. **Pathologist's workload:** Penalizes when a pathologist is given more than a full-time-equivalent (FTE) amount of workload on a day

5. **Schedule consistency:** Tracks the consistency of the subspecialties assigned to a pathologist throughout a week (consistency is defined as when a subspecialty is given to a pathologist for the entire week)

6. **Pathologist's rotations:** Penalizes schedules that do not rotate through a pathologist's rotation of tasks

Along with the soft constraints, the schedule must satisfy a number of conditions which are incorporated into the model as hard constraints (see Appendix A for a complete description of the model). These conditions can be listed as follows:

- Pathologists' assignment to a specific task (subspecialty) cannot be greater than 1 FTE (full-time equivalent) or between 0 and 0.1 of FTE.
- No pathologist is assigned to more than two subspecialties in a day
- No pathologist is given a pair of subspecialties on the same day unless that pair is admissible (a pair of subspecialties is admissible when it is specified in the "allowable combination" set by the department)
- The number of days a pathologist can be assigned to work in a week is restricted
- Model can leave holidays and specific subspecialties unassigned as necessary
- Pathologists are not assigned to any subspecialty when they are unavailable
- Model provides the option to pre-specify certain full-time assignments

- The number of pathologists assigned to each subspecialty in a week is restricted
- There are upper and lower bounds on the number of days a pathologist can be assigned to a subspecialty in a four week span
- User can specify when a given pathologist will be given x number of weeks or days off
- Specific pairings of subspecialties are not allowed even if the workload allows it
- A subspecialty is not assigned to more than 3 pathologists in a day

The above constraints define the set S that was introduced earlier in the previous chapter. The deterministic model follows a goal programming approach with six objectives, including demand coverage. The model's objective function is to penalize any deviations regarding the above criteria. In this chapter, we create the robust version of this model which is formulated around the coverage constraint, and then we apply the benders dual cutting plane method to solve it. In Chapter 6, we compare the proposed model with its deterministic version and investigate the impact of robustness on the other goals listed above.

5.1 The pathology scheduling problem

Let P denote the set of pathologists, T the set of tasks, D the set of days in the planning horizon, and $I = \{1, 2\}$ the set of assignment types (full-time and part-time). The decision variable x_{1ptd} is a binary variable representing whether pathologist p is assigned full-time to task t on day d. This assignment is equivalent to a full 8-hour daily shift. Similarly, the decision variable x_{2ptd} is a binary variable representing a part-time assignment of pathologist p to task t on day d. The amount of workload (weight) for this part-time assignment is determined by the parameter $W_t = b_t - \lfloor b_t \rfloor$, which is the fractional part of the demand for task t. In other words, the model does not consider any part-time assignments unless the demand has a fractional part (in keeping with current practice). Here we reformulate this problem similar to model (4.1):

$$
\begin{aligned}
\min_{x,h} \quad & J(h) + \sum_{t \in T} \sum_{d \in D} \delta_{td} \\
s.t. \quad & \sum_{p \in P} \left(x_{1ptd} + W_{td}\, x_{2pt} \right) + \delta_{td} = b_{td} \qquad \forall t, d \\
& \sum_{p \in P} x_{1ptd} \leq \lfloor b_{td} \rfloor \qquad \forall t, d \\
& \sum_{p \in P} x_{2ptd} \leq \lceil b_{td} \rceil - \lfloor b_{td} \rfloor \qquad \forall t, d \\
& x \in S \\
& x \in \{0, 1\} \quad \delta \in \mathbb{R}^{+}
\end{aligned} \tag{5.1}
$$

Again, for the sake of simplicity, we summarize all the other goals in the objective function as $J(h)$ and represent the constraints related to them by set S. Also, the deviations regarding the coverage constraint in task t on day d is denoted by δ_{td}.

In order to make this model robust, we need to change the decision variables from binary to continuous assignments. The reasons for this change is mainly because of the existence of the last two constraints in model (5.1), which define full-time and part-time assignments:

- If we solve the model using the first approach (cutting plane), then we need to bring the last two constraints of model (5.1) to the sub-problem as they include the uncertain parameter b_{td}. As a result of the existence of theses two constraints in the sub-problem, its structure would not be a simple recourse (where the decision-maker is addressing excess or shortage for each of the requirements independently [25]). Consequently, the proposed approach, which is specific to problems with simple recourse structure, would not be applicable here.
  Besides the above explanations, the complete recourse property (where the sub-problem is always feasible) would not hold anymore and the Benders algorithm would require feasibility cuts in addition to optimality cuts described in Algorithm 1.

- The second approach, specifically applying the transition from model (4.27) to model (4.28), will also not be applicable to this model in its current form. Since model (5.1) and consequently, its inner-model have constraints that involve both b_t and x_{pt}, reformulating the inner-model would not linearize the two-stage problem and the dual of model (4.27) would still be non-linear.

The proposed change will result in eliminating the weight parameter (W_t) and the last two constraints in model (5.1). Also, the decision variable $x_{iptd} \in \{0, 1\}$ will change to decision variable $x_{ptd} \in [0, 1]$ which is a continuous variable with the lower bound of 0 FTE and the upper bound of 1 FTE. Applying these modifications, the problem can be reformulated as follows:

$$\begin{aligned} min \quad & J(h) + \sum_{t \in T} \sum_{d \in D} \delta_{td} \\ s.t. \quad & \sum_{p \in P} x_{ptd} + \delta_{td} = b_{td} \qquad \forall t, d \\ & x \in S \\ & x \in [0, 1] \quad \delta \in \mathbb{R}^+ \end{aligned} \tag{5.2}$$

which has a similar form as model (4.1) introduced earlier. Based on the proposed approach for developing an adjustable robust model, and similar to problem (4.11), the two-stage robust version of the pathology problem can be written as:

$$\begin{aligned} min \quad & J(h) + \sum_{d \in D} \lambda_d \\ s.t. \quad & \lambda_d \geq \max_y \sum_{t \in T} (b_t - \sum_{p \in P} x_{ptd}) y_t \qquad \forall b \in B \quad \forall d \in D \\ & - w_2 \leq y_t \leq w_1 \qquad \forall t \in T \\ & x \in S \end{aligned} \tag{5.3}$$

The main difference between the above pathology model and the two-stage model proposed earlier is the planning periods. The pathology problem schedules for five weeks, whereas model (4.11) schedules for only one day. As a result, the recourse function of the pathology problem includes $|D|$ number of λ_d, instead of only one. We elaborate more on the differences in the next section.

5.2 Solving the pathology scheduling problem

The Pathology Department assigns 32 pathologists to 21 tasks (subspecialties). Since the number of tasks is not small, enumerating all of the extreme points in the sub-problem is not practical. Therefore, we cannot apply the second proposed solution approach which was to reformulate the two-stage robust problem.

In this section, we apply the cutting-plane strategy to the pathology problem with two minor modifications:

First, we need to change λ to λ_d, which implies having separate recourse functions for each day of the planning horizon. This is necessary since the first-stage creates different sets of assignments, one for each day. Therefore, the recourse values must be different for each day. We also consider first-stage and second-stage weights in the master problem's objective function, denoted by p_1 and p_2. The following equations represent the MP and SP at the $(j+1)^{th}$ iteration:

$$\begin{aligned} \boldsymbol{MP:} \quad & min_{x,\lambda} \quad p_1\, J(h) + p_2 \sum_{d \in D} \lambda_d \\ & s.t. \quad \lambda_d \geq \sum_{t \in T} (\dot{\boldsymbol{b}}_{\boldsymbol{ti}} - \sum_{p \in P} x_{ptd}) \dot{\boldsymbol{y}}_{\boldsymbol{ti}} \qquad \forall d \in D \quad \forall i \leq j \\ & \qquad x \in S \quad \lambda_d \in \mathbb{R}^+ \end{aligned} \tag{5.4}$$

The values of $\dot{\boldsymbol{b}}_{\boldsymbol{ti}}$ and $\dot{\boldsymbol{y}}_{\boldsymbol{ti}}$ are calculated based on equations (4.20) and (4.21) after solving the SP at each iteration.

$$\boldsymbol{SP:} \quad \sum_{t\,,\,d} \max \left\{ w_1(\bar{b}_t - \sum_{p \in P} \dot{\boldsymbol{x}}_{\boldsymbol{ptd(j+1)}})\ ,\ w_2(\sum_{p \in P} \dot{\boldsymbol{x}}_{\boldsymbol{ptd(j+1)}} - \bar{b}_t) \right\} + \max_{e\,\in\,E} \sum_{t\,,\,d} \Delta_{td}\, e_t \tag{5.5}$$

where set E is defined as:

$$E = \{ e: \quad \sum_{t \in T} e_t \leq \Gamma\,, \quad 0 \leq e_t \leq 1\,, \quad \forall t \in T \} \tag{5.6}$$

and Δ_{td} is:

$$\Delta_{td} = \max\Big\{w_1(\bar{b}_t + \hat{b}_t - \sum_{p\in P} \dot{\boldsymbol{x}}_{\boldsymbol{ptd(j+1)}})\ ,\ w_2(\sum_{p\in P} \dot{\boldsymbol{x}}_{\boldsymbol{ptd(j+1)}} - \bar{b}_t)\Big\} - \max\Big\{w_1(\bar{b}_t - \sum_{p\in P} \dot{\boldsymbol{x}}_{\boldsymbol{ptd(j+1)}})\ ,\ w_2(\sum_{p\in P} \dot{\boldsymbol{x}}_{\boldsymbol{ptd(j+1)}} - \bar{b}_t)\Big\}$$

The next modification incorporates the existence of holidays, pairs of (t, d) where there should not be any assignments. This requirement is incorporated into the pathology model using a binary parameter $Blank_{td}$. An assignment to task t on day d is needed only when $Blank_{td} = 1$.

This requirement is included in the master problem as a hard constraint on the assignments, whereas in the sub-problem and in the Benders cuts, the robust model falsely considers the entire demand as unmet where $Blank_{td} = 0$. This results in a large λ and a large missed assignments that may cause a wrong and quick convergence in the algorithm. To resolve this issue, the master and the sub-problem can be changed to:

$$\boldsymbol{MP:} \quad \min_{x,\,\lambda} \quad p_1\, J(h) + p_2 \sum_{d\in D} \lambda_d$$

$$s.t. \quad \lambda_d \geq \Big[\sum_{t\in T} Blank_{td}\, (\dot{\boldsymbol{b}}_{\boldsymbol{ti}} - \sum_{p\in P} x_{ptd})\dot{\boldsymbol{y}}_{\boldsymbol{ti}}\Big] \qquad \forall d \in D \quad \forall i \leq j$$

$$x \in S \quad \lambda_d \in \mathbb{R}^+ \qquad \forall d \in D$$

$$\boldsymbol{SP:} \quad \Big[\sum_{t\in T}\sum_{d\in D} Blank_{td} \times \max\Big\{w_1(\bar{b}_t - \sum_{p\in P} \dot{\boldsymbol{x}}_{\boldsymbol{ptd(j)}})\ ,\ w_2(\sum_{p\in P} \dot{\boldsymbol{x}}_{\boldsymbol{ptd(j)}} - \bar{b}_t)\Big\}\Big] + \max_{e\,\in\,E} \sum_{t\in T}\sum_{d\in D} \Delta_{td}\, e_t$$

and Δ_{td} changes to:

$$\Delta_{td} = Blank_{td}\Big[\max\Big\{w_1(\bar{b}_t + \hat{b}_t - \sum_{p\in P} \dot{\boldsymbol{x}}_{\boldsymbol{ptd(j)}})\ ,\ w_2(\sum_{p\in P} \dot{\boldsymbol{x}}_{\boldsymbol{ptd(j)}} - \bar{b}_t)\Big\} - \max\Big\{w_1(\bar{b}_t - \sum_{p\in P} \dot{\boldsymbol{x}}_{\boldsymbol{ptd(j)}})\ ,\ w_2(\sum_{p\in P} \dot{\boldsymbol{x}}_{\boldsymbol{ptd(j)}} - \bar{b}_t)\Big\}\Big]$$

Now that we have obtained the final master problem and sub-problem, we can implement the complete algorithm. In the following chapter, we present some results and analyses from solving the problem at hand using the proposed algorithm.

Chapter 6

Results and Conclusion

In this chapter, we present the results obtained from solving the two-stage robust model for the multi-objective problem faced by the Pathology Department at TOH. The model and the Benders-dual algorithm were coded in the IBM ILOG Optimization Programming Language (OPL) and solved using CPLEX 12.9.0 on a Intel(R) Core(TM) i7-8550U computer with 8.00 GB RAM. The model is tested with real deterministic data with a 30% variation in demand. The analysis of the results is divided into two main sections. First, we compare the schedule obtained from the deterministic model with the schedule from its adjustable robust counterpart regarding the number of missed-assignments. Then, we investigate the effect of model's robustness on the problem's objective function value and the other goals sought by the Pathology Department.

6.1 Model adjustments

The deterministic model for the pathology problem includes six main objectives as mentioned earlier in Chapter 5. When model (5.1) was modified to problem (5.2), the computation time to solve the MIP significantly increased. Some efforts were taken to improve the execution time, but they mostly were unsuccessful. Nevertheless, they are worth mentioning:

- The model was ill-conditioned and numerically unstable at first due to including float variables in the model. The solution of an ill-conditioned problem is usually unreliable and can change drastically with insignificant modifications in the problem data, as well as slow down the solution process. To resolve this issue, we tried lowering the constraint bounds and the values of the coefficients in the problem, as well as adjusting several CPLEX parameters in order to customize the way the CPLEX branch and bound algorithm operates [45]. While some of the CPLEX setting parameters were successful at reducing the numerical instability, the issue of slow performance persisted.

- We tried a lexicographic approach where a multi-objective problem is solved based on a pre-specified hierarchy rather than a weighting scheme. This approach resulted in fast few first iterations, but the rest of the iterations were still slow.

- We tried CPLEX's automatic tuning tool that prompts CPLEX to run tuning tests with different parameter settings. This effort was also not successful as the tool's suggestion was the default setting.

In the context lexicographic approach, it is often possible to use a hybrid method where a weighting scheme and a decision hierarchy are combined. In that case, instead of optimizing each individual objective in a separate iteration, we optimize a weighted combination of the problem's objectives in two or three iterations.

While applying this hybrid method to the pathology problem, we determined that some specific combinations of the problem's goals made the solution process slower. After trying different combinations, we decided to remove the back-to-back and the pathologist's rotations metrics as their combination with the problem's main objectives increased the computation time. We kept four of the problem's main goals and based our analyses on this smaller version of the problem. This resulted in successfully reducing the computation time of the deterministic model to less than two minutes. The objectives that were preserved are demand coverage, utilization, pathologist's workload, and schedule consistency.

6.2 Convergence of the algorithm

Before making any comparison between the deterministic and the robust model, we need to make sure that the applied algorithm functions correctly and converges in a finite number of iterations. Recall that the model's robustness can be adjusted by changing the budget of uncertainty (Γ), which is a parameter for defining the size of the uncertainty. In Table 6.1 we provide a summary of solving different robust models, each representing a level of robustness based on different Γ values. The optimality gap for the stopping criteria is set to 3%, and the maximum execution time is set to 300 minutes (5 hours); meaning if the gap has not reached within 5 hours, the algorithm will stop.

Table 6.1: Performance of the Benders dual algorithm

Metric \ Γ	3	4	5	6	7	10	12
Number of iterations	35	43	52	98	130	110	77
Optimality gap	3%	3%	3%	3%	10%	29%	40%
Time (m)	35	44	70	177	>300	>300	>300

In addition, figure 6.1 provides the values of lower and upper bounds in each iteration when $\Gamma = 5$.

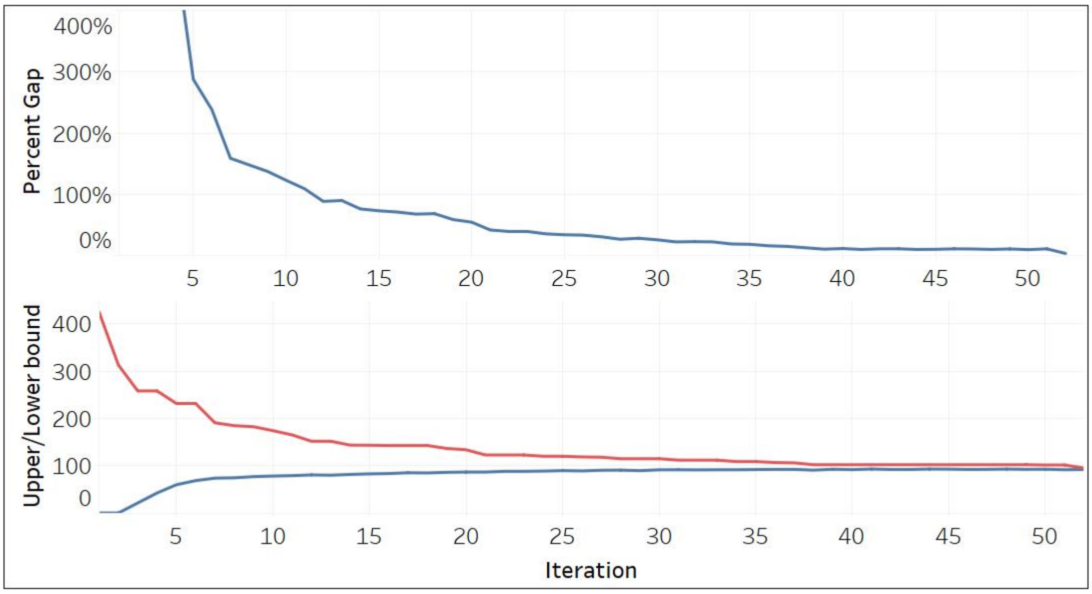

Figure 6.1: Convergence of Benders algorithm when $\Gamma = 5$

We can see from Table 6.1 that as the budget of uncertainty (Γ) increases, the number of iterations needed to reach optimality and the computation time increase.

In Figure 6.1, we can observe that at first, the speed of convergence is fast and as the algorithm proceeds, the convergence rate gets slower. Recall that the difference between the upperbound and the lowerbound in each iteration is the difference between $\sum_d \lambda_d$ in the master problem and the value of the sub-problem's objective function. At first, this difference is substantial and as the algorithm continues generating optimality cuts, this gap gets smaller.

In the next section, we analyze the performance of different robust models (based on different budgets of uncertainty) regarding the number of missed-assignments and the model's objective function value.

6.3 Model performance

In order to assess each robust model's policy, we generated 100 5-week samples of the demand for each task, assuming each task's demand follows a triangular distribution. The distribution is bounded between the demand's lower and upper bounds $[\bar{b}_t\ ,\ \bar{b}_t + 30\%]$, with a mode of $\bar{b}_t$. The complete list of tasks (subspecialties) and their bounds are provided in Appendix B. (Since the demand intervals are small and changes are not substantial in different replications, it suffices that we test on 100 samples.)

After solving robust models (based on different Γ values) and obtaining the corresponding schedule, we can compare the amount of missed assignments for each model. In Table 6.2, we summarize the average percentage of unmet demand and its 95% confidence interval for each model setting. This metric represents the percentage of demand which is not met over all of the tasks for a 5-week time span. Furthermore, we compare the percentage of unmet demand for each robust schedule with the percent of unmet demand for the deterministic schedule and conduct a paired t-test to show the improvement is statistically significant. The D in the table represents the deterministic model.

Table 6.2: Comparing the model performance for a 5-week schedule

Γ	Avg % of unmet demand	95% CI	p-value
D	11.59%	[11.52 , 11.67]	-
3	11.02%	[10.94 , 11.09]	$\leq$ 2.2e−16
4	10.30%	[10.22 , 10.37]	$\leq$ 2.2e−16
5	9.23%	[9.16 , 9.31]	$\leq$ 2.2e−16
6	8.72%	[8.65 , 8.80]	$\leq$ 2.2e−16
7	8.82%	[8.75 , 8.89]	$\leq$ 2.2e−16
10	7.28%	[7.21 , 7.35]	$\leq$ 2.2e−16
12	7.78%	[7.72 , 7.85]	$\leq$ 2.2e−16

As expected, by increasing the level of conservatism, the model's ability to cover the demand and lower the amount of missed assignments also increases. Also, the p-values corresponding to the comparisons between each robust model and the deterministic case show that there is a decrease in the level of unmet demand as a consequence of considering demand uncertainty in the robust model.

Figure 6.2 shows the average percentage of unmet demand with a 95% confidence interval for the different model settings. The values are based on Table 6.2.

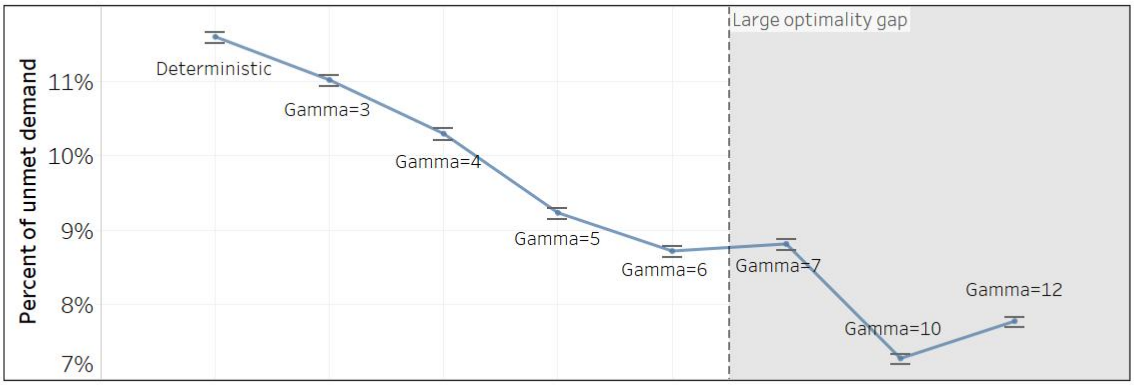

Figure 6.2: Number of missed assignments for different models

Again, we can observe that with an increase in the budget of uncertainty, the amount of missed assignments decreases. This decreasing trend should have a monotonous behavior as Γ enlarges, similar to what happens until $\Gamma = 6$. However, when Γ is greater than 6, the models do not reach to optimality and, consequently, the amount of missed assignment does not decrease strictly.

To better understand the differences in the resulting schedules, we compare the amount of assignment to the different tasks (subspecialties) created by each schedule. Since there are 21 subspecialties and 8 schedules, and summarizing the values in a table may not show changes in the best way, we represent these assignments in figures 6.3 and 6.4.

In Figure 6.3, the x and y axes represent the subspecialties and schedules, respectively, and the z axis shows the total amount of assignment. For a better visualization, subspecialties are sorted from largest to smallest based on their assignment values. Each wall (area) in the chart represents the total assignments in a sub-specialty, and the slope of the wall (area) represents changes in the assignments when going from a deterministic model to a robust model. For example, the two sub-specialties, CYTO (Cytology) and GU (Genitourinary), have the largest changes in the amount

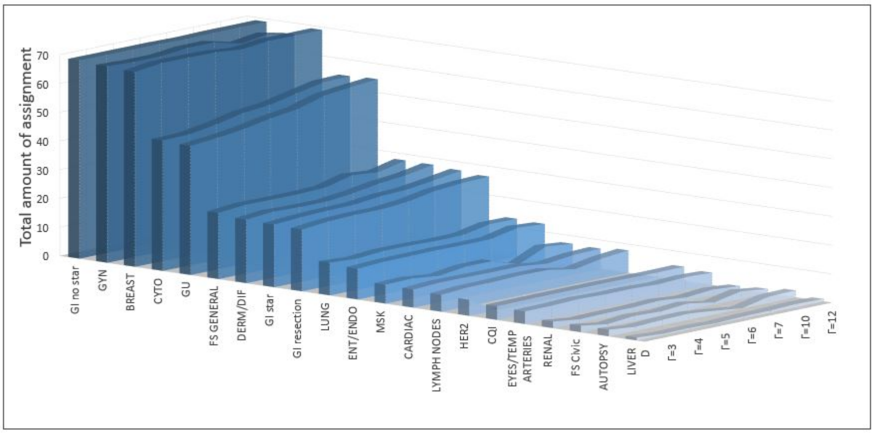

Figure 6.3: Total amount of assignment for each task (subspecialty) and each model setting

of assignment when moving from the deterministic to the robust model with the highest budget of uncertainty.

Figure 6.4 also demonstrates changes in the assignments for each individual subspecialty across schedules. In this radar graph, each subspecialty is given an individual axis and axes are placed radially around the centre. The amount of assignment for each subspecialty is shown with a marker on the subspecialty's axis, and the markers for each schedule are connected with a line.

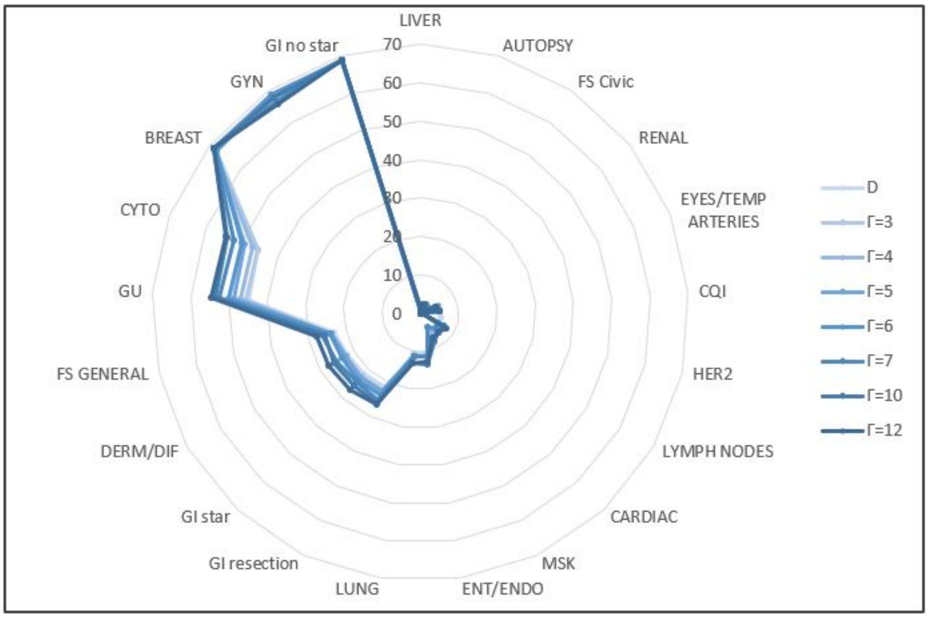

Figure 6.4: Total amount of assignment for each task (subspecialty) and model setting

Since we want to compare different schedules, each individual axis contains eight markers (one for each schedule) with different colors. Darker markers represent schedules with more uncertainty coverage, and lighter markers represent schedules with less uncertainty coverage. For example, we can see that the amount of assignment for subspecialties between CYTO (Cytology) and LUNG (Pulmonary) change the most when moving from a schedule with less coverage to a schedule with more coverage.

6.4 The cost of robustness

Increasing the demand coverage and model robustness can be expensive and result in higher objective values. In a multi-objective model, this can be perceived as a consequence of larger deviations from other targets. In this section, we first compare the value of the objective function as the model's robustness increases and then we compare each objective of the pathology problem individually across model settings.

6.4.1 Robustness and the objective function

Table 6.3 shows the value of the objective function corresponding to each robust model setting. We can see that by increasing the robustness, the objective value increases accordingly.

Table 6.3: Comparing the value of the objective function for different models

Metric \ Γ	3	4	5	6	7	10	12
Objective value	35272	42741	46021	49740	50966	48110	46121
% increase from $\Gamma = 3$	-	21.2%	30.5%	41.0%	44.5%	36.4%	30.8%
$\sum_d \lambda_d$	67.35	79.92	87.18	94.92	98.48	91.16	86.84

Figure 6.5 shows the relationship between changes in the budget of uncertainty and changes in the value of the objective function. We can see that we have an increasing trend in the values of the objective function as Γ increases for values of Γ smaller than 7. But when $\Gamma = 10$ or $\Gamma = 12$, the model cannot be solved to optimality and the value of the lower bound which is the model's objective function is still significantly less than its upper bound. This causes a drop in the value of the objective function when changing the budget of uncertainty from $\Gamma = 7$ to $\Gamma = 10$ or $\Gamma = 12$.

The reason for not including the deterministic model in Table 6.3 and Figure 6.5 is that the objective function of the deterministic and the robust model are not completely similar. The deterministic objective function minimizes missed assignments

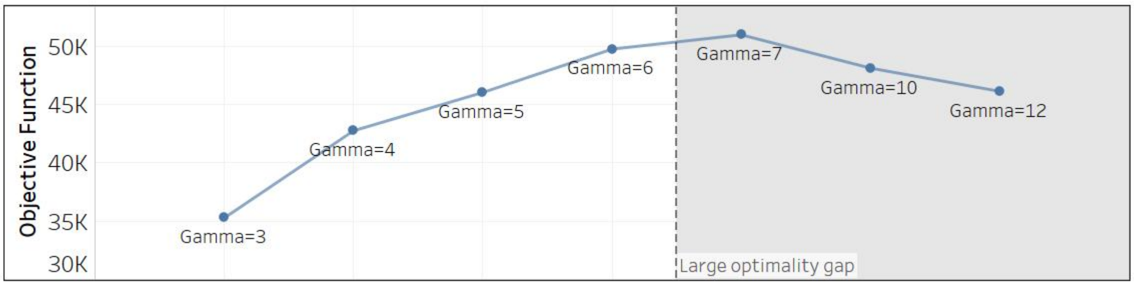

Figure 6.5: Objective values for different models

based on nominal demand, whereas the robust models minimize missed assignments based on worst case scenarios. In other words, when Constraint (4.2) in the deterministic model is substituted with the Benders cuts, the h_{ktd} which is the unmet demand is substituted with λ_d in the objective function. To compare the deterministic schedule with robust schedules, we would need to have a more detailed analysis where we can see the effect of robustness on each of the problem's goals individually.

6.4.2 Robustness and problem goals

When comparing the achievement of problem's goals across model settings, we should notice that the main purpose of a multi-objective optimization model is to minimize the summation of the weighted metrics, each corresponding to one of the problem's goals. Based on the different data and at each iteration, the optimization model may choose smaller or larger values for each individual metric in order to maintain a lower objective function value. Although we cannot expect consistency for the values associated with each individual goal, we believe that on average the values will be affected if there is any relationship between the demand coverage and that specific variable. Recall that there are three objectives other than demand coverage in our model: utilization, pathologist's workload, and schedule consistency. To analyze the relationship between coverage and each of these goals, we took the average of the last five iterations for each individual goal and model setting which are demonstrated in Figure 6.6.

The metrics corresponding to the three goals mentioned earlier are daily-overload, schedule inconsistency, and under-utilization. Daily overload represents how many

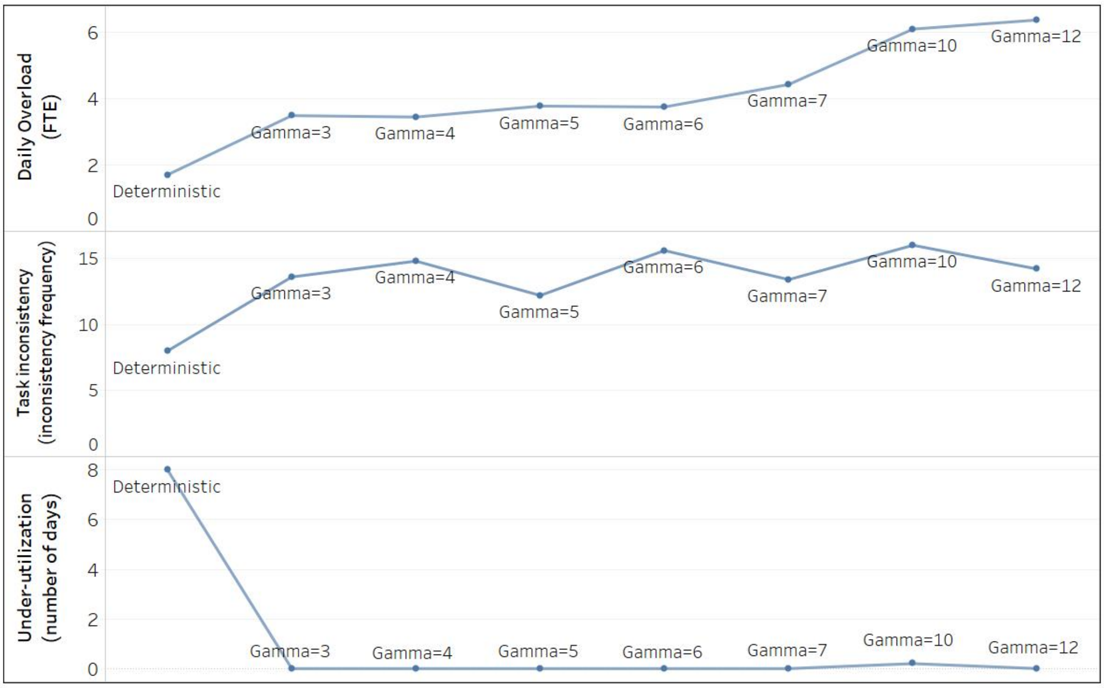

Figure 6.6: Effect of robustness on other problem metrics

pathologists are given daily assignments that is more than 1.1 FTE. Task inconsistency represents how many times pathologists' subspecialties changed throughout the week. Finally, under-utilization represents how many days a pathologist is not assigned to any subspecialty while s/he was available. In Figure 6.6, we can observe how these variables change across different model settings (Refer to Appendix A for each metric's mathematical formulation).

As the budget of uncertainty increases, the daily overload increases. This is expected since by increasing the demand coverage pathologists are assigned to more tasks.

We can also see that the value for under-utilization is 8 for the deterministic model, whereas we have almost no under-utilization in any of the robust model settings.

The task-inconsistency does not show any regular behavior while increasing the demand coverage, but we can see that, in general, the robust model settings have more inconsistent schedules than the deterministic one. This can happen because the consistency of a schedule does not necessarily have a strong relationship with the schedule's coverage policy. Also when the demand level is high, maintaining a consistent schedule is more difficult.

Now that we have the values for each individual goal, we can define a modified objective value where we replace the values of $\sum_d \lambda_d$ in the robust objective function and h_{ktd} (which is the unmet demand) in the deterministic objective function with the average unmet demand over the 100 replications mentioned earlier. As a result, we can compare the modified objective values of the deterministic model with those of the robust settings which are summarized in Table 6.4.

Table 6.4: Modified objective values

Metric \ Γ	D	3	4	5	6	7	10	12
Underutilization	8	0	0	0	0	0	0.2	0
Daily overload	1.7	3.5	3.4	3.8	3.7	4.4	6.1	6.4
Inconsistency	8	13.6	14.8	12.2	15.6	13.4	16	14.2
Unmet demand	58.7	55.0	51.4	46.1	43.5	44.0	36.3	38.8
Objective value	31919	29196	27515	24642	23700	23785	20421	21476

The modified objective values are calculated based on the exact weights from the models' real objective functions. Because the demand coverage is the most important goal to the problem and has the largest weight, and because changes in the values associated with the other goals are less significant, Figure 6.7 is very similar to Figure 6.2. It shows that when the budget of uncertainty increases, the schedule improves.

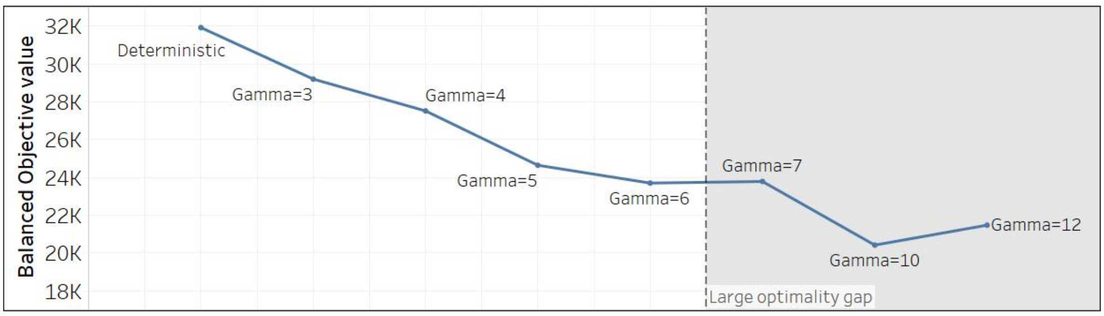

Figure 6.7: Modified objective values for the different model settings

6.5 Conclusion

In this chapter, we conclude that the proposed model is successful at reducing unmet demand and thus increasing demand coverage for the problem faced by the Pathology Department. In many problems, the analysis of robustness in demand coverage is coupled with the schedule's cost and what is of interest is the trade-off between the demand coverage and the schedule's cost. But when the model is multi-objective and we analyze multiple goals instead of only cost, it is a challenge to figure out how much to "pay" to have the desired level of robustness. In the pathology model, the objective function includes only the penalties from the model's soft constraints. Since the coverage is by far the most important goal for this problem and because the other objectives do not change significantly with the changes in coverage, increasing the model's robustness improves the problem's objective function. As the budget of uncertainty increases, the value of the problem's objective function decreases.

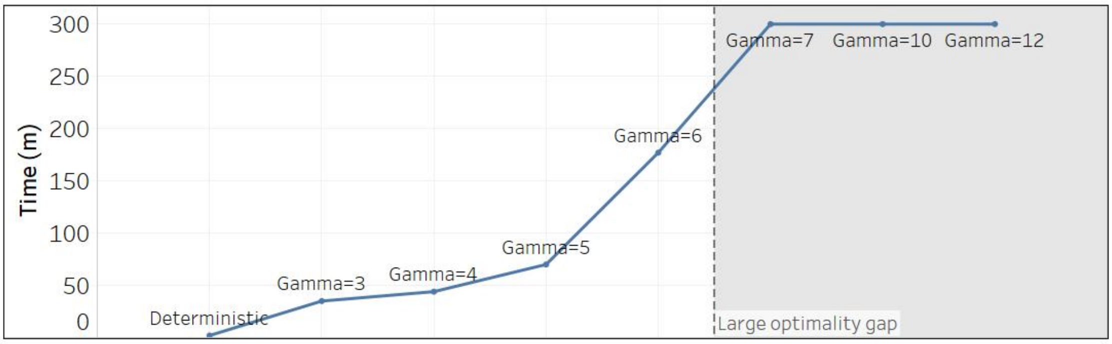

Figure 6.8: Solution time (minutes)

Conversely, as the budget of uncertainty increases, the solution time at each iteration and the number of iterations needed by the algorithm also increase. Figure 6.8 demonstrates the solution time for the different model settings in minutes (Recall that the algorithm was stopped after 300 minutes of execution time regardless of the reached gap)

Although it is the decision-maker's choice to select a budget of uncertainty based on the model's performance and solution time, we believe that when $\Gamma = 6$ the model provides sufficient robustness while being able to produce a solution in a reasonable amount of time. Figure 6.9 shows the convergence of the Benders algorithm when the budget of uncertainty is 6.

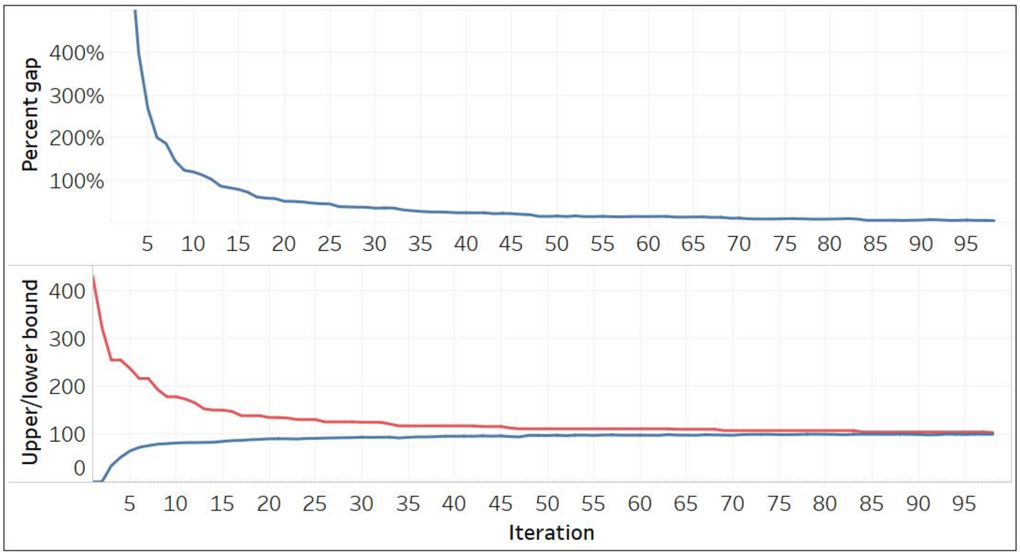

Figure 6.9: Convergence of Benders algorithm when $\Gamma = 6$

6.6 Limitations

The main limitation of this research is the solution time of the model for the pathology problem. Because of the model's performance and the slow MIP cuts, we had no choice but to remove two of the problem's goals from the objective function in order to accelerate the solution process in each iteration of Benders algorithm. As future work, it is possible to make modifications to the robust model to make the solution process faster. One could change the structure of the row generation algorithm and incorporate feasibility cuts to the algorithm so as to solve the first deterministic problem in model (5.1) with binary assignments, or try different cutting plane algorithms in order to find another approach with better performance, such as column-and-constraint generation [22].

The next limitation concerns the uncertainty set structure. In spite of applying the robust model to a problem with real deterministic data, we assumed that the demand in each task can deviate by up to 30% of its nominal value and that this variation creates a one-sided uncertainty set. For a more accurate model and uncertainty set, we need to have some information about the demand for each task. Furthermore, in our uncertainty set, all tasks can deviate from their nominal values, whereas with distributional data, we could remove some tasks from the uncertainty set due to small variations.

Bibliography

[1] A. Ben-Tal and A. Nemirovski, "Robust solutions of linear programming problems contaminated with uncertain data," *Mathematical programming*, vol. 88, no. 3, pp. 411–424, 2000.

[2] A. T. Ernst, H. Jiang, M. Krishnamoorthy, and D. Sier, "Staff scheduling and rostering: A review of applications, methods and models," *European journal of operational research*, vol. 153, no. 1, pp. 3–27, 2004.

[3] J. M. Tien and A. Kamiyama, "On manpower scheduling algorithms," *Siam Review*, vol. 24, no. 3, pp. 275–287, 1982.

[4] J. Van den Bergh, J. Beliën, P. De Bruecker, E. Demeulemeester, and L. De Boeck, "Personnel scheduling: A literature review," *European journal of operational research*, vol. 226, no. 3, pp. 367–385, 2013.

[5] D. P. Morton and E. Popova, "A bayesian stochastic programming approach to an employee scheduling problem," *Iie Transactions*, vol. 36, no. 2, pp. 155–167, 2004.

[6] A. Parisio and C. N. Jones, "A two-stage stochastic programming approach to employee scheduling in retail outlets with uncertain demand," *Omega*, vol. 53, pp. 97–103, 2015.

[7] P. Punnakitikashem, J. M. Rosenberber, and D. F. Buckley-Behan, "A stochastic programming approach for integrated nurse staffing and assignment," *IIE Transactions*, vol. 45, no. 10, pp. 1059–1076, 2013.

[8] A. L. Soyster, "Convex programming with set-inclusive constraints and applications to inexact linear programming," *Operations research*, vol. 21, no. 5, pp. 1154–1157, 1973.

[9] D. Bertsimas, D. B. Brown, and C. Caramanis, "Theory and applications of robust optimization," *SIAM review*, vol. 53, no. 3, pp. 464–501, 2011.

[10] A. Ben-Tal and A. Nemirovski, "Robust solutions of uncertain linear programs," *Operations research letters*, vol. 25, no. 1, pp. 1–13, 1999.

[11] D. Bertsimas, D. Pachamanova, and M. Sim, "Robust linear optimization under general norms," *Operations Research Letters*, vol. 32, no. 6, pp. 510–516, 2004.

[12] P.-S. Chen, Y.-J. Lin, and N.-C. Peng, "A two-stage method to determine the allocation and scheduling of medical staff in uncertain environments," *Computers & Industrial Engineering*, vol. 99, pp. 174–188, 2016.

[13] L. Zhen, "Task assignment under uncertainty: stochastic programming and robust optimisation approaches," *International Journal of Production Research*, vol. 53, no. 5, pp. 1487–1502, 2015.

[14] G. Carello and E. Lanzarone, "A cardinality-constrained robust model for the assignment problem in home care services," *European Journal of Operational Research*, vol. 236, no. 2, pp. 748–762, 2014.

[15] S. Liao, G. Koole, C. Van Delft, and O. Jouini, "Staffing a call center with uncertain non-stationary arrival rate and flexibility," *OR spectrum*, vol. 34, no. 3, pp. 691–721, 2012.

[16] J. Tang and Y. Wang, "An adjustable robust optimisation method for elective and emergency surgery capacity allocation with demand uncertainty," *International Journal of Production Research*, vol. 53, no. 24, pp. 7317–7328, 2015.

[17] M. Minoux, "On 2-stage robust lp with rhs uncertainty: complexity results and applications," *Journal of Global Optimization*, vol. 49, no. 3, pp. 521–537, 2011.

[18] A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski, "Adjustable robust solutions of uncertain linear programs," *Mathematical Programming*, vol. 99, no. 2, pp. 351–376, 2004.

[19] İ. Yanıkoğlu, B. Gorissen, and D. den Hertog, "Adjustable robust optimization–a survey and tutorial," *Available online at ResearchGate*, 2017.

[20] S. Mattia, F. Rossi, M. Servilio, and S. Smriglio, "Staffing and scheduling flexible call centers by two-stage robust optimization," *Omega*, vol. 72, pp. 25–37, 2017.

[21] S. Neyshabouri and B. P. Berg, "Two-stage robust optimization approach to elective surgery and downstream capacity planning," *European Journal of Operational Research*, vol. 260, no. 1, pp. 21–40, 2017.

[22] B. Zeng and L. Zhao, "Solving two-stage robust optimization problems using a column-and-constraint generation method," *Operations Research Letters*, vol. 41, no. 5, pp. 457–461, 2013.

[23] D. Bienstock and N. ÖZbay, "Computing robust basestock levels," *Discrete Optimization*, vol. 5, no. 2, pp. 389–414, 2008.

[24] B. L. Gorissen and D. Den Hertog, "Robust counterparts of inequalities containing sums of maxima of linear functions," *European Journal of Operational Research*, vol. 227, no. 1, pp. 30–43, 2013.

[25] A. Thiele, T. Terry, and M. Epelman, "Robust linear optimization with recourse," *Rapport technique*, pp. 4–37, 2009.

[26] V. Gabrel, M. Lacroix, C. Murat, and N. Remli, "Robust location transportation problems under uncertain demands," *Discrete Applied Mathematics*, vol. 164, pp. 100–111, 2014.

[27] J. E. Kelley, Jr, "The cutting-plane method for solving convex programs," *Journal of the society for Industrial and Applied Mathematics*, vol. 8, no. 4, pp. 703–712, 1960.

[28] D. Bertsimas, E. Litvinov, X. A. Sun, J. Zhao, and T. Zheng, "Adaptive robust optimization for the security constrained unit commitment problem," *IEEE transactions on power systems*, vol. 28, no. 1, pp. 52–63, 2012.

[29] R. Jiang, M. Zhang, G. Li, and Y. Guan, "Benders' decomposition for the two-stage security constrained robust unit commitment problem," in *IIE Annual Conference. Proceedings*, p. 1, Institute of Industrial and Systems Engineers (IISE), 2012.

[30] B. Du, H. Zhou, and R. Leus, "A two-stage robust model for a reliable p-center facility location problem," *Applied Mathematical Modelling*, vol. 77, pp. 99–114, 2020.

[31] T. L. Terry, *Robust linear optimization with recourse: Solution methods and other properties*. PhD thesis, University of Michigan, 2009.

[32] E. K. Burke, P. De Causmaecker, G. V. Berghe, and H. Van Landeghem, "The state of the art of nurse rostering," *Journal of scheduling*, vol. 7, no. 6, pp. 441–499, 2004.

[33] S. Topaloglu and I. Ozkarahan, "An implicit goal programming model for the tour scheduling problem considering the employee work preferences," *Annals of Operations Research*, vol. 128, no. 1-4, pp. 135–158, 2004.

[34] S. Topaloglu, "A multi-objective programming model for scheduling emergency medicine residents," *Computers & Industrial Engineering*, vol. 51, no. 3, pp. 375–388, 2006.

[35] J. Li, E. K. Burke, T. Curtois, S. Petrovic, and R. Qu, "The falling tide algorithm: a new multi-objective approach for complex workforce scheduling," *Omega*, vol. 40, no. 3, pp. 283–293, 2012.

[36] M. N. Azaiez and S. S. Al Sharif, "A 0-1 goal programming model for nurse scheduling," *Computers & Operations Research*, vol. 32, no. 3, pp. 491–507, 2005.

[37] S. Mattia, F. Rossi, M. Servilio, and S. Smriglio, "Robust shift scheduling in call centers," in *International Symposium on Combinatorial Optimization*, pp. 336–346, Springer, 2014.

[38] D. Bertsimas and M. Sim, "Robust discrete optimization and network flows," *Mathematical programming*, vol. 98, no. 1-3, pp. 49–71, 2003.

[39] D. Bertsimas and A. Thiele, "Robust and data-driven optimization: modern decision making under uncertainty," in *Models, methods, and applications for innovative decision making*, pp. 95–122, INFORMS, 2006.

[40] M. Minoux, "Robust lp with right-handside uncertainty, duality and applications," 2007.

[41] B. Zeng, "Solving two-stage robust optimization problems by a constraint-and-column generation method," *University of South Florida, FL, Tech. Rep*, 2011.

[42] D. Bertsimas and M. Sim, "The price of robustness," *Operations research*, vol. 52, no. 1, pp. 35–53, 2004.

[43] A. Montazeri, J. Patrick, W. Michalowski, and D. Banerjee, "Developing the pathologists' monthly assignment schedule: a case study at the division of anatomical pathology of the ottawa hospital," in *AMIA Annual Symposium Proceedings*, vol. 2015, p. 933, American Medical Informatics Association, 2015.

[44] J. Patrick, A. Montazeri, W. Michalowski, and D. Banerjee, "Automated pathologist scheduling at the ottawa hospital," *Interfaces*, 2019.

[45] I. I. Cplex, "Ibm ilog cplex optimization studio cplex user's manual, version 12, release 6," 2013.

Appendix A

The master model

Here we present the model in Patrick et al. [44] with some minor modifications:

List of Parameters (part A)

Pathologists	Set of pathologists
Tests	Set of tests (sub-specialties)
Days	Set of days
W	Number of weeks in the planning horizon
Weeks	Set of weeks
Rotation	Binary matrix representing the sub-specialties in pathologist's regular rotations
UpperBound	Matrix representing the maximum number of days in a 4-week period that a pathologist can be assigned to a specialty
Blanks	Binary matrix for day/specialty combinations than needs to be filled
TabooPairings	Binary matrix representing combinations of sub-specialties that cannot be assigned to any pathologist
Availability	Binary matrix representing the availability of each pathologist on each day

$FixedFullAssignments$	Matrix for fixing some of the full-time assignments with a pathologist
$FTEfraction$	Matrix representing the weekly amount of required FTE for each pathologist
$ProtectedWeeks$	Vector giving a pathologist a number of "protected" weeks
$ProtectedDays$	Vector giving a pathologist a number of "protected" days
$WorkloadRestriction$	Vector providing upper bounds on the pathologists' assignments on any day
C	Objective goal weights
$MaxPathologists$	Vector for restricting the number of pathologists assigned to a sub-specialty in a week
$AllowableCombinations$	Set of combinations of sub-specialties for a pathologist that is admissible (can be assigned together).
$iteration$	number of iterations in Benders algorithm
b	The required workload (demand) for each sub-specialtiy in each iteration
$InitialB$	The required workload (demand) for each sub-specialtiy in the first iteration

List of Decision Variables

X_{ptd}	$\in \mathbb{R}^+$	Pathologist's assignments
$OverUtilized_p$	$\in \mathbb{Z}^+$	Over-utilization of pathologist's hours
$UnderUtilized_p$	$\in \mathbb{Z}^+$	Under-utilization of pathologist's hours
$BackToBack_{ptw}$	$\in \{0,1\}$	Indicator for assigning the same task to a pathologist two weeks in a row
$DailyOverload_{pd}$	$\in \mathbb{R}^+$	Indicator for assigning more than 1.1 FTE to a pathologist
$EarlyStop_{ptd}$	$\in \{0,1\}$	Indicator for when a pathologist is finishing a specialty earlier than Friday
$LateStart_{ptd}$	$\in \{0,1\}$	Indicator for when a pathologist is starting a specialty later than Monday
$Over_{pt}$	$\in \mathbb{Z}^+$	Measures how much a pathologist is over their target for a sub-specialty
$Under_{pt}$	$\in \mathbb{Z}^+$	Measures how much a pathologist is under their target for a sub-specialty
$Penalty_{pt}$	$\in \mathbb{R}^+$	Penalizes when a pathologist is over their target for a specialty
$WeeklyAssignment_{ptw}$	$\in \{0,1\}$	Indicating if a pathologist is working on a sub-specialty in a week
$DayIndicator_{pd}$	$\in \{0,1\}$	Indicating if a pathologist is working on a day
$WeekIndicator_{pw}$	$\in \{0,1\}$	Indicating if a pathologist is working in a week
$TaskIndicator_{ptd}$	$\in \{0,1\}$	Indicating if a pathologist is working on a day in a sub-specialty
λ_d	$\in \mathbb{R}^+$	Maximum amount of unmet demand on a day for all sub-specialties
bin_{ptd}	$\in \{0,1\}$	

List of Parameters (part B)

Number of available days for each pathologist throughout the planning horizon

$$AvailableDays_p = \sum_{w} \min\Bigg(\Big\lceil FTEfraction_{pd} \times \sum_{d\,in\,w} Blanks_{(BREAST)d}\Big\rceil \,, \sum_{d\,in\,w} Availability_{pd} \mid Blanks_{(BREAST)d} = 1\Bigg)$$

Number of available weeks for each pathologist throughout the planning horizon

$$AvailableWeeks_{pw} = \max\Bigg(0\,,\,\Big\lfloor \min\Big(\frac{\sum_{d\,in\,w} Availability_{pd}}{4}\,,\,\frac{\sum_{d\,in\,w} Blanks_{(BREAST)d}}{4}\Big)\Big\rfloor\Bigg)$$

Pathologist's workload target in each specialty throughout the planning horizon

$$Target_{pt} = \lceil\frac{\sum_d Blanks_{td}}{W}\rceil \times Rotation_{pt} \times \min\Bigg[\max\Bigg(1\,,\,\Big\lfloor\frac{\sum_d \lceil InitialB_{td}\rceil \times Blanks_{td}}{\max(1, 5 \times \sum_{p1} Rotation_{p_1 t})}\Big\rfloor\Bigg)\,,\,\max\Bigg(1\,,\,\Big\lfloor\frac{AvailableDays_p}{5 \times \max(1, \sum_t Rotation_{p,t1})}\Big\rfloor\Bigg)\Bigg]$$

Objective Function

$$
\begin{aligned}
\min \Big[& C(1) \times \sum_{d} \lambda_d \\
& + C(2) \times \sum_{p} UnderUtilized_p \\
& + C(3) \times \sum_{p,t,w} BacktoBack_{ptw} \\
& + C(4) \times \sum_{p,d} Dailyoverload_{pd} \\
& + C(5) \times \sum_{p,t,d} EarlyStop_{ptd} + LateStart_{ptd} \\
& + C(6) \times \sum_{p,t} Rotation_{pt} \times (Under_{pt} + Penalty_{pt}) \Big]
\end{aligned}
$$

Model constraints:

First objective: Demand coverage (Benders cut constraints)

Tracks the maximum amount of unmet demand under different iterations

$$\lambda_d \geq \sum_t Blanks_{td}\Big(InitialB_t - \sum_p x_{ptd}\Big) \qquad \forall d$$

$$\lambda_d \geq \sum_t Blanks_{td} \times y_{itd}\Big(b_{it} - \sum_p x_{ptd}\Big) \qquad \forall d, i$$

Second objective: Utilization

Keeps track of a pathologist's workload during the planning horizon

$$x_{ptd} \leq 10 \times TaskIndicator_{ptd} \qquad \forall p, t, d$$

$$10 \times TaskIndicator_{ptd} \leq x_{ptd} \qquad \forall p, t, d$$

$$\sum_t TaskIndicator_{ptd} \leq 3 \times DayIndicator_{pd} \qquad \forall p, d$$

$$DayIndicator_{pd} \leq \sum_t TaskIndicator_{ptd} \qquad \forall p$$

$$\sum_d DayIndicator_{pd} + UnderUtilized_p - OverUtilized_p =$$

$$AvailableDays_p - 5 \times ProtectedWeeks_p - ProtectedDays_p \qquad \forall p$$

Third objective: Back-to-back workload

Tracks occasions where the same test is given to the same pathologist two weeks in a row

$$\sum_{d\,in\,w} TaskIndicator_{ptd} \leq 5 \times WeeklyAssignment_{ptw} \qquad \forall p, t, w$$

$$WeeklyAssignment_{ptw} \leq \sum_{d\,in\,w} TaskIndicator_{ptd} \qquad \forall p, t, w$$

$$WeeklyAssignment_{ptw} + WeeklyAssignment_{ptw+1} \leq BacktoBack_{ptw} + 1 \quad \forall p, t, w$$

Fourth objective: Pathologist's workload

Penalizes when a pathologist is given more than a full-time-equivalent (FTE) amount of workload on a day

$$\sum_t x_{ptd} - DailyOverload_{pd} \leq 1.1 \qquad \forall p, d$$

Fifth objective: Schedule consistency

Tracks the consistency of the tasks assigned to a pathologist throughout a week

$$if \quad Availability_{p(d+1)} = 1$$

$$Blanks_{t(d+1)} \left(TaskIndicator_{ptd} - TaskIndicator_{pt(d+1)}\right) \leq EarlyStop_{ptd} \qquad \forall p, t, w, d \; in \; w$$

$$if \quad Availability_{pd} = 1$$

$$Blanks_{td} \times (TaskIndicator_{pt(d+1)} - TaskIndicator_{ptd}) \leq LateStart_{pt(d+1)} \qquad \forall p, t, w, d \; in \; w$$

Sixth objective: Pathologist's rotation

These constraints allow the model to penalize policies that don't rotate through a pathologist's rotation of specialties

$$if \quad Rotation_{pt} \geq 1$$

$$\sum_d TaskIndictor_{ptd} - Over_{pt} + Under_{pt} = Target_{pt} \qquad \forall p, t$$

$$if \quad Rotation_{pt} \geq 1 \qquad Over_{pt} \leq 5 + Penalty_{pt} \qquad \forall p, t$$

Hard constraints:

Assignments can have the maximum amount of 1 FTE and cannot be between 0 and 0.1 FTE.

$$x_{ptd} \leq 1 \qquad \forall p, t, d$$

$$x_{ptd} \geq 0.1 - bin_{ptd} \qquad \forall p, t, d$$

$$x_{ptd} \leq 0 + (1 - bin_{ptd}) \qquad \forall p, t, d$$

Ensures no pathologist is given more than 2 sub-specialties in a day regardless of the service weight

$$\sum_{t} TaskIndicator_{ptd} \leq 2 \qquad \forall p, d$$

Prevents pathologists from being given a pair of specialties on the same day unless that pair is admissible

$if \quad (p, t1, t2) \; in \; AllowableCombinantions$

$$x_{p(t1)d} + x_{p(t2)d} \leq WorkloadRestriction_p + 1 \qquad \forall p, d, t \quad t1 < t2$$

$else$

$$x_{p(t1)d} + x_{p(t2)d} \leq WorkloadRestriction_p \qquad \forall p, d, t \quad t1 < t2$$

Restricts the number of days a pathologist can be assigned to work in a week

$$\sum_{d \, in \, w} DayIndicator_{pd} \leq 5 \times FTEfraction_{pw} \qquad \forall p, w$$

Allows model to leave holidays and specific specialties unassigned as necessary

$if \quad Blanks_{td} = 0 \qquad \sum_{p} x_{ptd} = 0 \qquad \forall p$

Prevents pathologists from being assigned on days when they are unavailable

$if \quad Availability_{pd} = 0 \qquad \sum_{t} x_{ptd} = 0 \qquad \forall p, d$

Provides the option to fix certain full assignments

$$if \quad FixedFullAssignment_{td} = p \qquad x_{ptd} = 1 \qquad \forall p, t, d$$

Ensures that restrictions on the number of pathologists on each sub-specialty per week is respected

$$\sum_{p} WeeklyAssignment_{ptw} \leq MaxPathologists_{t} \qquad \forall t, w$$

Allows the user to specify an upper and lower bound on the number of days a pathologist can be given a specialty in a four week span

$$\sum_{d\,in\,w} TaskIndicator_{ptd} \leq UpperBound_{pt} \qquad \forall p, t, w$$

$$\sum_{d\,in\,w} TaskIndicator_{ptd} \geq LowerBound_{pt} \qquad \forall p, t, w$$

Allows the user to specify that a given pathologist must be given x number of weeks off

$$\sum_{t,d\,in\,w} TaskIndicator_{ptd} \leq 10 \times WeekIndicator_{pw} \qquad \forall p, t, d$$

$$WeekIndicator_{pw} \leq \sum_{t,d\,in\,w} TaskIndicator_{ptd} \qquad \forall p, w$$

$$\sum_{w} (WeekIndicator_{pw} \times AvailableWeeks_{pw}) \leq \sum_{w} AvailableWeeks_{pw} - ProtectedWeeks_{p} \qquad \forall p$$

Allows the user to specify that a given pathologist must be given x number of days off

$$\sum_{d} DayIndicator_{pd} \leq \sum_{d} Availability_{pd} - ProtectedDays_{p} \qquad \forall p$$

Specifies pairings of sub-specialties that are not allowed even if technically the workload would be fine

$$TaskIndicator_{p(t1)d} + TaskIndicator_{p(t2)d} \leq 1 + TabooPairing_{(t1)(t2)} \qquad \forall d, p, t1, t2 \qquad t1 < t2$$

Prevents from assigning a specialty to more than 3 pathologists

$$\sum_{p} TaskIndicator_{ptd} \leq 3 \qquad \forall t, d$$

Appendix B

Sub-specialty bounds

Each sub-specialty's demand take values in $[\bar{b}_t\ ,\ \bar{b}_t + \hat{b}_t]$ intervals represented in the following table.

Table B.1: Sub-specialty intervals in the uncertainty set

Number	Sub-specialty	Interval
1	AUTOPSY	[0.10 , 0.13]
2	BREAST	[3.00 , 3.90]
3	CARDIAC	[0.30 , 0.39]
4	CQI	[1.00 , 1.30]
5	CYTO	[2.00 , 2.60]
6	DERM/DIF	[1.00 , 1.30]
7	ENT/ENDO	[0.50 , 0.65]
8	EYES/TEMP ARTERIES	[0.20 , 0.26]
9	FS Civic	[0.10 , 0.13]
10	FS GENERAL	[1.00 , 1.30]
11	GI no star	[3.00 , 3.90]
12	GI resection	[1.00 , 1.30]
13	GI star	[1.00 , 1.30]
14	GU	[2.00 , 2.60]
15	GYN	[3.00 , 3.90]
16	HER2	[0.00 , 0.00]
17	LIVER	[0.10 , 0.13]
18	LUNG	[0.50 , 0.65]
19	LYMPH NODES	[0.30 , 0.39]
20	MSK	[0.30 , 0.39]
21	RENAL	[0.10 , 0.13]