# ABSTRACT

POOYA, PEGAH. An Optimization Framework To Improve Patient Safety In The Radiation Therapy Process . (Under the direction of Julie Ivy and Osman Ozaltin.)

Radiation therapy (RT) is an important part of cancer treatment with approximately 50% of cancer patients undergoing RT during the course of their treatment. The risk of error and the concern for patient safety is high in RT as the treatment planning and delivery include complex processes that involve numerous hand-offs among healthcare providers. Despite the use of quality assurance procedures in radiation oncology clinics, errors still occur and sometimes result in devastating consequences including fatal injuries. It is estimated that RT errors occur for up to 5% of patients and the rate of lethal errors is about 1/10,000 patients in the U.S. These numbers highlight the need for adopting a systematic approach to improve patient safety in the RT.

The use of safety barriers (SB) in RT is widely recognized for detecting potential human and non-human errors before they reach the patient. Despite the critical role of SBs in ensuring patient safety, very little research effort has been focused on the potential impact, cost or optimal implementation of SBs. While it may seem desirable to have as many safety barriers as possible to improve patient safety and process reliability, performing too many SBs prolongs the overall treatment time and cause extra workload that lead to additional errors.

In this dissertation, I first present a discrete-event simulation model to assess the reliability of the RT process. Using the insights from the simulation model, I develop an optimization framework to identify reliable SB configurations. Specifically, I develop mathematical models to determine the optimal position and composition of SBs taking into account SB reliability and costs (i.e., time to perform the checks in SBs). Analyzing different parameters of the problem reveals that SB failure probabilities are the most influential on the expected costs. Finally, I propose a two-stage stochastic programming model to capture uncertainties in the RT process. This model includes expected cost and a risk component to incorporate different risk preferences. The computational results illustrate how the consideration of risk affects the solution policies.

# An Optimization Framework To Improve Patient Safety In The Radiation Therapy Process

by
Pegah Pooya

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in final fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Industrial Engineering

Raleigh, North Carolina

2017

APPROVED BY:

| | |
|---|---|
| Julie Ivy<br>Co-chair of Advisory Committee | Osman Ozaltin<br>Co-chair of Advisory Committee |
| Maria Mayorga | Yahya Fathi |

Lukasz Mazur
External

## DEDICATION

To my loving parents, *Saeed and Shahnaz*,

To my dearest brother, *Navid*,

and

To my best friend, *Ali*

# BIOGRAPHY

Pegah Pooya was born in Arak, Iran. She received her B.Sc. degree in Industrial Engineering in 2011 at Iran University of Science and Technology in Tehran, Iran. She joined the Ph.D program in Industrial Engineering at North Carolina State University in August 2011. Her research interests include healthcare delivery systems modeling and optimization. She is currently working as an analytical consultant at SAS institute.

## ACKNOWLEDGEMENTS

First and foremost, I wish to thank my advisors Dr. Julie S. Ivy and Dr. Osman Ozaltin for all their support and guidance throughout my graduate studies. I am truly grateful to have you as my advisors. You have always been generous in sharing your valuable knowledge and wisdom which helped me learn and grow in many ways.

I also wish to thank the other members of my committee, Dr. Maria Mayorga, Dr. Yahya Fathi, and Dr. Lukasz Mazur for providing thoughtful suggestions on my research work and dissertation.

I would like to thank our collaborators, Dr. Lukasz Mazur, Dr. Lawrence Marks, Dr. Bhisham Chera, Kathy Burkhardt, Dr. Prithima Mosaly and Gregg Tracton from the Department of Radiation Oncology at University of North Carolina (UNC) Lineberger Comprehensive Cancer Center for their support and invaluable input to this research. Without their support, it would not have been possible to conduct this research. In particular, I am grateful to Dr. Lukasz Mazur for being a great mentor and for his constant enthusiasm and encouragement.

I would like to thank my friends and fellow graduate students at North Carolina State University for their support and friendship. I am so lucky to have known you.

Last but not least, I would like to express my deepest gratitude to all my family for their unconditional love and support, and to my husband Ali for always encouraging me and believing in me every step of the way.

# TABLE OF CONTENTS

## LIST OF TABLES

# LIST OF FIGURES

# Chapter 1

# Introduction

## 1.1 Patient Safety in Healthcare

Patient safety and the quality of care have become major concerns for healthcare providers in United States. The landmark report published by the Institute of Medicine (IOM) in 1999, "To Err is Human: Building a Safer Health System", revealed that each year between 44,000 to 98,000 people die in the U.S. because of preventable medical errors which is equivalent to a large airplane crash every day in a year [22, 32, 65]. The IOM report highlighted the need to acknowledge the epidemic of medical errors as a national health problem that has to be managed in a manner similar to other epidemics like heart disease, diabetes and obesity [93]. This need is further emphasized by a recent study which estimated that medical errors kill more than 250,000 Americans every year [77]- surpassing the CDC's (Centers for Disease Control and Prevention) stated third-leading cause of death in the U.S. (i.e. respiratory diseases) [26].

The IOM defines medical error as "the failure of a planned action to be completed as intended or the use of a wrong plan to achieve an aim" [65]. While major medical errors occur primarily in operating rooms, emergency departments and intensive care units, patients receiving care at any healthcare institution are exposed to medical errors [65]. These errors can result in adverse drug events, improper transfusions, surgical injuries, hospital acquired infections and other serious injuries leading to disability or death [45, 65]. In addition to the enormous cost of human life and health, preventable medical errors are costly in many other ways resulting in total cost of between $17 billion and $29 billion per year in hospitals across the US [65]. The

indirect cost associated with medical errors includes but is not limited to the cost of patient and provider dissatisfaction with the healthcare system. While patients suffer from physical or mental health consequences, health professionals may struggle with feelings of frustration and demoralization associated with being unable to provide patients with safe and efficient care [65].

A variety of factors have been shown to contribute to medical errors and affect the quality of care [69, 45]. For example, scientific and technological advances bring with them challenges associated with translating these advances into practice. According to Lee and Mongan [69] advances in medical science and medical technologies are major contributors to the current chaos in today's health care system. The increasing number of diagnostic procedures and treatment options have increased the risk of medical errors and thus many patients do not receive optimal care [45].

New diagnostic and treatment procedures require the involvement of multiple physicians and healthcare professionals. However, the healthcare delivery system does not always facilitate effective communication and teamwork among healthcare providers [69]. Physicians often work in isolation while "coordinated care" for their patients should require them to interact with their colleagues. It is more likely that errors occur when patients see multiple providers and none of them are able to access the patient's complete health information [65].

Although human errors are unavoidable, the majority of medical errors are attributable to flawed systems and processes, not to individuals [65]. The frequency and consequences of medical errors can be mitigated by designing safer healthcare systems. In particular, the quality/safety assurance procedures followed during treatment delivery can be optimally designed via data-driven models that are informed by past errors [45]. Such computer-based models will help reduce risks associated with actual implementations, e.g., trials leading to errors and harm, by providing a virtual environment for experimenting with quality assurance alternatives.

## 1.2 Patient Safety in Radiation Therapy

Radiation Therapy (RT) is one of the most complex settings in healthcare [10]. RT is an important part of cancer treatment with approximately 50% of cancer patients (740,000 patients

in the U.S. per year) undergoing radiation therapy during the course of their treatment [53]. RT professionals design and deliver high-energy photon beams to cancer targets while minimizing risk to adjacent healthy tissue. The importance of patient safety and quality assurance has long been recognized in RT as it involves high-risk complex medical procedures [10, 55, 128]. However, the continuous adoption of rapidly-evolving health information technologies (health IT), increasing workloads on providers, and the lack of effective care coordination have recently caused increased quality/safety concerns in RT [32, 48, 128, 107].

In the past several years, there has been negative media coverage of RT errors. A front page New York Times article in 2010 spotlighted RT errors with the potential to harm patients and raised concerns about patient safety in the RT process [38]. The investigation by the New York Times involved 621 cases of RT errors that occurred in New York state from 2001 to 2009. The authors suggested that there may be a substantial bias towards the underestimation of the actual number of errors, because the research was mostly based on the errors significant enough to be reported to the New York State Department of Health [38]. The actual error rate and the seriousness of patient safety issues in RT are not clear due to lack of information on errors that have occurred. Because of legal concerns, often, only the people who were directly involved with an error have information about it and as a result many errors go undocumented [40, 37]. Ford and Terezakis [38] estimated an error rate of 1 in 600 patients while a study by Arnold et al. [10] reported error rates ranging from 0.06 percent to 4.66 percent depending on the error rate calculation method.

Some commonly reported errors in RT include (i) misidentification of the patient such as treating the wrong patient, (ii) misadministration including wrong procedure/examination/ treatment or dose, (iii) inadequate consideration of patient medical history, e.g., failure to determine if the patient is pregnant or has had prior radiotherapy, and (iv) radiation injury from an interventional procedure [22]. Major RT errors tend to evolve over time affecting a significant number of patients [10] and have potentially serious consequences including death in some cases [40]. The rate of serious/potentially-lethal errors is about 1/10,000 for patients in U.S. [80, 21]. These numbers compare unfavorably with other highly reliable industries such as aviation ($\sim$1 death in 4.7 million passenger flights) or other areas of medicine such as anesthesiology ($\sim$1 death in 200,000 procedures), suggesting that there are underlying systematic challenges that

need to be addressed to improve safety in RT [65]. This concern is also echoed by recent congressional hearings [117], and a public meeting sponsored by the Food and Drug Administration focusing on safety issues in RT [6]. The overarching goal of this research is to improve patient safety in radiation therapy by optimizing quality assurance in pre-treatment steps.

## 1.3 Classification of Errors

There is a lack of consistency in the definitions of the commonly used terms in patient safety such as error, incident and near miss and some terms are used interchangeably in literature. To ensure consistency, we introduce the following terms and classifications that will be used throughout this document: error and near miss. An error is defined as "an unwanted or unexpected change from normal system behavior which causes or has the potential to cause an adverse effect to persons or equipment" [37]. Reviewing literature, two factors distinguish different error types: "patient reached" and "patient harmed" [60]. Based on these factors, errors are classified into three groups that are shown in Figure 1.1 :

- Type 1 :"Errors that did not reach the patient"
- Type 2 :"Errors that reached the patient but did not cause harm"
- Type 3 :"Errors that reached the patient and caused harm"

In the case of "near misses" or "close calls" in healthcare settings, the negative consequences for patients are prevented. Some researchers [16, 57] use the term near-miss only for a type 1 error, i.e., when the error did not reach the patient as a result of timely intervention and correction, while others [13, 44] classify "near misses" as errors that did not harm patients regardless of the reason (type 1 and 2). The second definition includes both cases where (i) the error was caught before reaching the patient and serious harm did not occur as a result of effective corrective action and (ii) cases where potential harm was prevented either by chance or the patient's robustness. [60]. In this document, we use the first definition and only refer to error type 1 as a "near-miss".

Near-misses have been shown to be a valuable source of patient safety information. Besides providing information about corrective actions, there are many advantages and benefits regard-

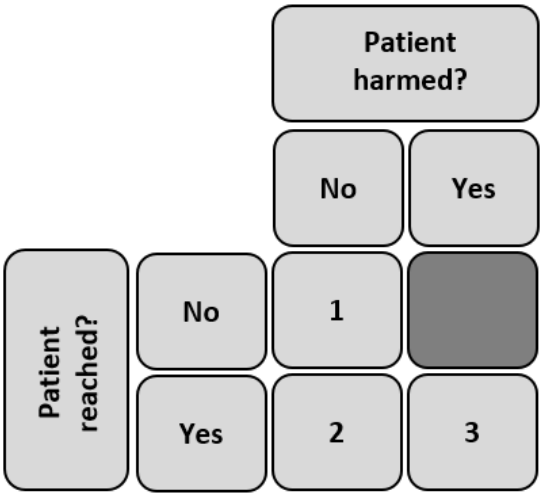


Figure 1.1: Classification of Errors - adapted from Kessels et al. [60]

ing reporting and analyzing near- misses. First, the frequency of occurrence of near misses is relatively high allowing for more near miss data to be collected over a shorter period of time. Second, near misses and errors that harm patients can share the same underlying causes. Therefore, investigating near-misses and eliminating their causes will help one prevent the occurrence of similar errors. Third, since near-misses did not cause serious harm to patients, employees are less fearful of legal actions and therefore are more likely to report near-misses [60, 14, 120]. However the lack of consensus regarding the definition of near-misses is likely to cause issues with understanding them. If providers are not educated about the definition of near-misses, they may not identify them correctly and consequently many near misses may go unreported and uninvestigated [35]. Kessels-Habraken et al. [60] argued that the optimal definition from a practical standpoint is subject to the context in which the organization is operating and even the standardized definitions may be interpreted differently in various organizational settings [60]. Hence the interpretation and analysis of errors and near-misses must consider the enviromental and organizational context.

## 1.4 Quality Assurance (QA) System in Radiation Therapy

The importance of quality assurance (QA) in RT has long been recognized [66, 128, 32, 55]. International organizations such as the European Society for Radiotherapy Oncology (ESTRO)

and the American Association of Physicists in Medicine (AAPM) have developed comprehensive guidelines for QA in RT [66]. Implementation of a comprehensive QA program has proven to be beneficial for preventing and detecting errors and improving the quality of care for patients who receive RT. A QA program is a "quality plan" which establishes the specific quality practices, resources, and activities associated with care delivery. Implementing a QA program requires a team of care providers that is committed to a policy of quality throughout for all activities it performs [66]. A QA program can improve safety performance as well as the safety culture through vertical-oriented interventions ("aimed at a specific process, technology or related errors") and horizontal-oriented interventions ("aimed at improving the culture or environment to reduce the number of human errors") [32].

In the next two sections we discuss key components of quality assurance programs: 1) safety barriers and 2) error reporting systems.

## 1.5 Safety Barrier (SB)

One of the widely recognized methods used to minimize the chance of error in the RT delivery is the use of safety barriers. A *safety barrier* (SB) is a quality assurance step that is formally embedded in the RT process to prevent an error from occurring or propagating [37]. Figure 1.2 (based on [102]) presents a schematic illustration of how errors are caught by safety barriers or missed by SBs and reach the patient based on the Swiss Cheese model of accident causation [101]. According to this metaphor, in a complex system, propagation of errors are prevented by a series of safety barriers. Each cheese slice corresponds to a SB. The holes in the slices represent the fact that SBs may fail to detect errors as they are largely dependent on individual performance. An error is classified as a near-miss if it is caught by one of the SBs before reaching the patient. Rework is expected after a near-miss, but there is no adverse consequence to the patient. If the holes in the cheese slices align, i.e., all SBs fail to catch the error, the patient may suffer devastating consequences.Although, the holes may not align frequently, when they do, the potential consequences are serious [20, 40].

A safety barrier checks one or multiple *patient elements*. A patient element is defined as any information specific to a patient such as patient name, diagnosis, prescription, etc. Each

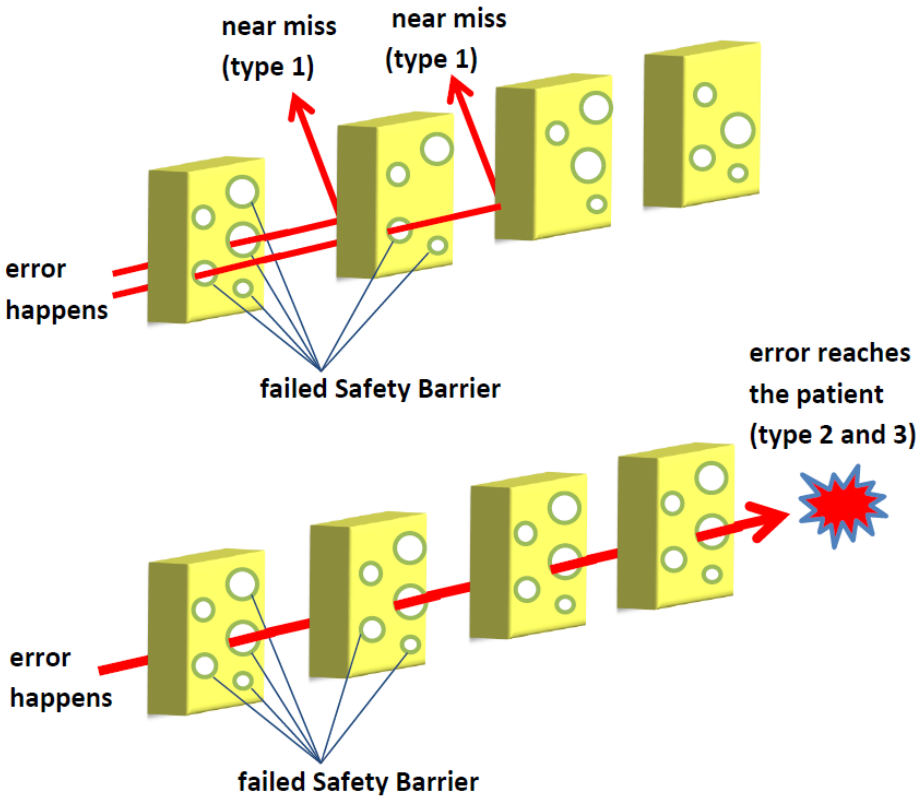


Figure 1.2: Swiss Cheese Model

element is related to a task completed or an outcome generated at a step in RT process. For example, "total dose" and "fractional dose" are two elements that are generated in "dose calculation" step. The elements are verified by SBs to make sure they are accurate and meet safety requirements before patient proceeds to the next step. For example, "dosimetry plan review" is a SB that checks the accuracy of total dose and fractional dose to make sure the correct amount of dose will be delivered to the patient.

It is estimated that about 40% of the steps in RT workflow are mainly focused on performing SBs to detect human and non-human errors [39]. Examples of SBs include "review of treatment plans by physicians and physicists, peer-review chart rounds, and measurement of intensity modulated radiation therapy (IMRT) beams before treatment". Although most of these SBs are not strictly regulated, they are considered standard practice and are addressed in the guidelines and reports produced by American Society for Radiation Oncology (ASTRO) and the American College of Radiology (ACR) [39].

## 1.6 Error Reporting System

As a part of a QA program an internal system of error reporting must be established aimed at: (i) reporting and documenting errors, (ii) analayzing errors and evaluating their impact on patients, and (iii) assessing the checking procedures (e.g. SBs) in a QA program [128]. Detailed analysis of data from error reporting systems provides valuable insights in preventing future errors. Several studies on error data demonstrate the benefits of SBs in improving patient safety [128, 25, 47, 10, 41]. Yeung et.al. [128] analyzed the data from 13,385 cancer patients that were treated at Northeastern Ontario Regional Cancer Centre (NEORCC) between November 1992 and December 2002. They showed that 58.7 % of errors related to "documentation" and "treatment planning" were detected by SBs before treatment delivery and no patient was harmed. Overall 97.9 % of errors resulted in a dose error of less than 5 % mainly because SBs improved early detection of errors and significantly reduced the number of high dose errors. Arnold et al. [10] analyzed three years (May 2004- 2007) of data from a radiation oncology department in Australia. Of the total errors reported, 22.5 % were errors that reached patients and 77.5 % were near misses. Most of the near-misses (65%) were detected in SBs before treatment delivery demonstrating the impact of SBs on preventing harm.

## 1.7 Research Objective

Despite the adoption of SBs, serious RT errors still occur and cause irreparable damage to patients [64]. While it may seem desirable to use as many SBs as possible to prevent errors from reaching the patient, unnecessary checks distract RT providers when performing critical procedures and cause extra workload that can lead to additional errors [128, 67, 82]. In the literature, there is little research on the potential impact, cost or optimal implementation of SBs [39]. Most of the research addressing safety challenges in RT focuses on the technical aspects of advanced RT technologies to optimize dose delivery [110, 23, 27, 28]. However, finding technical solutions alone, is not enough to achieve a desirable level of patient safety and other factors such as the implementation of SBs and other QA procedures should be studied and improved as well [80].

To the best of our knowledge, the only quantitative study was performed by Ford et al. [39]

where the effectiveness of a SB is defined as the percentage of errors that it can detect. They evaluated commonly used SBs in RT at two radiation oncology clinics and showed that the effectiveness of SBs largely depends on which SBs are being used and in which configuration. Ford et al. [39] assumed that SBs are 100% reliable, but in reality no SB is perfectly reliable.

We develop an optimization framework to identify reliable SB configurations to maximize patient safety. Specifically, we develop models to determine the optimal location and composition of SBs taking into account SB reliability and cost (i.e., time to perform the checks in SBs). The goal is to reduce the likelihood of harming patients due to medical errors during RT. We reduce the chance of errors reaching the patient by incorporating the penalty cost associated with errors in our objective function. In the following we discuss several challenges of our work and explain how they are addressed.

## 1.8 Challenge 1: Conflicting Objectives

The main challenge of our work is that it involves two conflicting objectives : 1) maximizing patient safety and 2) minimizing the cost of implementing SBs. As mentioned above, adding too many SBs to the RT process can be potentially harmful because unnecessary checks cause unnecessary delays, interruptions and extra workload which result in more errors [82]. We explicitly incorporate this trade-off in our mathematical and simulation models by incorporating the time-based costs associated with SBs as well as the patient safety implications such as expected cost/number of errors that reach patients.

## 1.9 Challenge 2: Quantifying The Performance of SBs

Another major challenge is to identify appropriate measures for evaluating the cost and effectiveness of SBs. In the analytical models proposed in Chapter 4, we quantify the performance of SBs in terms of cost measured in time. The cost has three elements:

- Checking cost (hour) : cost that is incurred when checking is performed at a SB.
- Rework cost (hour) : cost that is incurred when an error is caught and corrected at a SB.
- Penalty cost (hour) : cost that is incurred when an error reaches a patient.

We use the checking cost to take into account the amount of time that is spent on performing SBs to catch potential errors. Each time an error is detected by a SB, rework is necessary to correct the error. We assume the earlier an error is detected, the lower the rework cost associated with correcting it, will be. There is also a penalty cost for an error that reaches a patient. It is often challenging to estimate the penalty for an error reaching a patient especially without knowing whether it harmed the patient. An alternative way to evaluate the effectiveness of SBs is to measure the expected number or ratio of errors that reach patients in a given period of time. We used these measures in the simulation model proposed in Chapter 3.

## 1.10 Challenge 3: Complexity of RT Process

The RT care delivery process is complex with multiple steps, dependencies and constraints that makes modeling the process challenging and adds to the computational complexity of the analytical model. We address these complexity challenges in two ways: First, by our use of computer simulation models which will allow us to capture many of the complexities inherent in the real life RT process and second, we have developed time-efficient solution methods to reduce the computational complexity and achieve near-optimal solutions.

## 1.11 Challenge 4: Uncertainties in RT Process

The RT process is subject to various sources of uncertainty that complicates parameter estimation for our problem. To address this issue, we use stochastic programming which is commonly used for modeling uncertainty in the parameters of optimization models. In Chapter 5, we propose a two-stage stochastic programming framework to captures uncertainties in RT process.

## 1.12 Research Goals

Considering the challenges associated with modeling and optimizing the design of SBs in RT we seek to answer the following main research question: How can we optimally design SBs to maximize their impact on patient safety considering the implementation costs?

Considering the above research question, the relevant research goals are:

- To quantitatively measure the cost and effectiveness of SBs and capture the trade-off between them
- To utilize the data from error reporting systems to estimate the input parameters for our models
- To develop efficient solution methods to address the computational complexity issues
- To incorporate the uncertainties in model parameters (stochastic optimization)

The overall goal is to develop analytical model to help with the cost-effective design of SBs that results in early detection of errors and reduces the chance of harm to patients. In the proposed models, we take into account the penalty cost associated with errors that reach the patient to reduce the harm to patients.

The remainder of this dissertation is organized as follows: Chapter 2 provides an overview of the literature related to our problem. We discuss the similarities and differences between our problem and the existing literature on reliable facility location problems and inspection allocation problems. In Chapter 3, we introduce the discrete-event simulation model and the case study conducted at the University of North Carolina (UNC) Department of Radiation Oncology. Chapter 4 presents mixed integer programming models for identifying the optimal location and composition of SBs. We provide a detailed discussion of the model formulation, structural properties and the solution methods developed to solve the problem. In Chapter 5, we present a risk-averse two-stage stochastic programming framework to address uncertainties in RT process. Chapter 6 concludes this dissertation with a summary of main contributions and some ideas for future work.

We took advantage of these similarities to linearize our model and also benefit from the solution approaches proposed for RFLP. One alternative approach to formulate RRTP is to limit the number of errors that reach the patient by incorporating an additional constraint into the model that ensures the expected number of errors reaching the patient does not exceed a given $\epsilon$. In this approach, the objective function only consists of the checking cost and the rework cost associated with detecting errors. While this approach provides more flexibility in terms of constraining the number of errors that reach the patient, it would be more difficult to develop an efficient solution approach to solve the problem. Since our proposed model is motivated by RFLP, we were able to develop an efficient solution algorithm for RRTP similar to the approaches proposed for RFLP.

# Chapter 2

# Literature Review

In this chapter we review the literature related to our study. The modeling approach and solution techniques that we propose for our problem are inspired by the literature in two areas: (i) inspection allocation problems and (ii) reliable facility location problems. In Section 2.1 we present a discussion of the literature focused on inspection allocation problems and in Section 2.2 we review the preceding work in the area of reliable facility location problems. We conclude each section by discussing the differences of our work from the related literature.

## 2.1 Literature Review on Inspection Allocation Problems

Safety Barriers (SB) in Radiation Therapy (RT) share many similarities with inspection stations in multi-stage production systems. The ability to produce high quality products is one of the crucial success factors in manufacturing industry [84]. Therefore inspection procedures are performed to ensure that products comply with regulations and meet the required standards to be released in market. Inspection by humans, automated inspection by sensors or a combination of both may be used as a part of quality assurance (QA) for manufacturing processes [78]. Similarly, SBs are used at various stages in RT process to ensure effective and safe treatment for cancer patients. Inspection stations examine units to detect defective products before they are shipped to customers and SBs check and verify patient elements to catch errors before they reach patients. Performing an inspection is costly and to optimize the number of inspection stations and their location, one must consider the inspection costs as well as the benefit of

inspection in producing defect-free products. The same considerations should be taken into account regarding the cost of SBs and their impact on patient safety. These similarities motivated us to review the inspection allocation literature in order to gain a better understanding of our problem. Table 2.1, shows the relation between different elements of our problem and the inspection allocation problem.

Table 2.1: Comparing reliable radiation therapy problem to inspection allocation problem

| **Reliable Radiation Therapy Problem** | **Inspection Allocation Problem** |
|---|---|
| Patient Element | Production Unit |
| Error Type | Defect Type |
| Error Probability | Defect Probability |
| Safety Barrier | Inspection Station |
| SB Failure Probability | Probability of Accepting a Defective Item |
| Checking Cost | Inspection Cost |
| Rework Cost | Internal Failure Cost |
| Penalty of Errors that Reach Patients | External Failure Cost |

Inspection planning for production systems is a popular research domain and there are many studies focused on inspection allocation problems. Examples of studies in this area include determining the number and locations of inspection stations, defining inspection plans (full inspection or sampling), deciding the best action (e.g., rework, repair or scrap) to take when finding defective units, incorporating different constraints (e.g., inspection time, average outgoing quality required or budget limit) and modeling different kinds of production systems [50].

The additional cost of investing in inspection stations and sensing devices in a production system is expected to be outweighed by the benefits associated with detecting defective items and variation sources that lead to defects. Inspecting finished products results in a low inspection cost, but rework or scrapping a finished product can be very costly and in most cases it is not possible to identify the source of defects. On the other hand, a policy emphasizing inspecting unfinished products significantly increases the inspection costs, but at the same time scrapping and reworking unfinished products will be less expensive and diagnosing the defects or variation sources will be easier. Therefore, determining where to allocate inspection effort in a production

process is a challenging decision that requires exploring the trade-off between multiple objectives [78].

According to Mandroli et al. [78] the relevant inspection literature can be classified into two groups: i) inspection-oriented QA strategies and ii) diagnosis-oriented sensor-distribution strategies. An inspection oriented strategy determines an optimal level of inspection efforts by balancing the cost factors with respect to inspection, rework, repair and penalty in cases where a defective product is delivered to a customer. These types of strategies aim to produce a final defect-free product for customers, but since they do not improve the underlying manufacturing process, the "overall product quality" does not change.

On the other hand, diagnosis-oriented strategies also known as sensor-distribution strategies focus on identifying the major defect-causing sources. For these strategies, the assumption is that there is a set of unknown variables that cause quality issues but they are not directly observable. Any change in the status of these variables should be tracked using sensors to diagnose the variables that are causing defects [115]. The research on sensor distribution strategies is relatively new and is still in its early stages [78]. Our work on SBs in RT can be associated with the first group of studies that focuses on developing cost-effective inspection plans instead of root-cause diagnosis, therefore we primarily focus on the publications on inspection-oriented strategies.

The major decision in inspection-allocation problems is whether to inspect or not inspect a product after each manufacturing step. If it is decided to perform inspection, the next step is to determine the inspection threshold or specification limits to distinguish conforming and nonconforming products. If units are processed in batches, another decision is whether to inspect all items or a fraction of them. Upon completion of an inspection, the conforming units move to the next operation and the action for non-conforming units will be one of the following: 1) replacing with a conforming unit, 2) reworking or repairing, or 3) scrapping [111, 78]. Figure 2.1 shows inspection at a multi-stage manufacturing process.

In some studies, the location of inspection stations is assumed to be fixed and therefore the main decisions are concerning the fraction of units to be inspected or the appropriate action for non-conforming units This type of research is called a "parametric strategy" as opposed to the "allocation strategy" studies that focus on determining the location of inspection stations

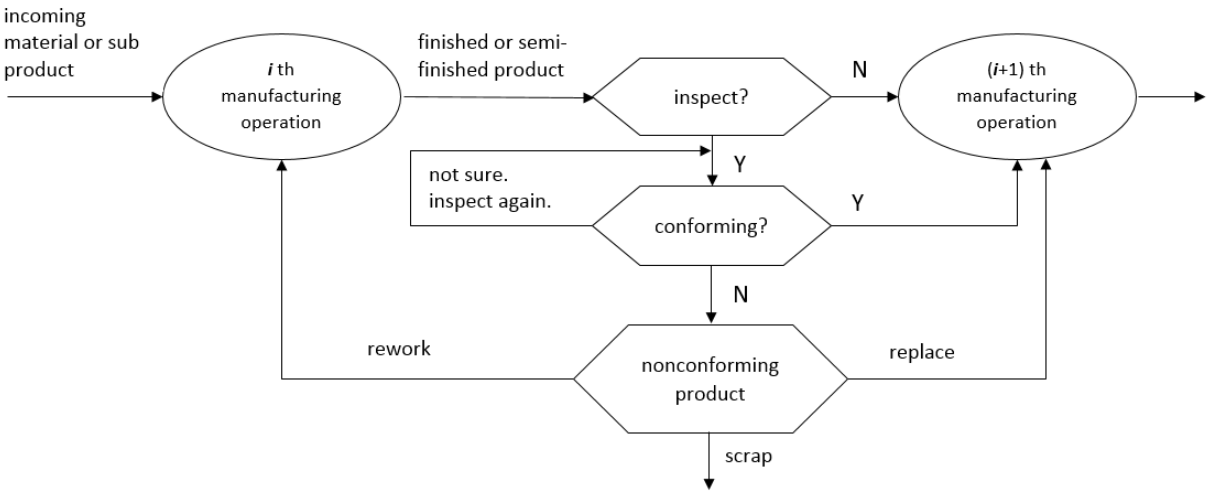


Figure 2.1: Inspection station in a multi-stage manufacturing process- adapted from Mandroli et al. [78]

[78]. In the following we explain some important aspects of inspection allocation problems that can be used to characterize a manufacturing process [78] and also discuss which category our problem falls into :

**Process Configurations** : There are three types of process configurations in a production facility: i) serial system, ii) assembly system, and iii) nonserial system. In a serial system, there is only one sequential flow path as opposed to nonserial systems which have multiple flow paths for input material. An assembly system is a nonserial system where each workstation has exactly one successor node and can have many predecessor workstations. A system that is neither serial nor assembly will be classified as a nonserial system [43, 129]. Our problem falls into the category of a serial system. We assume all patients pass through a single sequential path to complete their treatment.

**Defect Rate** : The defect rate is defined as the number of non-conforming units to the total number of items processed at a manufacturing stage. The defect rate may be defined as a constant or a variable based on a random distribution. Some researchers have assumed a single defect type [46, 122, 95], whereas others have looked at multiple defect types [97, 112, 100, 129]. Overall, four possible combinations can be found for defect rates: i) single type constant, ii)

single type variable, iii) multiple types constant, and iv) multiple types variable [78]. We can use the same criteria to classify RT errors. Different types of errors each associated with a specific patient element may occur in RT process, so we have multiple error types. In our deterministic model we have assumed constant error probabilities (the third combination) and in our stochastic programming model we considered error probabilities as random variables following a uniform distribution (fourth combination).

**Inspection Capability** : Ideally, an inspection station should detect all non-conforming units and accept all conforming ones. However, this is not the case in real world and inspection stations may not perform as expected due to various reasons such as human error. Two types of errors may occur when an performing inspection: i) Type I error which occurs when a conforming unit is rejected incorrectly and ii) Type II error when a non-conforming unit is falsely accepted. Some authors have considered both types of errors [46, 112, 100] and some have assumed only one type of error or an error-free inspection [97, 95, 36]. We consider type II errors given that SBs are not 100% reliable and may fail to detect all the errors associated with the elements they check. We assume the probability of false positive (type I) error is negligible and there is no false alarm if no error has occurred.

**Inspection Type** : We define the inspection type as the ability to detect different types of defects. Most studies have assumed that inspection stations are versatile meaning they can detect all types of defects produced by previous workstations [97, 46]. A few studies have considered specialized inspection stations that can only detect defects generated at certain workstations [103, 112]. We assume SBs belong to the second category (specialized) as each SB can verify certain elements and therefore can only detect certain errors.

**Cost Components** : The cost function usually consists of the manufacturing cost, inspection cost and quality failure costs. The inspection cost is the sum of a fixed and a variable cost. The fixed cost represents the cost of setting up inspection equipment and the variable cost is associated with each unit inspected. The majority of papers assumed a linear relationship between the total variable inspection cost and the number of units inspected. A few papers

used a quasi-concave function or a general non-linear function [78]. The internal failure cost includes the cost of repairing, replacing or scrapping a defective item that is detected before it is shipped to the customer. The external failure costs are incurred after the defective products are delivered to a customer. This cost include the cost of repair, replacement of defective parts and quality loss. External failure cost may depend on the defect type [123] or could be independent of it [112]. In some studies each type of cost is considered [112] while in others only a subset of them are considered [54]. According to a survey by Shetwan et al. [111] on allocation of inspection stations, most papers have not considered all elements of the external failure cost and instead assumed an aggregated penalty cost representing the cost of detective units being delivered to the customer. We define the cost components for our problem in a similar manner: we consider the cost of performing SBs (checking cost), the rework cost for those errors that are caught before reaching patients and the cost of errors that reach patients. We do not consider any fixed cost associated with investing in and setting up SBs and only assume a variable checking cost that is incurred each time a patient element is checked at a SB.

**Objective function** : Most papers that studied inspection allocation problems, seek to minimize the total cost consisting of manufacturing cost, inspection cost and quality failure costs, but there are a few papers that consider other objective functions such as maximizing the throughput or production capacity [99, 88]. Shetwan et al. [111] argued that this usually happens when an inspection allocation problem and a scheduling problem are considered simultaneously. In the case of cost minimization, the common objectives are: total expected cost and expected unit cost, but the studies may differ in how the number of units is counted [78]. For our problem, we minimize the total expected cost to evaluate and optimize the design of SBs.

**Constraints** : The constraints that are considered in inspection allocation problems often address the characteristics of the production system such as the process configuration and type of defect [111]. Other common constraints that may be used include: i) the average outgoing quality limit (AOQL); 2) a limit on the total number of inspection stations; 3) budget constraints restricting inspection resources; and 4) minimum throughput or production capacity [78].

**Solution Methods** : According to the surveys conducted by Mandroli et al. [78] and Shetwan et al. [111], dynamic programming (DP) is the most commonly used method for solving inspection allocation problems. Shetwan et al. [111] stated that what makes DP a desirable choice is probably the multistage structure of a manufacturing process. However some researchers [113, 114] have pointed out that DP becomes impractical when the problem size grows and developed heuristics to solve the problem. The second popular approach is nonlinear programming (NLP). Almost all papers deal with a non-linear total cost function with some integer variables (such as whether to inspect or not, the number of inspection stations, etc.), therefore almost every problem is a NLP with many also being integer programming problems. The computational efficiency of NLP drops substantially for large-sized problems. As a result, heuristic and meta-heuristic methods including simulated annealing (SA) and genetic algorithms (GA) are used to find efficient solutions for larger problems. In addition to numerical optimization techniques, several authors have used simulation techniques such as Monte Carlo simulation [111].

Table 2.2 summarizes some of the existing literature on inspection allocation problems based on their modeling characteristics: process configuration, defect rate, inspection type, inspection capability and solution methods. One of the main features that makes our model different from the previous studies in the inspection allocation literature is that we assume that each SB can detect a certain subset of errors and we optimize which elements (error types) should be assigned to it. There are a few studies that focused on specialized inspection stations. However, they either limited the capability of inspection stations and constrained them to only detect defects produced at the immediate preceding operation [103] or they defined a few inspection classes each associated with a few workstations and then assigned each inspection station to a class [112]. Our model is more flexible in terms of being able to customize SBs and assign a different combination of elements to each SB.

Table 2.2: Summary of Literature on Inspection Allocation Problems

| **Publication** | **Process Type** | **Defect Rate** | **Inspection Type** | **Inspection Capability** | **Solution Method** |
|---|---|---|---|---|---|
| Pruzan and Jackson, 1967 [97] | serial | multiple-constant | all defects | error free | DP |
| Rebello et al. 1995 [103] | serial | multiple-constant | specialized | I and II | heuristic |
| Veatch, 2000 [123] | serial | single-random | all defects | type II | |
| Zheng, 2000 [129] | assembly | multiple-random | all defects | error free | heuristic |
| Heredia et al. 2002 [46] | serial | single-constant | all defects | I and II | GA |
| Shiau, 2003 [112] | serial | multiple-constant | specialized | I and II | NLP- heuristic |
| Finkelshtein, 2005 [36] | serial | single-constant | all defects | error free | DP |
| Van Volsem et al. 2007 [122] | serial | single-constant | all defects | error free | GA |
| Penn and Raviv, 2008 [95] | serial | single-constant | all defects | error free | DP |
| Rau and Chu, 2015 [100] | non-serial | multiple-constant | all defects | I and II | NLP- heuristic |
| **Our work** | serial | multiple-constant | specialized | type II | search and cut algorithm |

## 2.2 Literature Review on Reliable Facility Location Problems

This section provides an overview of the literature on reliable facility location problems. We discuss how our model is comparable to reliable facility location model and also describe the unique features of our problem. In this section we review some of the relevant work in this area, and explain how different characteristics of reliable facility location problems can be translated into the RT setting.

Facility location models are well studied in the literature. These models aim to locate one or more facilities among a set of candidates and to assign customers to the facilities with the objective of minimizing cost or maximizing total profit [109, 30]. Most models in the literature assume facilities never fail [109], but in reality facilities may become unavailable for various reasons such as adverse weather, natural disasters or terrorist attacks [29]. Reliable facility location problems focus on supply chains with facilities that are susceptible to unexpected failures and disruptions [1]. In such supply networks, in the case of failures, customers have to be served by facilities other than the primary facility to which they were assigned [1]. To hedge against the unexpected failures and to diminish the impact of transportation and penalty costs that occur in the case of failures, each customer is assigned to multiple facilities that serve as backup [1]. The facilities are classified into "levels" and customers are assigned to facilities in the order of levels. A *"level-k"* assignment is one for which there are $k$ closer failable facilities that are open [116]. Each demand point is served by the lowest-level facility that is available. So, a higher level facility will only serve a customer if the facility is operational and all the lower level facilities have failed [109, 1]. The objective is to find a cost-efficient solution to minimize the operating and transportation costs under both normal and emergency conditions [29].

The problem of optimizing the design of SBs in RT can also be viewed as a reliable facility location problem: there is a network of SBs responsible for detecting several types of errors that may occur with different probabilities. Multiple SBs check each element in ordered levels, because SBs are not perfect and may fail for different reasons. If the first available SB fails to detect an error, the next SB that is assigned to the element, may catch the error with some probability. Although, assigning an element to backup SBs increases the checking cost, it also helps reduce excessive rework costs that occur when errors are caught late in the process or

penalty costs of errors that reach patients. The goal is to find a safe and cost-efficient SB design to minimize the total cost associated with errors and performing SB checks. Table 4.2 shows how different components of our problem can be mapped to similar concepts in the reliable facility location problem.

Table 2.3: Comparing reliable radiation therapy problem to reliable facility location problem

| **Reliable Radiation Therapy Problem** | **Reliable Facility Location Problem** |
|---|---|
| Patient Element | Customer |
| Error Probability | Demand |
| Safety Barrier | Facility |
| SB Failure Probability | Facility Failure Probability |
| Rework and Checking Cost | Transportation Cost |

The reliability issues in a supply chain network have attracted increased attention among researchers [29]. Snyder et al. [116] introduced the concept of reliability in facility location problems. They proposed reliability models based on two facility location problems: the P-median Location Problem (PMP) and the Uncapacitated Fixed-charge Location Problem (UFLP) assuming facilities have equal failure probabilities. In the case of the Reliable P-median Location Problem (RPMP), the objective function is the weighted sum of the operating costs under normal conditions and the failure costs (i.e., the expected transportation cost where facilities fail). In Reliable Uncapacitated Fixed-charge Location Problem (RUFLP) the cost of building facilities is included in the objective function. The authors developed Lagrangian-based solutions for both models and showed that reliability of a supply chain network can be significantly improved with a relatively small increase in operating costs [116].

Several studies later relaxed the assumption of uniform failure probabilities. Li et al. 2013 [70] extended the PRMP model proposed by Snyder et al. [116] and incorporated facility dependent failure probabilities. Berman et al. [18] studied the "median problem with unreliable facilities" (MPUF) with both uniform and nonuniform failure probabilities. They proposed a structure where a subset of facilities are assumed to be perfectly reliable and some facilities may fail with some probability. They focused on several structural and algorithmic properties of their general MPUF model and analyzed the optimal solution patterns. They introduced

and analyzed a pattern called "co-location" which is defined as locating more than a single facility at one location. According to their analysis, increasing failure probabilities results in an increased"co-location" effect as well as facilities becoming more centralized. They also developed several exact and heuristic approaches for both uniform and nonuniform MPUFs. Qi and Shen [98] considered a three-level supply chain network consisting of one supplier, multiple facilities and retailers where facilities distribute products provided by a 100% reliable supplier to retailers. They considered non-uniform failure probabilities as well as the cost of inventory and proposed a nonlinear optimization model [98]. Cui et al. [29] also assumed facility-dependent failure probabilities, and proposed a Lagrangian algorithm for mid-sized reliable uncapacitated fixed-charge location problems (RUFLP) and a Continuum Approximation (CA) model to find an efficient solution for large-scale RUFLP.

A scenario-based stochastic programming (SP) formulation for RUFLP was proposed by Shen et al. [109]. They assumed that in the first stage it is unknown which facilities are going to be operational and the decision is to determine which facilities should be open. In the second stage, when the operational facilities are revealed, each customer is assigned to an open and operational facility if the cost of assignment is lower than the penalty cost associated with failing to satisfy customer demand, otherwise penalty costs is incurred. The authors mentioned that the number of scenarios will be very large when assuming site-dependant failure probabilities and proposed a non-linear integer programming (NLIP) formulation for the case of independent failure probabilities. They evaluated the performance of two greedy heuristics for RUFLP-NLIP and a sample average approximation to solve RUFLP-SP. For a special case of the problem with equal failure probabilities, they developed a four approximation algorithm.

Lim et al. [72] formulated the general reliable facility location problem as a mixed-integer programming model considering "facility hardening" decisions which distinguishes their model from other models in literature. Facility hardening refers to investing in various protection plans for a facility to hedge against the risk of facility disruptions. They studied structural properties of the model and developed a Lagrangian-based solution for problem. Aboolian et al. [1] derived lower bounds for the RUFLP problem and developed a search and cut algorithm to solve the problem. They compared the algorithm against the Lagrangian-based method developed by Cui et al. [29] and demonstrated that their method outperforms Lagrangian relaxation. They also

argued that the performance of the search and cut method is not very sensitive to the maximum assignment level for each customer but can be significantly affected by the failure probabilities. Maximum assignment level refers to the maximum number of facilities that a customer can be assigned to. In our problem we define maximum assignment level as the maximum number of times that we can check an element.

Most studies on reliable facility location problems have assumed facility disruptions occur independently meaning there is zero correlation between the failures that occur at different facilities. Recently a few studies [71, 7, 75] have incorporated disruption correlations arguing that in practice disruptions tend to be strongly correlated. One reason to model disruption correlation is that natural disasters and severe weather conditions often affect large geographical areas in a relatively short period of time [75]. According to the results of the study by Lu et al. [75] ignoring disruption correlation can lead to significant losses.

Based on our review of the literature, we have identified a set of main characteristics that can be used to classify reliable facility location problems. In the following, we go over these criteria and then we use them to categorize the literature.

**Facility Failure Probability**: Facilities may have uniform or non-uniform failure probabilities. In the uniform case, all facilities have the same failure probability whereas in the non-uniform case the failure probabilities are facility-dependant. Some studies have studied both cases [18, 109]. In our model, we assume failure probabilities that are associated with elements and SBs.

**Modeling Approach**: Several modeling approaches are used in reliable facility location problems. We focus on classical models that are commonly found in literature: (i) Reliability P-median Problem (RPMP) and (ii) Reliable Fixed-charge Location Problem (RFLP) [30]. In RPMP, exactly $P$ facilities have to be located whereas in RFLP the constraint on the number of facilities is relaxed but there is a fixed cost for opening facility. We formulate our problem as a RPMP assuming there is a constraint on the number of available SBs.

**Disruption Correlation**: Correlated facility disruptions have been studied by a few recent

studies [71, 7]. In the real world, there is usually a strong correlation between facility disruptions mainly because closer facilities are more likely to be affected by similar events. Positive correlations cause neighboring facilities to fail simultaneously, while under negative correlations, customers are more likely to be served by neighboring facilities to avoid excessive transportation costs [71]. We assume SB failure probabilities are independent meaning that there is zero correlation between them.

**Solution Methods**: Various solution methods have been developed to solve reliable facility location problems. Some common methods are Lagrangian relaxation-based methods, greedy heuristics and approximation algorithms. We present the solution method for our problem in Chapter 4. The proposed heuristic method is based on continuous improvements of lower and upper bounds on objective function.

Table 2.4 presents a summary of literature based on these characteristics. We took advantage of the similarities between our problem and reliable facility location problem (RFLP) to linearize our model and also benefit from the solution approaches proposed for RFLP. Despite the similarities, there are still three major differences between our problem and the existing models for reliable facility location problem: first no customer can be served by a failed facility in the reliable facility location problem. However, in the reliable radiation therapy design problem, SBs consists of multiple elements meaning that when a safety barrier fails to detect an error in a particular element, it may still successfully detect errors in other elements. Second, elements will be assigned to SBs at levels associated with the position of SBs in the process. This means that the position of the SB checking an element at safety level $k$ must be earlier than the position of the SB checking the same element at safety level $k+1$. This is because SBs are performed sequentially with respect to the order of their positions in the RT process. In reliable facility location problem there is no constraint on the facilities that can serve a customer and customers can be assigned to facilities at levels regardless of their location in the network. Finally, the reliable facility location problem assumes that customers do not travel to failed facilities and the transportation cost only incurs when the customer demand is served by a backup facility. However, in the reliable radiation therapy design problem, the cost of checking

Table 2.4: Summary of Literature on Reliable Facility Location Problems

| **Publication** | **Failure Probabilities** | **Modeling Approach** | **Disruption Correlation** | **Solution Approach** |
|---|---|---|---|---|
| Snyder et al. 2005 [116] | uniform | RUFLP, RPMP | uncorrelated | Lagrangian |
| Berman et al. 2007 [18] | uniform, nonuniform | RPMP | uncorrelated | greedy heuristics |
| Qi and Shen, 2007 [98] | nonuniform | Other | uncorrelated | greedy heuristics |
| Lim et al. 2010 [72] | nonuniform | RUFLP | uncorrelated | Lagrangian |
| Cui et al. 2010 [29] | nonuniform | RUFLP | uncorrelated | Lagrangian, continuum approximation |
| Shen et al. 2011 [109] | uniform, nonuniform | RUFLP, RUFLP-SP | uncorrelated | approximation algorithm |
| Peng et al. 2011 [94] | uniform | RUFLP-Robust | uncorrelated | genetic algorithm |
| Aboolian et al. 2012 [1] | nonuniform | RUFLP | uncorrelated | search and cut algorithm |
| Li et al. 2013 [70] | nonuniform | RPMP, RUFLP | uncorrelated | Lagrangian |
| Li and Ouyang 2013 [71] | nonuniform | RUFLP | correlated | continuum approximation |
| An et al. 2014 [7] | uniform | RPMP-Robust | correlated | column-and-constraint generation algorithm |
| Jalali et al. 2015 [52] | uniform | RUFLP-SP | uncorrelated | meta-heuristics |
| Lu et al. 2015 [75] | nonuniform | RUFLP-Robust | correlated | decomposition algorithm |
| **Our work** | nonuniform | RPMP | uncorrelated | search and cut algorithm |

an element is still incurred even when a SB fails to detect an error related to that element.

## 2.3 Contribution of This Work

This research contributes to the existing literature in two ways: clinically and mathematically. From a clinical perspective, we quantify the effectiveness and cost of SBs and their impact on patient safety in the complex process of RT. We develop an optimization framework to determine the reliable design of SBs, i.e., the location and composition of SBs, taking into account SB failures. Our proposed framework simultaneously considers the objectives of minimizing cost and maximizing the effectiveness of SBs and finds the optimal design by exploring the trade-off between these two conflicting objectives. From a mathematical modeling standpoint, our problem is related to inspection allocation problems and reliable facility location problems and borrows concepts from both areas. Our problem can be viewed as an inspection allocation problem with customized inspection stations that can detect multiple types of defects. Moreover, our problem can be viewed as a reliable facility location problem where SBs are facilities and elements are customers. We formulate our problem by extending the reliable P-median facility location problem (RPMP). We incorporate additional constraints to the classic RPMP to address the fact that the location of SBs restricts the elements they can check. Plus, we consider failure probabilities that depend on both elements and SBs. In the next Chapter we discuss the simulation model that we develop to assess the reliability of SBs in RT process. Using our findings from the simulation model, in Chapter 4 we develop mixed integer programming models for the SB reliable design problem. In Chapter 5, develop a stochastic programming framework address the uncertainties in problem parameters.

# Chapter 3

# Simulation Modeling

In this chapter we present a discrete event simulation (DES) model of the radiation therapy (RT) process with the goal of assessing process reliability and patient safety. Patient safety is the primary concern in RT and safety considerations should be incorporated into all aspects of treatment planning and delivery. We develop a detailed DES model of the RT process at the Department of Radiation Oncology of University of North Carolina (UNC) Lineberger Comprehensive Cancer Center. We evaluate the performance of safety barriers (SB) and measure their effectiveness in reducing errors and preventing harm to patients. This chapter is structured as follows: first, we present a general overview of literature in Section 3.1 and then we describe the RT process and SBs used at UNC Department of Radiation Oncoloy in Section 3.2. We describe the DES model in Section 3.3 and we discuss the model verification in Section 3.4 followed by sensitivity analysis and model validation in Section 3.5. We present the results of what-if analysis in Section 3.6 and conclude and discuss future work in Section 3.7.

## 3.1 Literature Review

Uncertainty is inherent to healthcare systems and therefore decision making in healthcare organizations is a challenging task. The application of computer simulation can effectively addresses the stochastic and dynamic nature of problems in healthcare settings [87]. Katsaliaki et al [59] have conducted a comprehensive literature review on the simulation research in healthcare and highlighted the importance of simulation in providing insights to deal with these systems.

Several applications of the simulation techniques have been addressed in this review including health risk assessment [31, 74, 8], prognostic and transmission models of health interventions, [79, 62, 51], planning of healthcare systems [49, 118, 108], public health policy evaluation [11, 3, 2], and modeling health systems disruptions [24, 68, 9].

Reviewing the healthcare simulation literature, reveals that only few researchers have used simulation techniques to study and improve the RT process from an operational standpoint. Among the simulation studies in the RT context, the majority have focused on reducing long delays and waiting times due to inefficient scheduling and resource allocation policies. Ogulata et al [91] developed a slack capacity approach for scheduling patients in RT and conducted a simulation analysis to determine the appropriate scheduling parameters. A DES model by Kapamara et. al [56] was used to analyze patient flow in RT, revealing the intricacies and potential bottlenecks of the process. Ebert et al [34] proposed an approach to determine optimal waiting times and maximized the efficiency of RT. Their analysis suggests that tumor doubling time is the key factor in determining optimal waiting time. Wang and Lee [125] developed a DES model to improve patient flow in the Radiation Oncology Department at Central Baptist Hospital in Lexington, KY. They evaluated patient's length of stay (LOS) and waiting time under several what-if scenarios for staffing levels, arrival pattern, scheduling policies to enhance the operational efficiency of the clinic. Similarly, Seplveda and Cahoon [106] used simulation to improve the cancer treatment process at M.D. Anderson Cancer Center in Orlando, FL. Their analysis was not limited to the Radiation Oncology department and it aimed to improve patient flow from the time they arrive to the hospital until they leave. Using a DES model, they compared and evaluated alternative floor layouts and patient scheduling techniques, to identify the potential bottlenecks and optimize patient flow. Monte Carlo simulation was used in a study by Munro and Potter [86] to estimate 95 % confidence intervals for projected waiting times. Thomas [121] also performed a Monte Carlo analysis to determine the maximum patient capacity limits to meet target waiting times. In a more recent study, Werker et al [126] used DES to model the planning section of the RT process at the British Columbia Cancer Agency and recommended improvements to RT planning waiting times. They showed that delays associated with oncologist tasks have the biggest impact on planning times and to achieve faster RT planning, reducing oncology-related delays must be the first priority.

To the best of our knowledge, this is the first simulation study that addresses the patient safety in RT process. We used DES which is recognized as an effective and flexible technique for modeling and analyzing health systems and processes [59]. According to Katsaliaki et al. [59] applications of DES in healthcare can be classified into the following groups: planning for healthcare services, health economics models and modeling contagious disease interventions. The first category refers to the application of DES to evaluate and improve the performance of delivery systems by exploring the relationships between components in a complex system such as hospitals and emergency departments [59, 85, 19, 15]. The DES model that we develop falls in this category as we also model the RT process to improve patient safety and process reliability. DES allows decision makers to predict the effect and results of implementing a new policy or changes in an existing policy or strategy [87, 59]. This is advantageous for our problem since modifying a process or policy can potentially be very costly and have significant impacts on patient-related outcomes. Therefore it would be helpful to test proposed changes in a risk-free simulation environment.

## 3.2 Case Study

In this section, we describe a case study undertaken at the UNC Department of Radiation Oncology, a part of the UNC Lineberger Comprehensive Cancer Center. The UNC Cancer Center opened in September 2009 and currently serves patients from every county in North Carolina with more than 135,000 patient visits each year. The UNC department of Radiation Oncology offers the following services: 1) external beam radiation therapy (EBRT) 2) intensity-modulated and image-guided radiation therapy 3) Tomo-therapy 4) CyberKnife robotic radiosurgery and 5) Brachytherapy. We focused on the EBRT which is the most common form of RT used for cancer treatment at UNC Department of Radiation Oncology.

The goal of EBRT is to deliver high-energy photon beams to malignant tumors while minimizing risks to neighboring healthy tissues. The process can be divided into seven stages shown in Figure 3.1. The stages of the process are as follows:

- *Patient decision and patient assessment*: In the first step, radiation oncologist examines the patient to determine if RT is an appropriate treatment; if so, a tentative prescription is generated. RT may be used alone or in combination with surgery, chemotherapy or both.

- *Simulation and imaging for RT planning*: The ideal treatment position is determined and immobilization tools are applied and then computed tomography scans of the patient are taken in the treatment position.

- *Treatment planning*: RT professionals (i.e., oncologist, physicist, and dosimetrists) design radiation beams based on the scans, electronic medical records, and diagnostic images.

- *Pre-treatment review and verification*: Treatment instructions are reviewed to ensure that they are correct and in accordance with physician's directive. The final treatment plan is entered into a software that controls the radiation delivery machine.

- *Treatment delivery*: Radiation therapists position the patient as planned on the treatment couch, images are obtained for accuracy, and RT is delivered. Depending on the cancer type and the patient characteristics, a course of treatment is usually delivered in five sessions per week for several weeks.

- *On-treatment Planning and Verification*: The oncologist reviews treatment progress and makes changes as needed.

- *Post-treatment completion*: The treatment course and patient outcomes are evaluated.

The potential for errors in RT is high. Error can occur at any stage and if not caught early, they may cause severe consequences for patients. Errors mainly occur because RT planning and delivery are complex processes and involve numerous handoffs between healthcare providers, each of whom interpreting and documenting medical information via multiple electronic systems [53]. Figure 3.2 shows the people involved at each stage of RT process. As shown in the figure, planning and delivering RT is a team effort which requires healthcare providers of different disciplines to interact and collaborate.

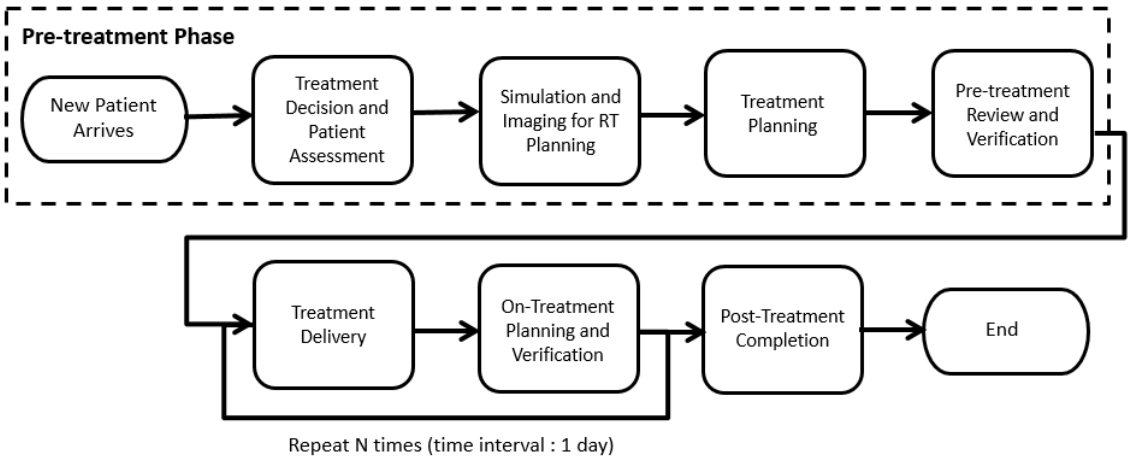


Figure 3.1: Stages of Radiation Therapy Process

Despite the fact that RT providers use sophisticated tools to ensure patient safety, most steps remain largely dependent on individual performance. For example, to minimize the radiation exposure to healthy tissues, dose distributions are optimized using special softwares and then are transferred through computer networks to treatment machines for automated dose delivery [58]. However, a safe and effective RT delivery still relies on the interaction between the computerized systems and providers to tailor the treatment according to the patient's needs. Errors can occur at any step, yet several studies have found the "pre-treatment" phase to be the most common source of errors [33, 81]. The "pre-treatment" phase as highlighted in Figure 3.1, include the four stages that are completed before treatment delivery: (i) Patient decision and patient assessment, (ii) Simulation and imaging for RT planning, (iii) Treatment planning and (iv) Pre-treatment review and verification. Of the 3,111 adverse errors that occurred during RT delivery as identified by the World Health Organization, more than 50% were attributed to the pre-treatment phase [127]. Therefore, we focus on the steps in pre-treatment phase.

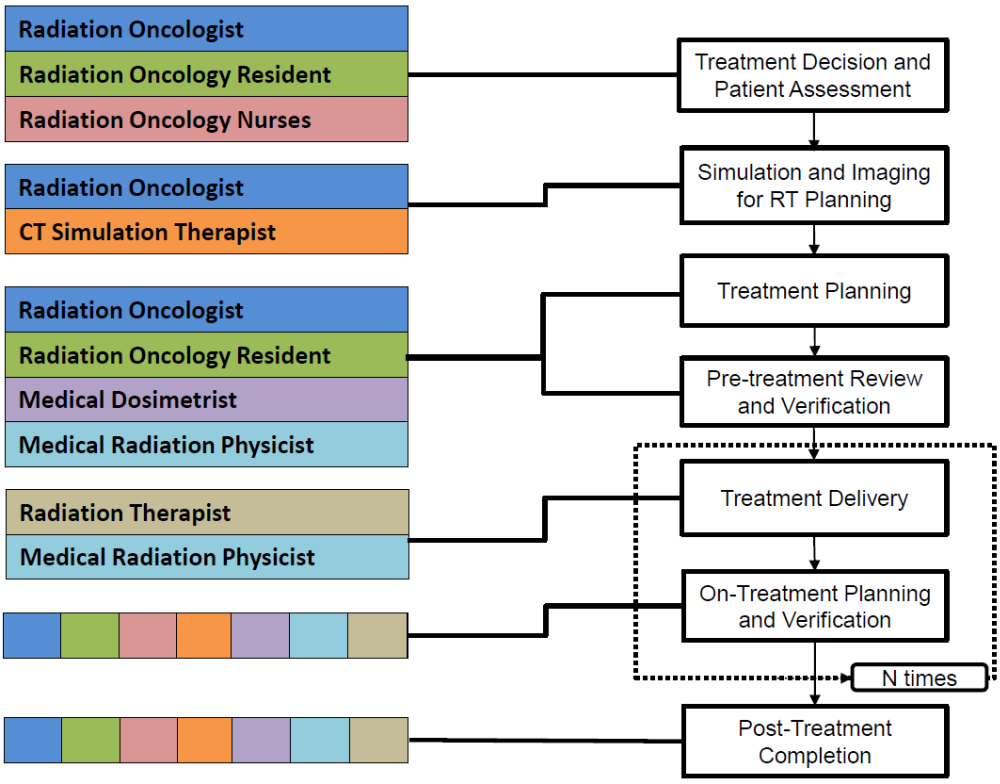


Figure 3.2: Healthcare providers involved in RT process

### 3.2.1 Safety Barriers

As a part of the quality assurance (QA) program, safety barriers (SB) are used throughout the process to prevent, control, or mitigate undesired errors. Each stage of the RT process consists of multiple steps which are either process steps or SBs. Each process step generates a set of *patient elements*. A patient element is defined as any information specific to a patient. For example, dose calculation is a process step that generates the following elements: (i) total dose (ii) fractional dose. Each SB checks one or more *elements* which means the tasks or outcomes associated with the elements are verified by the SB to ensure that the elements are accurate and they meet safety standards before the patient proceeds to the next process step. For example "dosimetry plan review" is a SB that checks the accuracy of total and fractional dose to make sure no mistake was made in the dose calculation step and the right amount of dose will be delivered to the patient.

We identified 12 steps in the the UNC RT process that generate a total of 65 elements. The

name of the first 15 elements are shown in Table 3.1. While each element is linked to a single process step, a process step may be associated with one or multiple elements. The 65 elements are checked by 10 safety barriers in the "pre-treatment" phase as shown in Table 3.2. The first two columns in Table 3.2 show the stages in the process and the process steps completed at each stage. The patient elements produced at each step are shown in the third column of Table 3.2 and the fourth column represents the SB performed at the end of each step. The person who is responsible for performing the SB is shown in the last column of Table 3.2. Figure 3.3 illustrates the position of SBs in the first two stages of UNC RT process. The box on the left shows that "treatment decision and patient assessment" consists of two process steps A and B and two safety barriers SB1 and SB2. Similarly, the box on the right shows that there are two process steps C and D and one safety barrier SB 3 in "simulation and imaging for RT planning". A SB can only check elements generated at previous process steps. For example, SB 2 can check elements produced at step A and step B while SB3 can check elements from steps A, B, C and D. Figure 3.4 shows examples of elements checked at each SB.

Table 3.1: Examples of Patient Element

| Element Number | Element Name |
|---|---|
| 1 | First Name |
| 2 | Last Name |
| 3 | Date of Birth |
| 9 | Treatment Site |
| 11 | IV Contrast |
| 15 | Treatment Side |
| 18 | Pacemaker |
| 19 | Pregnancy |
| 20 | Previous Treatment |
| 21 | Concurrent Chemo |
| 36 | Contours |
| 46 | Site Setup |
| 49 | Treatment Plan Note |
| 50 | Spatial Dose Distribution |
| 57 | Total Dose and Fractional Dose |

Table 3.2: SBs in UNC Department of Radiation Oncology

| Stage | Step | Elements | Safety Bariers(SB) | Person |
|---|---|---|---|---|
| Treatment Decision and Patient Assessment | A | 1-14 | SB1 | Nurse |
| | B | 15-25 | SB2 | Nurse/Phyisican |
| Simulation and Imaging for RT Planning | C | 26-29 | - | - |
| | D | 30-34 | SB3 | Therapist |
| Treatment Planning | E | 35-51 | SB4 | Dosimetrist |
| | F | - | SB5 | Sim Review |
| | G | 52-57 | SB6 | Dosimetrist |
| | H | 58-59 | SB7 | Physician |
| | I | - | - | - |
| | J | 60-64 | SB8 | Physicist |
| | K | - | SB9 | Therapist |
| Pre-Treatment Review and Verification | L | 65 | SB10 | Therapist |

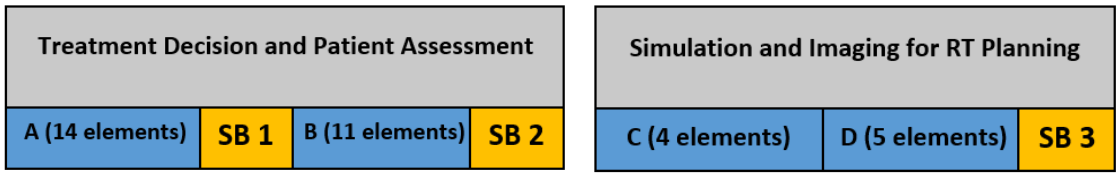


Figure 3.3: Process steps and SBs in the first two stages of UNC RT Process

## 3.3 Model Development

The simulation model was created using the simulation package SIMIO version 8.139. We focused on SBs in treatment planning and how they affect patient safety. The high-level process flow diagram representing the key stages in RT process was shown in Figure 3.1. Each patient completes seven planning and treatment stages. We built a model representing patient flow from arrival until treatment delivery. SBs are performed to catch potential errors before they reach the patient. Each SB is located after a process step. Figure 3.5 illustrates the patient flow between two consecutive non-SB process steps ($i$ and $(i+1)$) with a SB between them. At each step, with some probability an error occurs in one of the elements that are generated at the step. Although in reality, multiple errors may occur in multiple elements, for modeling

| Step | Element Name | SB 1 | SB 2 | SB 3 | SB 4 | SB 5 | SB 6 | SB 7 | SB 8 | SB 9 | SB 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| A | First Name | ■ | ■ | ■ | ■ | | ■ | | ■ | ■ | ■ |
| A | Last Name | ■ | ■ | ■ | ■ | | ■ | | ■ | ■ | ■ |
| A | DOB | ■ | ■ | ■ | | | | | | | |
| A | Insurance Info | ■ | | | | | | | | | |
| A | Appt. Verification | ■ | | | | | | | | | |
| A | MDs Info (RadOnc) | ■ | | | ■ | | ■ | | | | |
| A | Emergency Contact Info | ■ | | | | | | | | | |
| A | Patient Schedule | | | | ■ | | ■ | | | ■ | |
| A | Tx Site | | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |
| A | Path Report | | | | | | | | | | |
| A | IV Contrast | | ■ | ■ | | | | | | | |
| A | Patient Consent | | | ■ | | | | | ■ | | ■ |
| A | MR# | ■ | | ■ | ■ | | ■ | | ■ | | ■ |
| A | IMRT SBRT Preauth | | | | | | | | | ■ | ■ |
| B | Tx Side | | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |
| B | Vital Signs | | ■ | | | | | | | | |
| B | Lab Orders | | ■ | | | | | | | | |
| B | Pacemaker | | ■ | ■ | ■ | | ■ | ■ | ■ | | |
| B | Pregnancy | | ■ | ■ | ■ | | ■ | | | | |
| B | Previous Tx | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | | |
| B | Concurrent Chemo | | | | | ■ | | | | | |
| B | Creatinine | | ■ | ■ | | | | | | | |
| B | Allergies | | | ■ | | | | | | | |

Figure 3.4: Configuration of SBs in UNC RT Process

purposes we assume only one error occurs at a time. When the patient passes through a SB that checks the element, the SB may detect the error according to the some probability, i.e., (1-SB failure probability). Upon the detection of the error, the patient will repeat the step that generated the error. If the error was not detected, the patient proceeds to the next process step. We assume there is a SB failure probability associated with element $i$ which is the probability that a SB fails to catch an error in element $i$. For the simulation model, we assume that SB failure probabilities only depend on elements, meaning that the SB failure probability for a given element is equal at all SBs.

### 3.3.1 Input Data and Model Assumptions

To estimate the parameters for our simulation model, we took advantage of several resources at UNC Department of Radiation Oncology. In addition to historical data captured by the error reporting system, we sought input from care providers and clinical staff during several meetings.

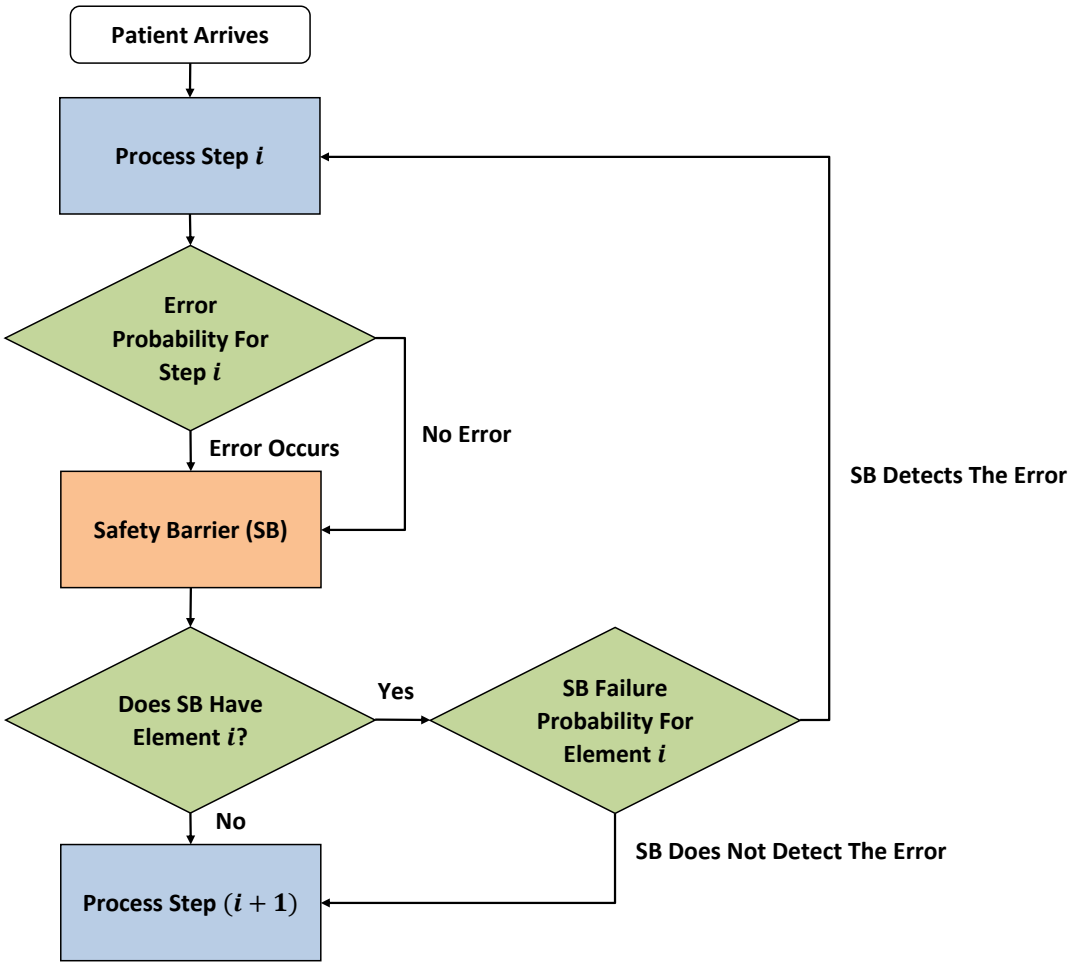


Figure 3.5: Patient Flow

### Arrival Process

Historical data indicates that on average 10 new patients arrive each day according to a non-homogeneous Poisson process. It is assumed that 2 patients arrive every hour from 7:00 am to 10:00 am and 1 patient arrive every hour from 10:00 am to 14:00 am. Table 3.3 shows the arrival rates.

### Error Probability

The error probability for each element is calculated using the historical error data. The error reporting system at UNC Department of Radiation Oncology reports, analyzes and manages er-

Table 3.3: Arrival Rate

| Time Interval | Patient Arrival Rate |
|---|---|
| 07:00 am- 08:00 am | 2 |
| 08:00 am- 09:00 am | 2 |
| 09:00 am- 10:00 am | 2 |
| 10:00 am- 11:00 am | 1 |
| 11:00 am- 12:00 pm | 1 |
| 12:00 pm- 01:00 pm | 1 |
| 01:00 pm- 02:00 pm | 1 |

rors that had or could potentially have a negative impact on patient safety and the care delivery process. Staff members are encouraged to report errors that reached patients and near-misses detected at different stages of the RT process. During monthly meetings, the staff review and investigate the reported errors. The investigation involves asking a series of key questions such as when the error occurred, when it was caught, which SBs failed to catch it and what factors caused the error. Answers to these questions help identify ineffective SBs and causes of errors so similar errors can be prevented in the future. Figure 3.6 shows an example of error analysis. The step where the error was initiated and the SB that caught the error are highlighted in red where the yellow cells show the SBs that failed to catch the error.

To estimate the error probabilities we analyzed the error data over a two year period: May 2012 - May 2014. Since the patient level error sample sizes were small, aggregate error probability for each step in the RT process were calculated. We assumed the error probability was uniformly distributed across the elements within that step, so we divided the error probability of each step by the number of elements that are generated at that step, to obtain the error probability at the element level. Table 3.4 shows the estimated error probabilities.

**SB Failure Probability**

As previously mentioned, for each element we define a SB failure probability that represents the probability that a SB fails to catch an error related to that element. We estimated the failure probabilities based on the workload associated with each element. A preliminary analysis was performed on the cognitive workload associated with patient elements to calculate a relative

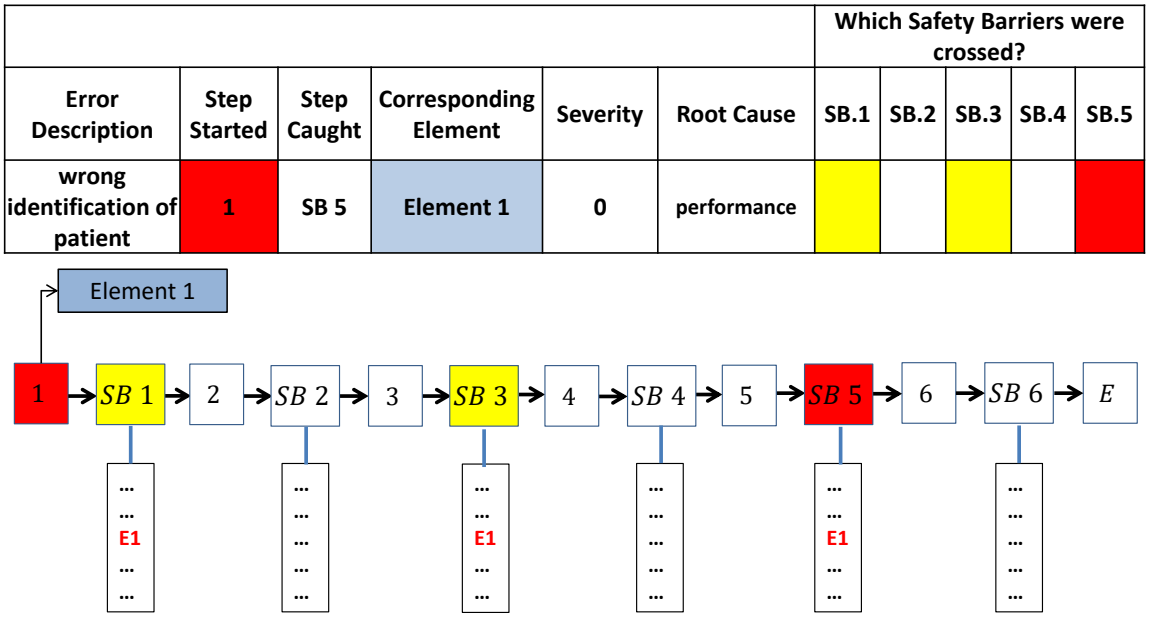


Figure 3.6: Error Reporting System

workload score for each element. We classified the elements into three groups of low, medium and high workload and assumed the elements with higher workload also have a higher failure probabilities. Based on these assumptions, we estimated the failure probability for each element type. In Section 3.5 we will describe the calculation of failure probabilities in more detail.

### Other Assumptions

Additionally, based on the data, the following assumptions are made to build the model:

- Error reporting rate is uniform throughout the year, therefore the estimated error probabilities are constant throughout the year.
- Error probability is the same for all the elements initiated at a given step
- Only one error at a time may occur for a patient
- Duration of each step follows a uniform distribution (we conducted a time study to estimate the times)
- For a given element, the failure probabilities are equal for all SBs.
- The time required for checking an element follows a uniform distribution [0.5, 2] minutes.

Table 3.4: Error probabilities in UNC RT process

| stage | steps that initiate elements | error probability at this step | number of elements | error probability for each element |
|---|---|---|---|---|
| Treatment Decision and Patient Assessment | A | 0.40% | 14 | 0.03% |
| | B | 1.10% | 11 | 0.01% |
| Simulation and Imaging for RT Planning | C | 0.77% | 4 | 0.19% |
| | D | 1.14% | 5 | 0.22% |
| Treatment Planning | E | 1.13% | 17 | 0.06% |
| | G | 1.56% | 6 | 0.26% |
| | H | 0.15% | 2 | 0.08% |
| | J | 1.23% | 5 | 0.25% |
| Pre-treatment Review and Verification | L | 0.39% | 1 | 0.39% |

## 3.4 Model Verification

To ensure that the model is implemented correctly and works as intended, we checked the model components after each step in model development and made the necessary adjustments. Furthermore, we analyzed several scenarios considering a single patient element. For each scenario, we ran the model for 20 replications with a run length of 6 months. In each scenario, we measured the expected rate of errors that reach patients ($\mathbf{E}[RE_P]$) for an element with a given error probability. $\mathbf{E}[RE_P]$ was calculated by dividing the total expected number of errors that reach patients ($\mathbf{E}[NE_P]$) by the total number of patients who complete treatment planning. We compared $\mathbf{E}[RE_P]$ across the three scenarios with error probabilities of 0.002, 0.01 and 0.05. In each scenario we also varied the SB failure probability from 0% to 100% using 10% increments and examined the impact on $\mathbf{E}[RE_P]$. Tables 3.5, 3.6, and 3.7 provide the results for three scenarios. The results indicate that adding an element to additional SBs or decreasing SB failure probabilities, results in lower $\mathbf{E}[RE_P]$. Moreover, if we compare the $\mathbf{E}[RE_P]$ across the three tables, we observe that for a given number of SBs and SB failure probability for an element, as the error probability increases, errors reach patients at a higher rate. The results demonstrate that the model implementation is accurate and matches the conceptual description we presented.

Table 3.5: $\mathbf{E}[RE_P]$ for element $i$ with error probability = 0.002

| SB failure probability | Number of safety barriers for a given element $i$ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0% | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10% | 0.02% | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20% | 0.05% | 0.01% | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 30% | 0.06% | 0.02% | 0.01% | 0 | 0 | 0 | 0 | 0 | 0 |
| 40% | 0.08% | 0.03% | 0.02% | 0.01% | 0 | 0 | 0 | 0 | 0 |
| 50% | 0.10% | 0.05% | 0.03% | 0.01% | 0.01% | 0 | 0 | 0 | 0 |
| 60% | 0.13% | 0.08% | 0.05% | 0.03% | 0.02% | 0.01% | 0 | 0 | 0 |
| 70% | 0.14% | 0.10% | 0.07% | 0.05% | 0.04% | 0.03% | 0.02% | 0.01% | 0.01% |
| 80% | 0.16% | 0.13% | 0.12% | 0.08% | 0.07% | 0.06% | 0.04% | 0.03% | 0.03% |
| 90% | 0.18% | 0.18% | 0.14% | 0.13% | 0.12% | 0.10% | 0.11% | 0.09% | 0.08% |
| 100% | 0.20% | 0.21% | 0.20% | 0.20% | 0.19% | 0.19% | 0.20% | 0.20% | 0.20% |

Table 3.6: $\mathbf{E}[RE_P]$ for element $i$ with error probability = 0.01

| SB failure probability | Number of safety barriers for a given element $i$ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0% | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10% | 0.10% | 0.01% | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20% | 0.24% | 0.04% | 0.01% | 0 | 0 | 0 | 0 | 0 | 0 |
| 30% | 0.32% | 0.10% | 0.04% | 0.01% | 0 | 0 | 0 | 0 | 0 |
| 40% | 0.42% | 0.16% | 0.08% | 0.04% | 0.01% | 0 | 0 | 0 | 0 |
| 50% | 0.52% | 0.26% | 0.14% | 0.06% | 0.04% | 0.02% | 0.01% | 0 | 0 |
| 60% | 0.64% | 0.38% | 0.26% | 0.14% | 0.10% | 0.06% | 0.02% | 0.01% | 0 |
| 70% | 0.72% | 0.50% | 0.34% | 0.26% | 0.18% | 0.14% | 0.08% | 0.04% | 0.03% |
| 80% | 0.80% | 0.66% | 0.58% | 0.38% | 0.34% | 0.28% | 0.20% | 0.16% | 0.14% |
| 90% | 0.92% | 0.88% | 0.72% | 0.66% | 0.62% | 0.52% | 0.54% | 0.46% | 0.42% |
| 100% | 1% | 1.03% | 1.02% | 1.02% | 0.97% | 0.96% | 0.98% | 1% | 1.02% |

Table 3.7: $\mathbf{E}[RE_P]$ for element (i) with error probability = 0.05

| SB failure probability | Number of safety barriers for a given element $i$ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0% | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10% | 0.50% | 0.06% | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20% | 1.20% | 0.20% | 0.04% | 0 | 0 | 0 | 0 | 0 | 0 |
| 30% | 1.60% | 0.50% | 0.20% | 0.06% | 0.02% | 0.01% | 0 | 0 | 0 |
| 40% | 2.10% | 0.80% | 0.40% | 0.20% | 0.06% | 0.02% | 0.01% | 0 | 0 |
| 50% | 2.60% | 1.30% | 0.70% | 0.30% | 0.20% | 0.08% | 0.07% | 0.02% | 0 |
| 60% | 3.20% | 1.90% | 1.30% | 0.70% | 0.50% | 0.30% | 0.09% | 0.07% | 0 |
| 70% | 3.60% | 2.50% | 1.70% | 1.30% | 0.90% | 0.70% | 0.40% | 0.20% | 0.18% |
| 80% | 4.00% | 3.30% | 2.90% | 1.90% | 1.70% | 1.40% | 1.00% | 0.80% | 0.70% |
| 90% | 4.60% | 4.40% | 3.60% | 3.30% | 3.10% | 2.60% | 2.70% | 2.30% | 2.10% |
| 100% | 5.00% | 5.20% | 5.10% | 5.10% | 4.80% | 4.80% | 4.90% | 5.00% | 5.10% |

## 3.5 Sensitivity Analysis and Model Validation

In this section we present the sensitivity analysis results on some of the model inputs. First, we focused on SB failure probabilities and examined the performance measures to determine the failure probability for each element. Using the probabilities from the analysis of failure probabilities, we validated the model and then conducted the sensitivity analysis on the error probabilities.

### 3.5.1 SB Failure Probability

As mentioned previously, we considered three tiers of failure probabilities based on the workload associated with elements. The low workload is defined as the base case and other tiers are compared to it. It is assumed that the failure probability for elements with medium workload is 15% higher than the failure probability for elements with low workload (base case) and the the failure probability for elements with high workload is 15% higher than the failure probability of elements with medium workload. Table 3.8 shows the calculation of failure probability for each type of element. There was limited information regarding the SB failure probabilities for elements, so we tested several scenarios to determine failure probabilities that accurately represent the RT process at UNC. We varied the failure probability of low-workload elements from 0.1 to 0.7 in increments of 0.1 and calculated the failure probabilities for medium and high

Table 3.8: Effect of element workload on SB failure probability

| Element Workload | Failure Probability |
|---|---|
| low workload (e.g., patient's name) | low failure probability (base case) |
| medium workload (e.g., treatment site) | low failure probability + 15% |
| high workload (e.g., dose distribution) | medium failure probability + 15% |

workload elements accordingly. The failure probabilities associated with each scenario are listed in Table 3.10. For each scenario, we measured patient throughput, the total expected number of errors that occurred ($\mathbf{E}[NE]$), the total expected number of errors that reached patients ($\mathbf{E}[NE_P]$) and the expected ratio of errors that reached patients ($\mathbf{E}[RE_P]$). The summary of performance measures is provided in Table 3.9. We can see that some measures such as $\mathbf{E}[NE]$ and patient throughput were nearly unchanged across different scenarios. Comparing $\mathbf{E}[NE]$ and throughput with historical data on patient volume and errors, shows the numbers are within acceptable ranges. Performance measures $\mathbf{E}[RE_P]$ and $\mathbf{E}[NE_P]$ are largely driven by the failure probabilities. Figure 3.7 illustrates the $\mathbf{E}[NE_P]$ associated with the seven scenarios in a box plot. We conservatively choose the first scenario as the base case for our model. Based on similar rates reported by other researchers [10, 128], the $\mathbf{E}[NE_P]$ for the first scenario can be viewed as a lower bound for actual $\mathbf{E}[NE_P]$. In addition, the failure probabilities and performance measures associated with this scenario are verified by UNC clinical staff as reasonable.

Table 3.9: Performance Measures

| performance measure | definition |
|---|---|
| $\mathbf{E}[NE]$ | expected number of errors that occurred |
| $\mathbf{E}[NE_P]$ | expected number of errors that reached patients |
| $\mathbf{E}[RE_P]$ | expected rate of errors that reached patients : $\frac{\mathbf{E}[NE_P]}{\text{patient throughput}}$ |

It is also interesting to evaluate the effect the varying failure probabilities on performance measures assuming all SBs have the same failure probability regardless of the element workload. Table 3.11 shows the results for this case. From the table we can see that increasing failure probability has much less significant impact on $\mathbf{E}[NE_P]$ and $\mathbf{E}[RE_P]$ compared to the case with non-uniform failure probabilities.

Table 3.10: Sensitivity analysis on SB failure probabilities (non-uniform case)

| Scenarios | Element Tiers | Failure Probabilities | Throughput | $\mathbf{E}[NE]$ | $\mathbf{E}[NE_P]$ | $\mathbf{E}[RE_P]$ |
|---|---|---|---|---|---|---|
| Scenario 1 | tier 1 | 0.1 | 1791.1(13.2) | 183.4(5) | 24.1(1.7) | 0.13(0.008) |
| | tier 2 | 0.115 | | | | |
| | tier 3 | 0.13225 | | | | |
| Scenario 2 | tier 1 | 0.2 | 1809.4(12.1) | 182.83(3.7) | 33.2(1.8) | 0.18(0.009) |
| | tier 2 | 0.23 | | | | |
| | tier 3 | 0.2645 | | | | |
| Scenario 3 | tier 1 | 0.3 | 1800.9(14.2) | 185.6(4.4) | 47.5(2.6) | 0.26(0.01) |
| | tier 2 | 0.345 | | | | |
| | tier 3 | 0.39675 | | | | |
| Scenario 4 | tier 1 | 0.4 | 1798.3(14.7) | 183.9(5.4) | 63.9(2.6) | 0.35(0.009) |
| | tier 2 | 0.46 | | | | |
| | tier 3 | 0.529 | | | | |
| Scenario 5 | tier 1 | 0.5 | 1804.1(15.2) | 183.8(5.2) | 83.2(3.3) | 0.45(0.01) |
| | tier 2 | 0.575 | | | | |
| | tier 3 | 0.66125 | | | | |
| Scenario 6 | tier 1 | 0.6 | 1791.1(15.6) | 176.7(4.7) | 108.7(3.8) | 0.62(0.01) |
| | tier 2 | 0.69 | | | | |
| | tier 3 | 0.7935 | | | | |
| Scenario 7 | tier 1 | 0.7 | 1793.6(15.4) | 177.7(3.8) | 143.9(3.5) | 0.81(0.01) |
| | tier 2 | 0.805 | | | | |
| | tier 3 | 0.92575 | | | | |

Numbers in parentheses represent a 95% confidence interval

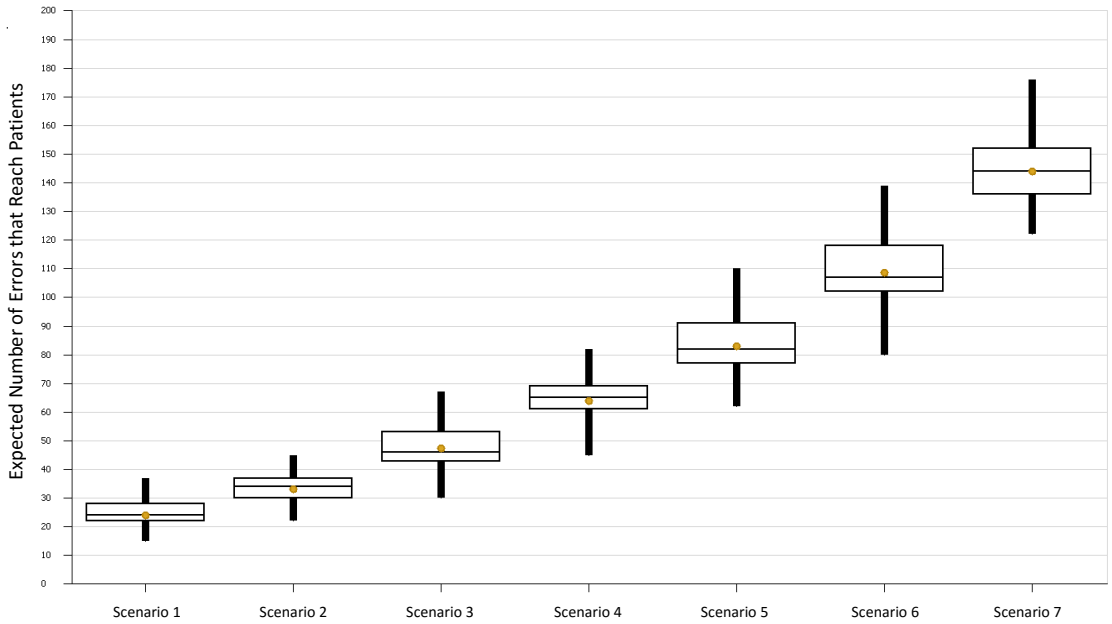


Figure 3.7: $\mathbf{E}[NE_P]$ as a function of SB failure probability

Table 3.11: Sensitivity analysis on SB failure probabilities (uniform case)

| Scenarios | Element Tiers | Failure Probabilities | Throughput | $\mathbf{E}[NE]$ | $\mathbf{E}[NE_P]$ | $\mathbf{E}[RE_P]$ |
|---|---|---|---|---|---|---|
| Scenario 1 | tier 1 | 0.1 | 1792.8(12.7) | 186.5(5.1) | 21.8(1.6) | 0.12(0.007) |
| | tier 2 | 0.1 | | | | |
| | tier 3 | 0.1 | | | | |
| Scenario 2 | tier 1 | 0.2 | 1802.8(16.8) | 187.1(4.5) | 29.1(1.9) | 0.15(0.009) |
| | tier 2 | 0.2 | | | | |
| | tier 3 | 0.2 | | | | |
| Scenario 3 | tier 1 | 0.3 | 1789.7(14.9) | 183.9(4.6) | 37.2(2.0) | 0.20(0.01) |
| | tier 2 | 0.3 | | | | |
| | tier 3 | 0.3 | | | | |
| Scenario 4 | tier 1 | 0.4 | 1795.5(11.1) | 182.3(4.3) | 48.1(2.7) | 0.26(0.008) |
| | tier 2 | 0.4 | | | | |
| | tier 3 | 0.4 | | | | |
| Scenario 5 | tier 1 | 0.5 | 1788.7(11.2) | 179.4(4.3) | 59.8(2.3) | 0.33(0.01) |
| | tier 2 | 0.5 | | | | |
| | tier 3 | 0.5 | | | | |
| Scenario 6 | tier 1 | 0.6 | 1794.8(13.3) | 183.7(4.8) | 73.1(2.8) | 0.40(0.009) |
| | tier 2 | 0.6 | | | | |
| | tier 3 | 0.6 | | | | |
| Scenario 7 | tier 1 | 0.7 | 1788.4(12.5) | 177.5(3.1) | 88.9(3.5) | 0.50(0.01) |
| | tier 2 | 0.7 | | | | |
| | tier 3 | 0.7 | | | | |

Numbers in parentheses represent a 95% confidence interval

### 3.5.2 Error Probability

To investigate the effect of error probability on model outputs, we varied the error probabilities for all elements between 0%-100% with 10% increments and measured the performance measures used in previous analysis as well as the flow time. The results are summarized in Table 3.12. Although, increasing error probabilities results in higher rate and number of errors that reach patients ($\mathbf{E}[NE_P]$ and $\mathbf{E}[RE_P]$), the impact is not as significant as increasing SB failure probabilities. The effect of error probabilities is illustrated in Figure 3.8. In fact, varying error probabilities by 10%, affects $\mathbf{E}[NE_P]$ and $\mathbf{E}[RE_P]$ by up to only 7%. This is because changing SB failure probabilities for a given element will influence the performance of all SBs that check the element and as a result, the rate and number of errors that reach patients change significantly.

The impact of error probabilities on flow time is shown in Figure 3.9. We hypothesized that increasing error probabilities results in larger flow times, simply because rework happens more frequently to correct the errors and the average amount of time spent for each patient increases. The plots shown in Figure 3.9 confirm our hypothesis. Comparing Figures 3.9 and 3.8 also indicates that flow time exhibits relatively low variability compared to $\mathbf{E}[NE_P]$.

Table 3.12: Sensitivity analysis on error probabilities

| Scenarios | Error Probabilities | Throughput | $\mathbf{E}[NE]$ | $\mathbf{E}[NE_P]$ | $\mathbf{E}[RE_P]$ | Flow time |
|---|---|---|---|---|---|---|
| Base case | +0% | 1791.2 (13.2) | 183.4 (4.9) | 24.1 (1.6) | 0.13 (0.008) | 16.07 (0.01) |
| Scenario 1 | +10% | 1802.6 (14.9) | 199.5 (5.7) | 25.3 (1.5) | 0.13 (0.007) | 16.11 (0.01) |
| Scenario 2 | +20% | 1795 (12.8) | 222.2 (5.4) | 28.6 (2.0) | 0.13 (0.008) | 16.15 (0.01) |
| Scenario 3 | +30% | 1794.9 (17.6) | 243.8 (6.2) | 32.3 (1.9) | 0.13 (0.007) | 16.18 (0.01) |
| Scenario 4 | +40% | 1787.9 (14.5) | 258.1 (4.9) | 34.8 (2.1) | 0.13 (0.008) | 16.20 (0.01) |
| Scenario 5 | +50% | 1804.4 (14.7) | 277.4 (5.9) | 37.0 (1.7) | 0.13 (0.006) | 16.23 (0.01) |
| Scenario 6 | +60% | 1818.8 (12.8) | 303.1 (4.9) | 39.6 (2.0) | 0.13 (0.006) | 16.27 (0.01) |
| Scenario 7 | +70% | 1809.5 (13.4) | 314.1 (6.1) | 40.1 (1.9) | 0.13 (0.006) | 16.30 (0.01) |
| Scenario 8 | +80% | 1803.2 (13.1) | 334.6 (6.5) | 44.1 (2.3) | 0.13 (0.006) | 16.32 (0.01) |
| Scenario 9 | +90% | 1809 (12.8) | 357.7 (6.4) | 46.5 (1.9) | 0.13 (0.005) | 16.37 (0.01) |
| Scenario 10 | +100% | 1804.2 (13.4) | 376.1 (6.9) | 48.4 (2.3) | 0.13 (0.006) | 16.40 (0.01) |

Numbers in parentheses represent a 95% confidence interval

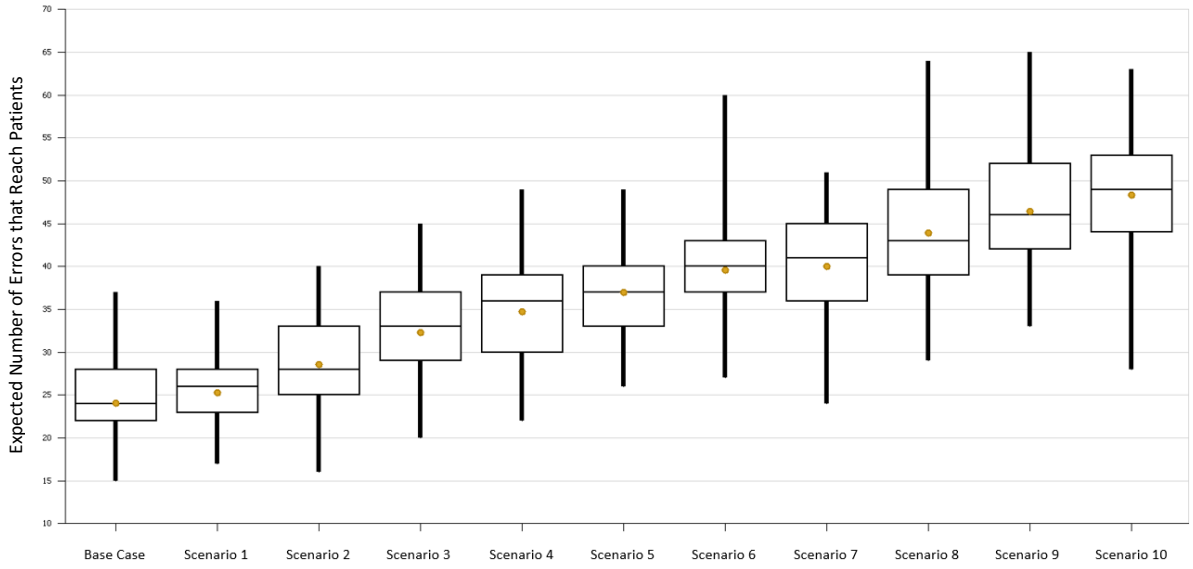


Figure 3.8: $\mathbf{E}[NE_P]$ as a function of error probability

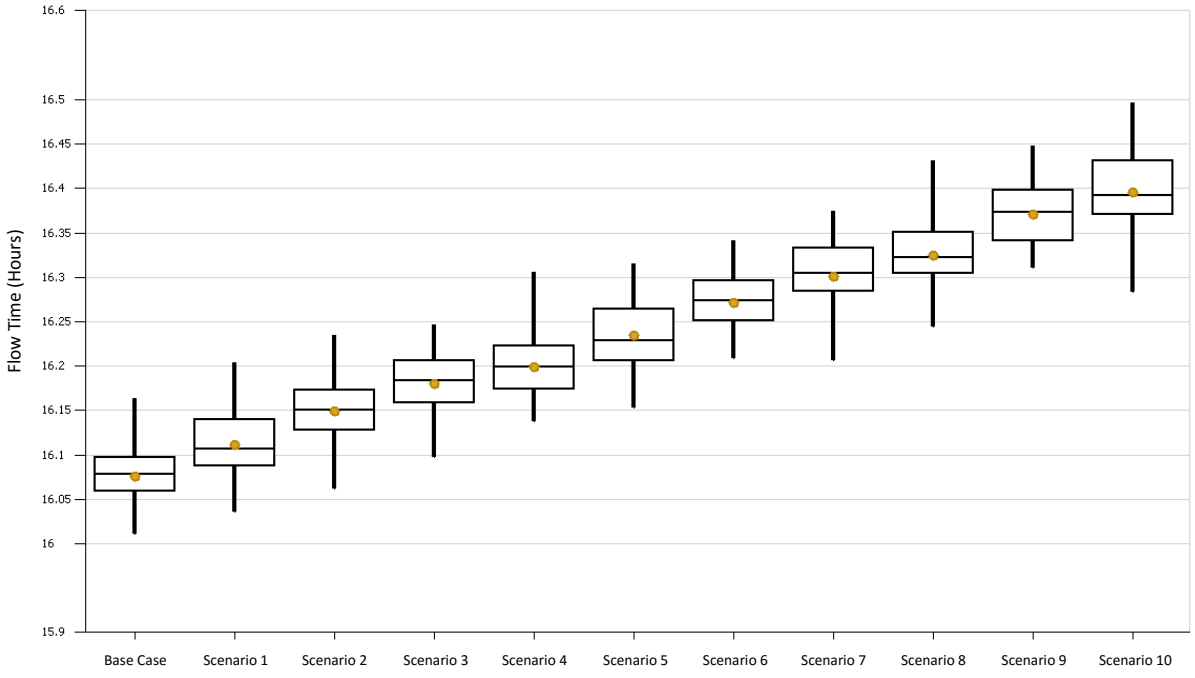


Figure 3.9: Flow Time as a function of error probability

## 3.6 What-if Analysis

In this section we present the results of a what-if analysis under several scenarios. The purpose is to evaluate the effect of changes to the current SB design, such as adding more SBs or changing their composition. First, we considered a scenario where we added a SB at end of treatment planning, assuming this SB checks all patient elements. In the second scenario we considered adding two SBs that both include all elements. Table 3.13 summarizes the results for the base case (no additional SB) and the two scenarios we described. According to the results, adding the first SB reduces $\mathbf{E}[NE_P]$ and $\mathbf{E}[RE_P]$ by 83% and if we add two SBs, $\mathbf{E}[NE_P]$ and $\mathbf{E}[RE_P]$ improves by another 10%. Figure 3.10 shows the $\mathbf{E}[NE_P]$ in a box plot format. We also calculated the total checking time for each scenario and the results indicate that each new SB increases the total checking time per patient by about 1.2 hours.

Table 3.13: What-if analysis - Case 1: additional SB with all elements

| Scenario | Throughput | $\mathbf{E}[NE]$ | $\mathbf{E}[NE_P]$ | $\mathbf{E}[RE_P]$ | Flow Time | Checking Time |
|---|---|---|---|---|---|---|
| Base case | 1802.9 (14.10) | 185.4 (4.73) | 24.25 (1.73) | 0.13 (0.008) | 19.19 (0.01) | 4.76 (0.005) |
| Scenario 1 | 1792.6 (14.25) | 187.9 (4.72) | 3.95 (0.71) | 0.02 (0.004) | 20.53 (0.01) | 6.01 (0.006) |
| Scenario 2 | 1791.2 (16.25) | 185.1 (5.57) | 1.5 (0.36) | 0.01 (0.002) | 21.72 (0.02) | 7.28 (0.007) |

Numbers in parentheses represent a 95% confidence interval

Although adding SBs that verify all elements, results in significantly reduced $\mathbf{E}[NE_P]$ in the model, we caution that improvement may not be as great in real world. The reason is that the reliability of SBs is strongly affected by the workload associated with the elements they check [83]. Adding too many elements to a SB leads to an excessive workload and higher workload increases the chance of the SB failing to catch errors (larger failure probability). To avoid this issue, we consider adding SBs that only include a subset of of elements instead of checking all of them. Patient elements are not equal in terms of their contribution to $\mathbf{E}[NE_P]$. The histogram shown in Figure 3.11 provides the breakdown of $\mathbf{E}[NE_P]$ by elements. As can be seen in the figure, elements 24, 25, 28, 48, 59 and 61 have the highest $\mathbf{E}[NE_P]$ which are highlighted in

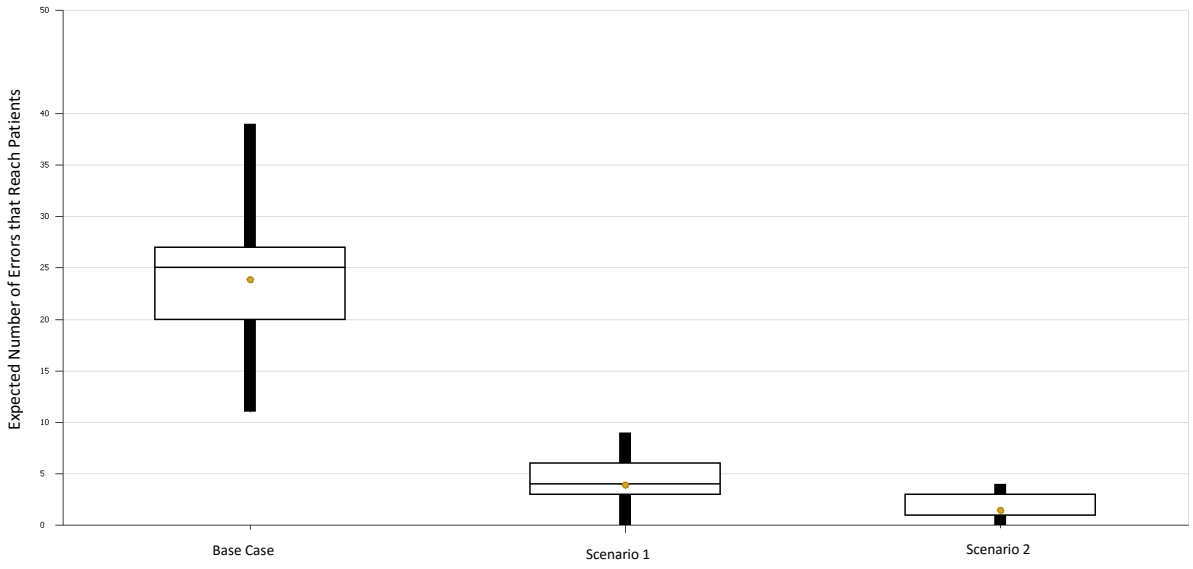


Figure 3.10: $\mathbf{E}[NE_P]$ box plot - Case 1: additional SB with all elements

red. The name of these elements and the stage they are generated in are shown in Table 3.14. In the next set of what-if scenarios, we examined the effect of adding SBs that only check the top six critical elements. Again, in the first scenario we considered only one additional SB and in the second scenario we ran the model assuming two additional SBs at the end of treatment planning. The results shown in Table 3.15, indicate that adding one SB for six critical elements, decreases the $\mathbf{E}[NE_P]$ and $\mathbf{E}[RE_P]$ by 60% and the second SB, reduces $\mathbf{E}[RE_P]$ by only an additional 3%. The SBs with 6 elements have only a slightly less significant impact on $\mathbf{E}[NE_P]$ compared to SBs that check all elements. However it must be noted that we are not taking into account the impact of workload on SB failure probabilities. Furthermore, in the case of adding SBs with fewer elements, the checking time invested per patient is much less than the time required to check all elements. In fact, the checking time associated with each new SB with six elements is less than 10 minutes per patient meaning that investing a few more minutes for each patient can have a significant impact on $\mathbf{E}[NE_P]$.

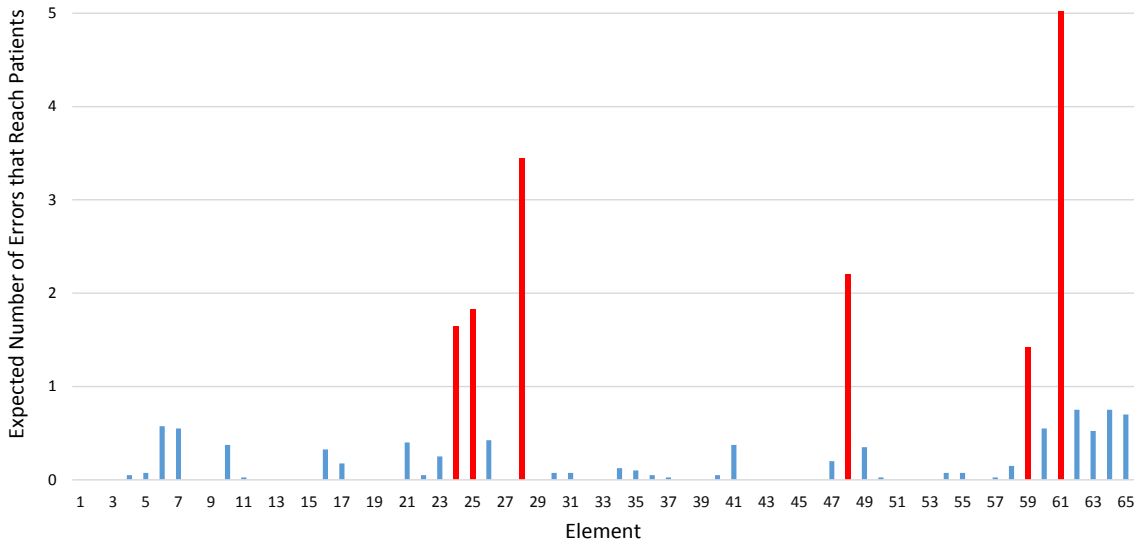


Figure 3.11: $\mathbf{E}[NE_P]$ histogram by element

Table 3.14: Elements with maximum $\mathbf{E}[NE_P]$

| Element Number | Element Name | Stage |
|---|---|---|
| 24 | Diabetes | Simulation and Imaging for RT Planning |
| 25 | Hypertension | Simulation and Imaging for RT Planning |
| 28 | Imaging Note | Simulation and Imaging for RT Planning |
| 48 | Treatment Field | Treatment Planning |
| 59 | Ports Approved by MD | Treatment Planning |
| 61 | IMRT QA Analysis | Treatment Planning |

Table 3.15: What-if analysis - Case 2: additional SB with top six elements

| Scenario | Throughput | $\mathbf{E}[NE]$ | $\mathbf{E}[NE_P]$ | $\mathbf{E}[RE_P]$ | Flow Time | Checking Time |
|---|---|---|---|---|---|---|
| Base case | 1802.9 (14.10) | 185.4 (4.73) | 24.25 (1.73) | 0.13 (0.008) | 19.19 (0.01) | 4.76 (0.005) |
| Scenario 1 | 1793.1 (13.25) | 189.4 (3.28) | 9.68 (1.02) | 0.05 (0.005) | 19.31 (0.01) | 4.81 (0.004) |
| Scenario 2 | 1809.6 (16.73) | 190.7 (5.27) | 8.93 (1.04) | 0.05 (0.005) | 19.42 (0.01) | 4.96 (0.005) |

Numbers in parentheses represent a 95% confidence interval

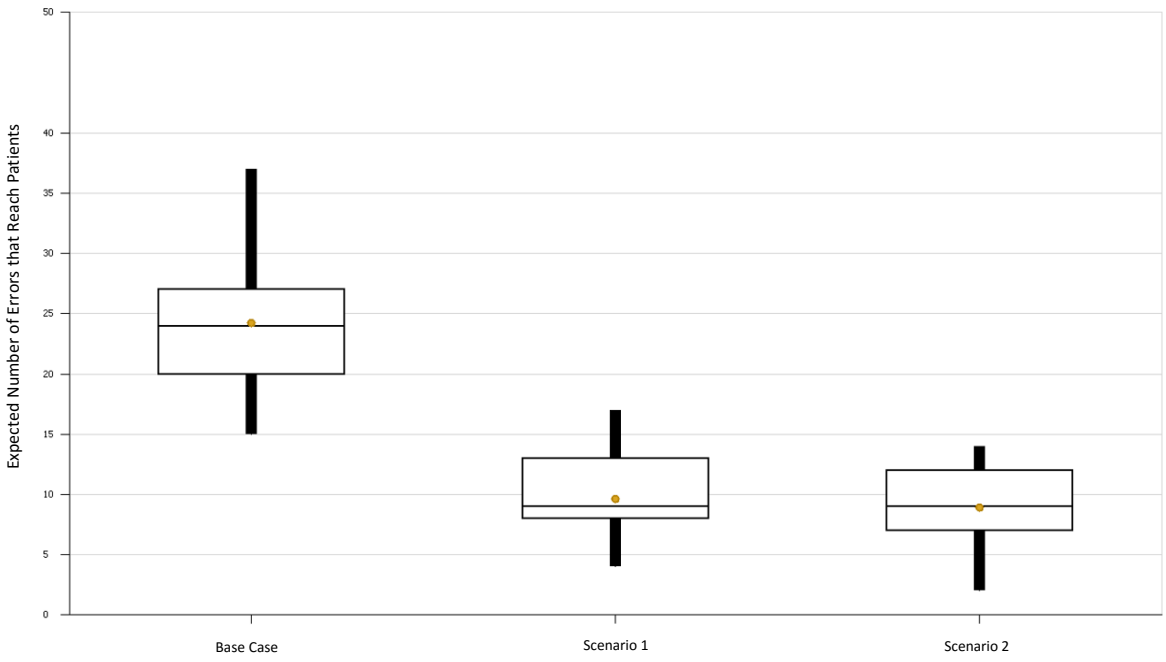


Figure 3.12: $\mathbf{E}[NE_P]$ box plot - Case 2: additional SB with top six elements

## 3.7 Conclusion

In this chapter we developed a discrete-event simulation to assess reliability and patient safety in RT process. We studied SBs in treatment planning and measured their impact on patient safety and RT workflow under several scenarios. Using sensitivity analysis, we investigated the effect of some input parameters such as error probabilities and SB failure probabilities on safety-related outcomes including the expected number of errors that reach patients. Our results indicate that the model outputs are more sensitive to changes in the SB failure probabilities. Furthermore, we used DES model to evaluate modifications to SB design and investigate various what-if scenarios. The results demonstrate that error prevention and detection can be designed into the process through an effective combination of SBs to reduce the chance of harm to patients. Although, the simulation model proved to be an effective tool for evaluating proposed changes to the process and identifying areas of improvement, the model does not provide optimal design of SBs by answering questions such as where SBs should be located? which combination of SBs should be used? or which elements should be checked at each SB? To answer these questions, we develop mathematical models based on our findings from the simulation model. We use the

mathematical models to determine the optimal number, location and composition of SBs taking into account their implementation costs and their impact on patient safety.

# Chapter 4

# Deterministic Models

In this chapter we present mathematical models developed for reliable design of safety barriers (SB) in radiation therapy (RT). Specifically, we develop models to determine the optimal design (i.e. number, location and composition) of SBs to improve patient safety considering SB implementation costs. This chapter is outlined as follows: After the introduction and problem description in Section 4.1, we present a mixed integer programming model in Section 4.2. In Section 4.3, we review the literature related to our work. In Section 4.4, we present a special case of the proposed model and investigate its structural properties. We describe solution algorithms for our problem in Section 4.5 and discuss computational results and sensitivity analysis in Section 4.6. Section 4.7 concludes this chapter and discusses future research.

## 4.1 Problem Description

The radiation therapy (RT) process consists of seven stages as described in Chapter 3. Each stage of the RT process consists of multiple steps which are either process steps or safety barriers (SBs). Each process step generates a set of *patient elements*. A patient element is defined as any information specific to a patient. For example, dose calculation is a process step that generates the following elements: (i) total dose and (ii) fractional dose. Each SB checks one or more *elements* which means the tasks or outcomes associated with the elements are verified by the SB to ensure that the elements are accurate and they meet safety standards before the patient proceeds to the next process step. For example the "the dosimetry plan review" is a SB that

checks the accuracy of the total and fractional dose to make sure the correct amount of dose will be delivered to the patient.

Analyzing the pre-treatment phase of RT process at our partner radiation oncology clinic in UNC Lineberger Comprehensive Cancer Hospital, we identified 11 process steps that generate a total of 65 patient elements. These elements are verified by 10 SBs. For example, we represent the RT reliable design problem using a sequential process that consists of three process steps as shown in Figure 4.1. Let $\mathcal{I} = \{1, \ldots, I\}$ be the set of elements generated in the RT process. There is a candidate SB step following each process step. Let $\mathcal{M}_i$ be the set of all candidate SBs that can check element $i$. For example, in Figure 4.1, $\mathcal{M}_A = \{\text{SB1, SB2, SB3, Patient}\}$ and $\mathcal{M}_E = \{\text{SB3, Patient}\}$. The checks of an element are assumed to be imperfect, therefore an element may be checked in multiple SBs. Each check of an element by a SB corresponds to a *safety level* for that element, i.e., the first check is the first safety level, the second check is the second safety level, and so on. The collection of safety levels for an element impacts the rework cost. The maximum number of safety levels for element $i$ is equal to $|\mathcal{M}_i|$. For example, in Figure 4.1, $|\mathcal{M}_A|$ is 4 since element A can be checked in SB1, SB2 and SB3. We denote the set of safety levels for element $i$ by $\mathcal{K}_i = \{1, \ldots, |\mathcal{M}_i|\}$.

The problem is to determine the locations of a given number of SBs and assign a set of elements to be checked in each SB. In a feasible solution, element $i$ can be checked in SB $j \in \mathcal{M}_i$ at safety level $k \in \mathcal{K}_i$, only if $k-1$ SBs check that element before SB $j$. For example, in Figure 4.1, element A can be checked in SB3 at safety level 2 given that either SB1 or SB2 checks element A at the first safety level. However, element A can not be checked in SB2 at safety level 3 since there is only one SB (i.e., SB1) that can check element A before SB2. The highest possible safety level for element $i$ in SB $j$ in denoted by $\rho_{ij}$. In Figure 4.1, the highest possible safety level for element D in SB3, for example, is equal to two ($\rho_{D,SB3} = 2$). Thus, element D can be checked in SB3 at safety level 1 or 2, but not at any higher safety level. Intuitively, if element $i$ is checked in SB $j$ at safety level $k \leq \rho_{ij}$, then it cannot be checked in SB $\ell$ at any safety level that is less than $k$ if $\rho_{i\ell} > \rho_{ij}$. In Figure 4.1, for example, if element B is checked in SB2 at safety level 2, then it cannot be checked in SB3 at safety level 1.

Let $t_{ij}$ denote the cost of checking element $i$ in SB $j$, which corresponds to the time required to verify element $i$ in SB $j$. This cost is assumed to equal zero in the last SB ($t_{i,J} = 0$), because

the last SB $J$ is the patient who detects all errors without checking. Moreover, let $r_{ij}$ denote the rework cost, i.e., the amount of time required to correct the RT process, if an error related to element $i$ is detected in SB $j$. The earlier the error is detected, the lower the rework cost of correcting it will be. If an error reaches the patient (i.e., error is not detected by any of the SBs before the patient), then correcting it is not possible and the rework cost $r_{i,J}$ corresponds to the penalty associated with the medical consequences caused by delivering improper radiation.

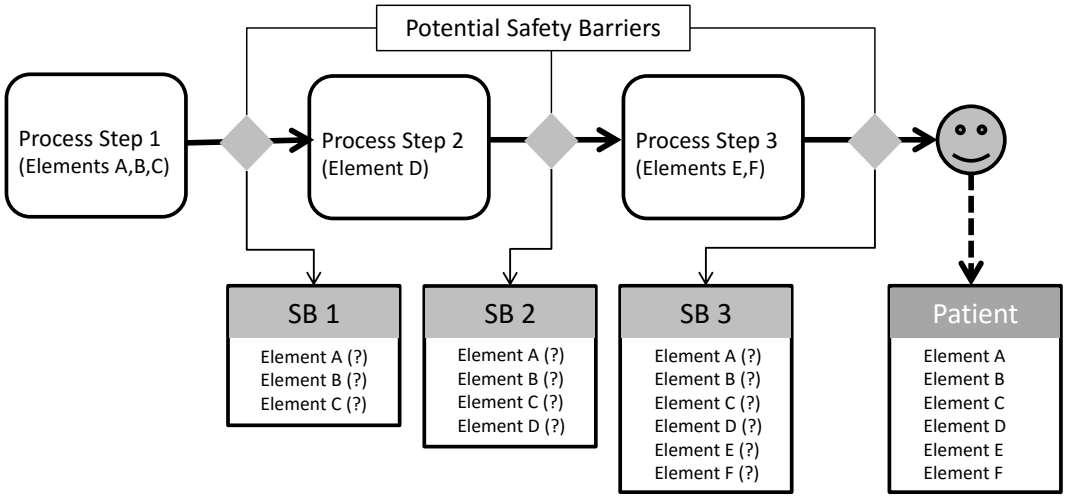


Figure 4.1: In this sample care path example, process step 1 generates elements A,B,C; process step 2 generates element D; and process step 3 generates elements E,F. A SB can be placed after each process step. Each SB checks one or more elements generated in previous process steps. An element may be checked multiple times in different SBs. The patient is considered to be the last SB in the RT process that checks all elements.

## 4.2 Mixed Integer Programming (MIP) Model for Reliable Radiation Therapy Problem (RRTP)

We formulate a mathematical programming model to $(i)$ determine the locations of a given number of SBs in the RT process, and $(ii)$ assign the set of elements to be checked in each SB. An error related to element $i$ occurs with probability $\gamma_i$. If a SB does not check element $i$, it cannot catch errors in $i$. If a SB checks element $i$, it may still fail to detect an error in $i$. This

Table 4.1: Summary of Notation

| | |
|---|---|
| **Indices** | |
| $\mathcal{I}$ | set of elements to be checked, $\{1, \ldots, I\}$, |
| $\mathcal{J}$ | set of candidate safety barriers (SBs), $\{1, \ldots, J\}$, (SB $J$ is the patient) |
| $\mathcal{M}_i \subseteq \mathcal{J}$ | set of SBs that can check element $i \in \mathcal{I}$, |
| $\mathcal{K}_i$ | set of safety levels for element $i$, $\{1, \ldots, \lvert\mathcal{M}_i\rvert\}$ |
| **Parameters** | |
| $b$ | number of available SBs to be opened |
| $\gamma_i$ | the probability of error in element $i \in \mathcal{I}$, |
| $t_{ij}$ | time required to check and verify element $i \in \mathcal{I}$ in SB $j \in \mathcal{M}_i$ |
| $r_{ij}$ | rework cost if an error in element $i \in \mathcal{I}$ detected in SB $j \in \mathcal{M}_i$ |
| $\beta_{ij}$ | probability of failing to detect an error in element $i \in \mathcal{I}$ in SB $j \in \mathcal{M}_i$ |
| $\rho_{ij}$ | the highest possible safety level for element $i \in \mathcal{I}$ in SB $j \in \mathcal{M}_i$ |
| **Decision Variables** | |
| $X_j$ | $\begin{cases} 1 & \text{if SB } j \in J \text{ is available} \\ 0 & \text{otherwise.} \end{cases}$ |
| $Y_{ijk}$ | $\begin{cases} 1 & \text{if element } i \in \mathcal{I} \text{ is checked in SB } j \in \mathcal{M}_i \text{ at safety level } k \in \{1, \ldots, \rho_{ij}\} \\ 0 & \text{otherwise.} \end{cases}$ |
| $P_{ik}$ | probability that an error in element $i$ propagates to safety level $k$ |

failure (i.e., false negative) represents the fact that SBs are not 100% reliable. The probability of failing to detect an error related to element $i$ in SB $j$ is denoted by $\beta_{ij}$. The probability of failing to detect an error related to element $i$ when that error reaches the patient is assumed to be zero ($\beta_{iJ} = 0$). For simplicity and practical purposes the probability of a false positive is assumed to be zero, i.e., a SB does not falsely indicate an error.

The model has three sets of decision variables. The binary variable $X_j$ equals 1 if SB $j$ is available and 0 otherwise. The binary decision variable $Y_{ijk}$ equals 1 if element $i$ is checked in SB $j \in \mathcal{M}_i$ at safety level $k \in \{1, \ldots, \rho_{ij}\}$ and 0 otherwise. The continuous variable $P_{ik} \geq 0$ is a auxiliary variable that represents the probability that all SBs checking element $i$ in the first $k-1$ safety levels did not catch an error in element $i$. The value of this variable is determined by the reliability of the SBs checking element $i$ in the first $k-1$ safety levels.The notation is summarized in Table 4.1. The reliable radiation therapy problem (RRTP) can be formulated as :

$$\min Z = \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{M}_i} \sum_{k\in\mathcal{K}_i:k\le\rho_{ij}} (t_{ij}Y_{ijk} + \gamma_i r_{ij}(1-\beta_{ij})P_{ik}Y_{ijk}) \tag{4.1a}$$

$$\text{s.t.} \sum_{j\in\mathcal{J}\setminus J} X_j = b \tag{4.1b}$$

$$\sum_{j\in\mathcal{M}_i:k\le\rho_{ij}} Y_{ijk} + \sum_{s<k} Y_{iJs} = 1 \qquad k\in\mathcal{K}_i, i\in\mathcal{I}, \tag{4.1c}$$

$$\sum_{k\in\mathcal{K}_i} Y_{iJk} = 1 \qquad i\in\mathcal{I} \tag{4.1d}$$

$$\sum_{k\in\mathcal{K}_i:k\le\rho_{ij}} Y_{ijk} \le X_j \qquad j\in\mathcal{M}_i, i\in\mathcal{I} \tag{4.1e}$$

$$Y_{ijk} \le \sum_{m\in\mathcal{M}_i:k-1\le\rho_{im}<\rho_{ij}} Y_{im,k-1} \qquad i\in\mathcal{I}, j\in\mathcal{M}_i, k\in\mathcal{K}_i : 1<k\le\rho_{ij} \tag{4.1f}$$

$$P_{i1} = 1 \qquad i\in\mathcal{I} \tag{4.1g}$$

$$P_{ik} = P_{i,k-1} \sum_{m\in\mathcal{M}_i:k-1\le\rho_{im}} \beta_{im}Y_{im,k-1} \qquad i\in\mathcal{I}, k\in\mathcal{K}_i \tag{4.1h}$$

$$X_j, Y_{ijk} \in \{0,1\},\ P_{ik} \ge 0 \qquad k\in\mathcal{K}_i, j\in\mathcal{M}_i, i\in\mathcal{I} : k\le\rho_{ij} \tag{4.1i}$$

The objective function (4.1a) minimizes the total cost associated with checking elements and the expected rework cost of errors including those that reach the patient. Note that checking cost $t_{ij}$ and rework cost $r_{ij}$ for SBs other than the patient (i.e., SB $J$) are in time units. However, $r_{i,J}$ represents the medical cost of an error reaching the patient. Therefore, it is assumed that $r_{i,J}$ is weighted with an appropriate parameter before adding it to the workload costs in the objective function. That weight represents the decision maker's willingness to incur the extra workload associated with the quality assurance required to try to prevent RT errors from harming the patient.

Constraint (4.1b) ensures that the number of available SBs (excluding the patient) in the optimal solution is equal to $b$. This constraint represents the trade-off between the number of SBs in the process and the workload (number of elements) associated with each SB. Constraints (4.1c) ensure that each element $i$ is either assigned to a SB at safety level $k$, or assigned to the "patient" $J$ at safety level $s \le k$. Constraints (4.1d) require that each element is checked in

SB $J$ (patient) at some safety level $k$. Constraints (4.1e) ensure that elements are only checked in available SBs. Constraints (4.1f) state that if element $i$ is assigned to SB $j$ at level $k > 1$, then it must have been checked in a previous SB $m$ at safety level $k-1$. Constraints (4.1g) and (4.1h) are recursive probability equations. At the first level $k = 1$, constraints (4.1g) state that $P_{i,1} = 1$ since element $i$ is not checked before its first safety level. At later safety levels $k > 1$, constraints (4.1h) state that $P_{ik}$ is equal to the product of the probability that an error in element $i$ is checked at safety level $k-l$ and the probability that the SB checking element $i$ at level $k-1$ fails. Finally, constraints (4.1i) enforce the binary and non-negativity requirements on variables. The nonlinear term in $P_{ik}Y_{ijk}$ in the objective (4.1a) and in constraints (4.1h) is a product of a continuous variable and a binary variable. Thus, it can represented by an auxiliary variable $W_{ijk}$:

$$W_{ijk} \leq P_{ik}$$

$$W_{ijk} \leq Y_{ijk}$$

$$W_{ijk} \geq P_{ik} + Y_{ijk} - 1$$

$$W_{ijk} \geq 0$$

As mentioned in Chapter 2, our proposed model is motivated by the *reliable facility location problem (RFLP)*. In the next section, we provide an overview of the RFLP and the corresponding literature, and we describe the relationship between our problem and RFLP.

## 4.3 Literature Review

In this section, we review relevant studies focused on the RFLP and highlight the differences and similarities between our work and existing models in literature. We took advantage of these similarities to linearize our model and also benefit from the solution approaches proposed for RFLP. One alternative approach to formulate RRTP is to limit the number of errors that reach the patient by incorporating an additional constraint in the model that ensures the expected number of errors reaching the patient does not exceed a given $\epsilon$. In this approach, the objective function only consists of the checking cost and the rework cost associated with detecting errors.

While this approach is more flexible for restricting the number of errors that reach the patient, constructing feasible solutions to generate upper bounds would be more difficult (see section 4.5.2). Since our proposed model utilizes some features of the RFLP, we were able to develop an efficient solution algorithm for the RRTP similar to the approaches proposed for the RFLP.

The reliable SB design problem can be modeled as a RFLP which considers supply chains with facilities that are susceptible to failures [1]. In such supply networks, customers have to be served by other facilities when the primary facility to which they were assigned fails [1]. To mitigate the impact of extra transportation and penalty costs in case of failures, each customer is assigned to multiple facilities that serve as backup [1]. The facilities are classified into *backup levels*, and customers are assigned to facilities in the order of the levels. A "level-$k$" facility assignment for a customer is the one for which there are $k-1$ closer facilities that are open. Each customer is served by the lowest-level facility that is available after a disruptive event. The objective is to find a cost-efficient solution to minimize the operating and transportation costs under both normal and post-disruption conditions [29]. Table 4.2 maps the components of the reliable radiation therapy problem to similar concepts in the reliable facility location problem. Safety barriers are modeled like the RFLP facilities while elements are equivalent to "customers". Elements are assigned to SBs where each SB has a failure probability. When a SB fails to catch an error, other SBs serve as backup and may detect the error with some probability. The goal is to determine SB locations and composition, taking into account the cost of checking elements and the expected rework cost.

Table 4.2: Comparing reliable radiation therapy problem to the reliable facility location problem

| Reliable Facility Location Problem | | Reliable Radiation Therapy Problem |
|---:|:---:|---|
| Customers | $\Rightarrow$ | Elements |
| Demand Rate | $\Rightarrow$ | Error Probability |
| Facilities | $\Rightarrow$ | Safety Barriers (SBs) |
| Facility Failure Probability | $\Rightarrow$ | SB Failure Probability |
| Transportation Cost | $\Rightarrow$ | Rework and Checking Cost |

The reliable facility location problem was first introduced by Snyder and Daskin [116]. They incorporated reliability into two classic problems: the uncapacitated fixed-charge location

problem (UFLP) and the P-median problem (PMP). The UFLP determines facility locations and assignment of customers to facilities to minimize total fixed and transportation cost. PMP is a facility location problem in which the number of facilities to be located is fixed. The two proposed models by Snyder and Daskin [116] assume that one or more facilities may fail and become unavailable and the goal is to find the facility locations to minimize the operating cost while minimizing the expected transportation cost after facility failures. They presented a Lagrangian relaxation algorithm to solve these models and also showed how to derive the trade-off curve between the operating cost and the expected cost of failures. The main limitations of their proposed models are the assumptions that all failable facilities are uncapacitated and have the same probability of failure. Many studies later relaxed the assumption of uniform failure probabilities [18, 29, 72, 70, 109]. Lim et al [72] studied the general facility reliability problem (FRP) to take into account random facility failures in the UFLP considering a mixture of unreliable and perfectly reliable facilities with different costs. They allowed site-specific failure probabilities and showed that the problem can be reduced to the classical UFLP for extreme failure probabilities.

Berman et al [18] proposed an integer programming model for the $P$-median problem with unreliable facilities (MPUF) and heterogeneous failure probabilities. They demonstrated that optimal location patterns strongly depend on the probability of failure and facilities become more centralized as the probability of failure increases. They also developed a number of exact and heuristic approaches to solve MPUF. Similarly, Cui et al [29] studied reliable facility location models considering site-dependent failure probabilities. They introduced a compact mixed integer program (MIP) formulation and a continuum approximation model. Shen et al [109] developed models for the uncapacitated reliable facility location problem. For the special case with uniform failure probabilities they proposed an approximation algorithm with a worst case bound of four. For the general problem with facility specific failure probabilities, they formulated the problem as a two-stage stochastic programming (SP) model as well as a non-linear integer programming (NLIP) model and proposed several heuristics to generate efficient solutions.

O'Hanley and Church [92] developed strategies for coping with the worst cases of facility loss. They proposed a bi-criteria mixed-integer programming model for locating $p$ facilities in

order to maximize a combination of (i) the initial coverage of all facilities and (ii) the minimum coverage level after the failure of the most critical $k$ facilities. Berman et al [17] studied a $p$-median problem for a network subject to the failure of the most damaging link which can be severed either by a deliberate terrorist or natural disaster.

Most studies have assumed no correlation between facility failures implying that facility disruptions occur independently. Recently a few papers [71, 7, 75] incorporated disruption correlations since closer facilities are more likely to be affected in a similar way from disruptive events.

Despite their similarities, there are three significant differences between the reliable radiation therapy problem (RRTP) and the reliable facility location problem (RFLP). First, no customer can be served by a failed facility in the reliable facility location problem. However, in the reliable radiation therapy problem, SBs are composed of multiple elements which means when a safety barrier fails to detect an error in a particular element, it may still successfully detect errors in other elements. Another significant difference is between the backup levels and the safety levels. In the reliable facility location problem, there is no restriction when assigning a customer to facilities at different backup levels. However, in our formulation the position of the SB checking an element at safety level $k$ must be earlier than the position of the SB checking the same element at safety level $k+1$. This is because SBs are performed sequentially with respect to the order of their positions in the RT process. Finally, the reliable facility location problem assumes that customers do not travel to the failed facilities, and hence the transportation cost is only incurred when customers patronize operational facilities. However, in the reliable radiation therapy design problem, the cost of checking an element is still incurred even when a SB fails to detect an error related to that element.

## 4.4 RRTP - Special Case

In this section, we present a special case of reliable radiation therapy problem (RRTP) which we will denote by RRTP-SC and demonstrate some of its structural properties that help us gain additional insights about the original problem. The RRTP-SC makes the following assumptions:

- The time required to check element $i$ is the same for all SBs ($t_{ij} = t_{im} \ \forall \ j, m \in \mathcal{M}_i$)

- The probability of failing to detect error $i$ is the same for all SBs ($\beta_{ij} = \beta_{im}\ \forall\ j, m \in \mathcal{M}_i$)

Given these assumptions, we can substitute $\beta_{ij}$ and $t_{ij}$ with $\beta_i$ and $t_i$ and obtain the following objective function:

$$\sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{M}_i} \sum_{k\in\mathcal{K}_i:k\leq\rho_{ij}} \Psi_{ijk} Y_{ijk} \tag{4.3a}$$

where

$$\Psi_{ijk} \quad = \quad \begin{cases} \left(t_i + \gamma_i r_{ij} \beta_i^{k-1}(1-\beta_i)\right) & \text{if } j \in \mathcal{M}_i \setminus \{J\} \\ \left(\gamma_i r_{iJ} \beta_i^{k-1}\right) & \text{if } j = J \end{cases}$$

Since the recursive probability equations are incorporated in the new objective function, we can relax the constraints (4.1g) and (4.1h). The following proposition shows that assuming SBs with higher order have higher rework cost for a given element $i$, it would be optimal to assign the element to the SB with a lower order first.

**Proposition 1.** *If $r_{im} < r_{in}$ for $i \in \mathcal{I}$, $m, n \in \mathcal{M}_i \setminus \{J\} : \rho_{im} < \rho_{in}$, then $Y_{imk} \geq Y_{ink}$ for $k \in \mathcal{K}_i : k \leq \rho_{im} < \rho_{in}$ in any optimal solution to RRTP-SC.*

*Proof.* Suppose that $(X, Y)$ is an optimal solution to RRTP-SC in which, $Y_{imk} = 0$, $Y_{ink} = 1$ and $r_{im} < r_{in}$. We prove that if we swap $m$ and $n$, the objective function will improve:

The assignments $Y_{imk} = 0, Y_{ink} = 1$ contribute $\Psi_{ink}$ to the objective function. If we assigned $i$ to $m$ at level $k$, the objective function will change by $\Psi_{imk} - \Psi_{ink}$ which will be:

$$\Psi_{imk} - \Psi_{ink} = \left(t_i + \gamma_i r_{im} \beta_i^{k-1}(1-\beta_i)\right) - \left(t_i + \gamma_i r_{in} \beta_i^{k-1}(1-\beta_i)\right) = (r_{im} - r_{in})\beta_i^{k-1}(1-\beta_i) < 0$$

The difference between the objective function values is negative which shows the objective function improves after swapping the assignments and this contradicts the assumption of the optimality of the original solution. Therefore $Y_{imk} \geq Y_{ink}$. □

Proposition 1 implies that $Y_{ijk} \leq \sum_{m\in\mathcal{M}_i:k-1\leq\rho_{im}<\rho_{ij}} Y_{im,k-1} \quad i \in \mathcal{I}, j \in \mathcal{M}_i, k \in \mathcal{K}_i : 1 < k \leq \rho_{ij}$, so we can relax the ordering constraint (4.1f). Since $\rho_{im} < \rho_{ij}$, the rework cost at SB$j$ is larger than the rework cost at SB $m$ ($r_{im} < r_{ij}$) and therefore $Y_{ijk} \leq Y_{imk}$. After

relaxing constraints (4.1f), (4.1g) and (4.1h) constraints we obtain the following formulation for RRTP-SC:

$$\min \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{M}_i} \sum_{k\in\mathcal{K}_i:k\leq\rho_{ij}} \Psi_{ijk} Y_{ijk} \tag{4.4a}$$

$$\text{s.t.} \sum_{j\in\mathcal{J}\setminus J} X_j = b \tag{4.4b}$$

$$\sum_{j\in\mathcal{M}_i:\rho_{ij}\geq r} Y_{ijk} + \sum_{s<k} Y_{iJs} = 1 \qquad k\in\mathcal{K}_i, i\in\mathcal{I}, \tag{4.4c}$$

$$\sum_{k\in\mathcal{K}_i} Y_{ijk} = 1 \qquad i\in\mathcal{I} \tag{4.4d}$$

$$\sum_{k\in\mathcal{K}_i:k\leq\rho_{ij}} Y_{ijk} \leq X_j \qquad j\in\mathcal{M}_i, i\in\mathcal{I} \tag{4.4e}$$

$$X_j, Y_{ijk} \in \{0,1\} \qquad k\in\mathcal{K}_i, j\in\mathcal{M}_i, i\in\mathcal{I} : k\leq\rho_{ij} \tag{4.4f}$$

The objective function (4.4a) minimizes the total expected cost that consists of checking cost and expected rework cost. Constraints (4.4b)-(4.4f) are constraints from RRTP. The constraints regarding the recursive probabilities are relaxed, because in RRTP-SC the probability that an error in element $i$ propagates to safety level $k$ is independent of the location of SBs that check element $i$ and only depends on failure probability $\beta_i$ and $k$.

**Structural Properties of RRTP-SC**

In this section, we derive structural properties of RRTP-SC based on the marginal value of the assignment of an element to a SB. We define $v_{ijk}$ as the marginal value of assigning element $i\in\mathcal{I}$ to SB $j$ at level $k\in\mathcal{K}_i : k\leq\rho_{ij}$. In other words $v_{ijk}$ is the total savings gained by adding element $i$ to SB $j$ at level $k\in\mathcal{K}_i : k\leq\rho_{ij}$. The marginal value of checking element $i$ at SB $j$ at level $k$, $v_{ijk}$ is given by:

$$v_{ijk} = \gamma_i r_{iJ}\beta_i^{k-1}(1-\beta i) - (t_i + \gamma_i r_{i,j}\beta_i^{k-1}(1-\beta i)) = \gamma_i(r_{iJ}-r_{ij})\beta_i^{k-1}(1-\beta i) - t_i$$

Adding element $i$ to SB $j$ at level $k$ reduces the total cost by:

$$\gamma_i r_{iJ} \beta_i^{k-1} - \gamma_i r_{iJ} \beta_i^k = \gamma_i r_{iJ} \beta_i^{k-1}(1-\beta_i)$$

and increases the total cost by:

$$t_i + \gamma_i r_{ij} \beta_i^{k-1}(1-\beta i)$$

Therefore the total reduction in cost will be:

$$v_{ijk} = \gamma_i r_{iJ} \beta_i^{k-1}(1-\beta i) - (t_i + \gamma_i r_{ij} \beta_i^{k-1}(1-\beta i)) = \gamma_i (r_{iJ} - r_{ij}) \beta_i^{k-1}(1-\beta i) - t_i$$

The maximum marginal value $Max(v_{ijk})$ for element $i$ at level $k \in \mathcal{K}_i$, is achieved by assigning element $i$ to the first SB that can check it i.e., SB $j$ where $j \in \mathcal{M}_i$ and $\rho_{ij} = k$. This is due to the fact that the marginal value for element $i$ at level $k \in \mathcal{K}_i$ is smaller at SBs with higher order. We demonstrate this in Proposition 2:

**Proposition 2.** *For $i \in \mathcal{I}, m, n \in \mathcal{M}_i$ and $k \in \mathcal{K}_i : k \leq \rho_{im} < \rho_{in}$ :*

$$v_{imk} \geq v_{ink}$$

*Proof.* By contradiction, if we assume $v_{imk} < v_{ink}$, then we have:

$$v_{im,k} < v_{ink} \quad \rightarrow \gamma_i (r_{iJ} - r_{im}) \beta_i^{k-1}(1-\beta i) - t_i < \gamma_i (r_{iJ} - r_{in}) \beta_i^{k-1}(1-\beta i) - t_i \rightarrow r_{im} > r_{in}$$

which contradicts the assumption that SBs with higher order have higher rework cost, therefore $v_{imk} < v_{ink}$. □

We also show that for $j \in \mathcal{M}_i$ the marginal value of checking element $i$ at SB $j$ at safety level $k \in \mathcal{K}_i : k \leq \rho_{ij}$ is greater than the marginal value of checking the element at SB $j$ at safety level $(k+1) \in \mathcal{K}_i : (k+1) \leq \rho_{ij}$.

**Proposition 3.** *For $i \in \mathcal{I}, j \in \mathcal{M}_i$ and $k \in \mathcal{K}_i : k \leq \rho_{ij}$ :*

$$v_{ijk} \geq v_{ij(k+1)}$$

*Proof.* By contradiction if we assume $\mathrm{v}_{ijk} < v_{ij(k+1)}$, then we have:

$$\begin{aligned}
& v_{ijk} < v_{ij(k+1)} \\
& \rightarrow \gamma_i(r_{iJ} - r_{ij})\beta_i^{k-1}(1-\beta i) - t_i < \gamma_i(r_{iJ} - r_{ij})\beta_i^{k}(1-\beta i) - t_i \\
& \rightarrow \gamma_i(r_{iJ} - r_{ij})\beta_i^{k-1}(1-\beta i) < \gamma_i(r_{iJ} - r_{ij})\beta_i^{k}(1-\beta i) \\
& \rightarrow (r_{iJ} - r_{ij})\beta_i^{k-1} < (r_{iJ} - r_{ij})\beta_i^{k} \\
& \rightarrow (r_{iJ} - r_{ij}) < (r_{iJ} - r_{ij})\beta_i \\
& \rightarrow r_{iJ}(1-\beta_i) < 0
\end{aligned}$$

which is a contradiction since $\beta_i < 1$ and $r_{iJ} > 0$. □

The marginal value of element $i$, i.e., $v_{ijk}$ can be used to determine the optimal assignments $(Y_{ijk})$. Using the implications of Proposition 3 and given the SB location vector $(X_j)$, we can easily find the optimal assignments for RRTP-SC. Once the optimal locations of SBs are known, the problem can be divided into $i$ problems each optimizing the assignments for a given element $i$. For each element $i$, we sort SBs in $\mathcal{M}_i$ in an increasing order of $\rho_{ij}$. Then, we start at level $k = 1$ and assign the element to the first SB $j \in \mathcal{M}_i$ if the SB is open and $v_{ijk} > 0$. We assign the element to SBs level by level until we reach a level where either adding the element to the next available SB is not cost-effective ($v_{ijk} \leq 0$) or there is no available SB to include element $i$ at level $k$. We show that if the failure probability for an element is larger than a certain value, it will not be optimal to include that element in any of the SBs.

**Proposition 4.** *For each element $i \in \mathcal{I}$ if $\beta_i \geq Max\{1 - \frac{t_i}{\gamma_i(r_{iJ} - r_{ij})}\}$ $j \in \mathcal{M}_i$, then it is not optimal to check element $i$ in any SB.*

*Proof.* We know that it is not optimal to assign element $i$ to any $SB$ $j \in \mathcal{M}_i$ if for every $j$ and every $k \in \mathcal{K}_i : k \leq \rho_{ij}$, $v_{ijk} \leq 0$. According to Proposition 3, $v_{ij1} \geq v_{ijk}$ for $k \in \mathcal{K}_i$, therefore it

is not optimal to check element $i$ in any SB, if $v_{ij1} \leq 0$ for all $j \in \mathcal{M}_i$:

$$
\begin{aligned}
& v_{ij1} \leq 0 && j \in \mathcal{M}_i \\
\rightarrow & \gamma_i(r_{iJ} - r_{ij})\beta_i^{1-1}(1-\beta_i) - t_i \leq 0 && j \in \mathcal{M}_i \\
\rightarrow & \gamma_i(r_{iJ} - r_{ij})(1-\beta_i) \leq t_i && j \in \mathcal{M}_i \\
\rightarrow & (1-\beta_i) \leq \frac{t_i}{\gamma_i(r_{iJ} - r_{ij})} && j \in \mathcal{M}_i \\
\rightarrow & \beta_i \geq 1 - \frac{t_i}{\gamma_i(r_{iJ} - r_{ij})} && j \in \mathcal{M}_i
\end{aligned}
$$

Therefore if:

$$\beta_i \geq Max\{1 - \frac{t_i}{\gamma_i(r_{iJ} - r_{i,j})}\} \qquad j \in \mathcal{M}_i$$

it is not optimal to check element $i$ in any SB. □

## 4.5 A Solution Approach for RRTP

Given the size of the problem for a standard RT facility (e.g., 11 process steps and 65 elements in UNC Cancer Center), off-the-shelve solvers may not find solutions of acceptable quality even within a few hours. This fact motivates us to derive an efficient solution method.

In the literature, different solution methods have been developed to solve the reliable facility location problem including Lagrangian relaxation [116, 29], search and cut algorithm [1], sample average approximation [109], continuum approximation [29, 71] and greedy heuristics[109]. However, these algorithms do not directly apply to the reliable RT problem because of the significant differences between RRTP and RFLP as described in Section 4.3. To solve the RRTP, we present a solution approach similar to the search and cut procedure in Aboolian et al. [1]. This approach generates initial upper and lower bounds for RRTP and then successively improves them. In Section 4.5.1, we formulate a mixed-integer program that provides a lower bound for RRTP. In Section 4.5.2, we present a neighborhood search heuristic to obtain an upper bound for RRTP. Finally, in Section 4.5.3, we present the solution approach that consecutively improves the lower and upper bounds.

### 4.5.1 A Lower Bound for RRTP

Let $\beta_{i[1]} \leq ... \leq \beta_{i[m_i-1]} \leq \beta_{i[m_i]}$ be a nondecreasing ordering of the failure probabilities in $M_i$. We define :

$$Q_{ik} = \prod_{t=1}^{k-1} \beta_{i[t]}$$

that is the probability that all SBs checking element $i$ in the first $k-1$ safety levels fail to detect an error when element $i$ is checked by the most reliable $k-1$ SBs. Therefore, $Q_{ik}$ is an optimistic estimate of $P_{ik}$ providing a lower abound for it [1]. We replace $P_{ik}$ with $Q_{ik}$ $\forall i \in \mathcal{I}$, $k \in \mathcal{K}_i$ in RRTP to obtain the following lower bounding problem (LBP):

$$\min \ Z_L = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{M}_i} \sum_{k \in \mathcal{K}_i : k \leq \rho_{ij}} (t_{ij} Y_{ijk} + \gamma_i r_{ij} (1 - \beta_{ij}) Q_{ik} Y_{ijk})$$

$$\text{s.t. } (4.1b) - (4.1f)$$

$$X_i, Y_{ijk} \in \{0, 1\} \qquad k \in \mathcal{K}_i, j \in \mathcal{M}_i, i \in \mathcal{I} : k \leq \rho_{ij}$$

Note that $Q_{ik}$ is a constant parameter and $Z_L$ provides a lower bound to $Z$.

### 4.5.2 An Upper Bound for RRTP

An upper bound can be obtained by a generating a feasible solution based on SB positions obtained from LBP problem. Given SB positions $\hat{X}$, we generate a feasible solution to RRTP by assigning elements to available SBs level by level, i.e., we assign each element $i$ to the first available SB $j \in \mathcal{M}_i$ at safety level 1, then we assign the element to the second available SB $j \in \mathcal{M}_i$ at safety level 2 and so on. We continue this procedure until element $i$ is assigned to all available SBs in $\mathcal{M}_i$. The corresponding objective value $Z_{RRTP}(\hat{X}, \hat{Y})$ provides an upper bound for RRTP.

To improve the upper bound we use a neighborhood search approach. Let subset $S_{\hat{X}} \subseteq \mathcal{J}$ denote the set of SBs that are available given the SB positions $\hat{X}$ obtained by solving the LBP.

We define $N_p(S_{\hat{X}})$, the distance-$p$ neighborhood of $S_{\hat{X}} \subseteq \mathcal{J}$ as:

$$N_p(S) = \{S' \subseteq \mathcal{J} : \left|S - S'\right| + \left|S' - S\right| \leq p\}$$

that is $S'$ is in the distance-$p$ neighborhood of $S$ if the number of nonoverlapping elements in the two sets does not exceed $p$. After defining the neighborhood, we use $S_{\hat{X}}$ generated by solving the LBP, as a starting solution. We evaluate the value of the objective function for all feasible solutions in the distance-$p$ neighborhood of $S_{\hat{X}}$. We choose the solution with the greatest improvement as the next starting solution and repeat the neighborhood search. The search continues until no better solution can be found in the neighborhood. Let $S_{\bar{X}}$ be the solution subset after performing the neighborhood search approach, and $Y(\bar{X})$ be the assignment vector where elements are assigned to available SBs in $S_{\bar{X}}$ as previously described. The new upper bound is the value of the objective function $Z_{RRTP}(\bar{X}, Y(\bar{X}))$.

### 4.5.3 Search and Cut (SnC) Algorithm

We solve RRTP by improving the lower and upper bounds iteratively. In the first iteration $\ell = 1$, the initial lower and upper bounds are found as described in Sections 4.5.1 and 4.5.2. Then, in the subsequent iterations, we improve the lower bound by solving LBP with additional cuts to exclude previously evaluated solutions from the feasible region.

Let $A_{LBP(t)}, t = 1, 2, .., \ell - 1$ denote the set of all starting solutions used in the neighborhood search approach. The lower bounding problem with additional cuts at iteration $\ell$, i.e., LBP($\ell$) is formulated as follows:

$$\min \ Z_{LBP(\ell)} = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{M}_i} \sum_{k \in \mathcal{K}_i : k \leq \rho_{ij}} (t_{ij} Y_{ijk} + \gamma_i r_{ij} Q_{ijk} Y_{ijk})$$

s.t. (4.1b) – (4.1f)

$$\sum_{j \in S} X_j - \sum_{j \in M \setminus S} X_j \leq |S| - p - 1 \qquad \forall S \in A_{LBP(t)},\ t = 1, 2, .., \ell - 1 \ \text{(1g)}$$

$$X_i, Y_{ijk} \in \{0, 1\} \qquad k \in \mathcal{K}_i, j \in \mathcal{M}_i, i \in \mathcal{I} : k \leq \rho_{ij}$$

The constraints (1g) ensures that for $\forall S \in A_{LBP(t)},\ t = 1, 2, .., \ell - 1$ all solutions in the $p$-

neighborhood of $S$ are infeasible. Note that to solve LBP($\ell$), we need to solve LBP($t$) for $t = 1, 2, .., \ell - 1$. Note that LBP(1) does not include constraint (1g) and is the original LBP presented in Section 4.5.1. Also note that constraints (1g) exclude all the available SB sets that have already been examined without excluding the unexamined ones. The procedure continues until the gap between the lower bound and upper bound is within a precision level. Assuming a precision level, $\epsilon \geq 0$, the proposed search and cut algorithm is outlined as follows:

Algorithm 1 (The search and cut algorithm)

*Step* 0. Set $\ell = 1$, $S_{X^*} = \{\}$ , *Lower Bound*$= 0$ and *Upper Bound*$= \infty$

*Step* 1. Solve $LBP(\ell)$. If there is a feasible solution, find: (i) $X^*_{LBP(\ell)}$ the optimal SB positions in $LBP(\ell)$; (ii) $S_{X^*_{LBP(\ell)}}$ the set of available SBs (i.e., open SBs) under $X^*_{LBP(\ell)}$; (iii) $Y(X^*_{LBP(\ell)})$ the element assignment vector, where elements are assigned to available SBs in $S_{X^*_{LBP(\ell)}}$ level by level; (iv) $Z^*_{LBP(\ell)}$ the value of the objective function for $LBP(\ell)$; and then set *Lower Bound*$(\ell) = Z^*_{LBP(\ell)}$ and proceed to Step 2. Otherwise, go to Step 5.

*Step* 2. If *Lower Bound*$(\ell) >$ *Lower Bound*, let *Lower Bound* $=$ *Lower Bound*$(\ell)$.
If (*Upper Bound* $-$ *Lower Bound*)$/$*Lower Bound* $< \epsilon$, go to Step 5. Otherwise go to Step 3.

*Step* 3. Perform the neighborhood search approach starting from the set of available SBs $S_{x^*_{LBP(\ell)}}$ and find (i) $A_{LBP(\ell)}$ the set of all the starting solutions in the neighborhood search approach; (ii) $X(\ell), Y(X(\ell))$ the SB positions and element assignments of the best found solution; (iii) $S_{X(l)}$ the set of available SBs based on $X(\ell)$ and their corresponding objective function value $Z_{RRTP}(X(\ell), Y(X(\ell)))$. Set *Upper Bound*$(\ell) = Z_{RRTP}(X(\ell), Y(X(\ell)))$

*Step* 4. If *Upper Bound*$(\ell) <$ *Upper Bound* then *Upper Bound* $=$ *Upper Bound*$(\ell)$, $X^* = X(\ell), Y(X^*) = Y(X(l))$ and $S_{X^*} = S_{X(\ell)}$. Set $\ell = \ell + 1$ and go to Step 1.

*Step* 5. Stop. The solution to the search-and-cut algorithm is as follows: (i) the set of available SBs is $S_{X^*}$, (ii) SB positions vector is $X^*$, (iii) element assignment vector is $Y(X^*)$, and

(iv) the objective function value $Z^* = UpperBound$.

Note that the algorithm terminates in either Step 1 when all the potential SB positions are examined or Step 5 when the gap between upper and lower bound falls below the predefined precision level $\epsilon$.

**Branch and Bound Algorithm**

If the search and cut algorithm (SnC) fails to improve the gap between the lower and upper bounds over a specified number of iterations, we use branch and bound to expedite the convergence. We branch on the SB location variables $X_j$. At each branch and bound node, we select an unfixed SB that is closest to the patient $J$, and set the corresponding $X_j$ to 0 and then 1. Node selection is performed in a best-bound manner. At each branch and bound node, we solve the LBP and perform the neighborhood to find the lower and upper bounds for that node. The tree is fathomed at a given node if the gap between the lower bound at the node and the objective function value of the incumbent solution falls below $\epsilon$, or if the model is infeasible.

## 4.6 Computational Results

In this section, we report and analyze the results obtained by our numerical experiments. In Section 4.6.1, we describe how we estimate the parameter values for RRTP. In Section 4.6.2 we evaluate the performance of our proposed solution approach and compare it against Gurobi. In Section 4.6.3 we discuss the behaviors of cost and solution policies under different scenarios.

### 4.6.1 Instance Generation

We estimated the parameters for RRTP based on the RT process at the Department of Radiation Oncology of University of North Carolina (UNC) Lineberger Comprehensive Cancer Center. As mentioned previously, RT process consists of seven stages four of which are in the "pre-treatment" phase. These four stages and the corresponding process steps that generate elements are listed in Table 4.3. There are 65 elements produced at process steps. The percentage of elements generated at each process step is shown in the third column of Table 4.3. To estimate

the parameter values for an instance of RRTP with $n$ elements, we first assign each element to a process step according to the probabilities shown in the third column. These probabilities are estimated from UNC data. In the following we describe how we estimate different model parameters.

**Error Probability**

We determine the error probability for each element according to the step the element is assigned. The last column in Table 4.3 shows the error probabilities for elements generated at each step. The error probabilities are estimated based on the errors reported over a two-year period: May 2012 - May 2014. Since the error sample sizes were small at element level, we calculated aggregate error probability for each step. We assumed the error probability was uniformly distributed across the elements within that step, so we divided the error probability of each step by the number of elements that are generated at that step, to obtain the error probability at the element level.

Table 4.3: Error probabilities by stage and step and the percentage of elements generated per step

| Stage | Steps that initiate elements | Percent of elements generated at each step | Error probability for each element |
|---|---|---|---|
| Treatment Decision and Patient Assessment | A | 22% | 0.03% |
| | B | 17% | 0.01% |
| Simulation and Imaging for RT Planning | C | 6% | 0.19% |
| | D | 8% | 0.22% |
| Treatment Planning | E | 26% | 0.06% |
| | G | 9% | 0.26% |
| | H | 3% | 0.08% |
| | J | 8% | 0.25% |
| Pre-treatment Review and Verification | L | 2% | 0.39% |

**SB Failure probability and Checking Cost**

For each element, we estimated the SB failure probabilities and checking costs based on the workload of the element i.e. the effort required to check the element. Each generated element, will correspond to the following types: (i) low workload, (ii) medium workload, or (iii) high workload. Table 4.4 shows the percentage of low, medium or high workload elements at each process step. We assume that the checking cost for a given element is the same at all SBs before patient, i.e., $t_{im} = t_{in}\ \forall\ m, n \in M_i \setminus J$, but the failure probability is greater at later SBs. This is in line with the fact that it is relatively easier and more likely to catch an error earlier in the process. Therefore, for each element, we assign a base $\beta_{ij}$ that represents the failure probability at the earliest SB $j$ in $M_i$ and we assume the failure probability at the second earliest SB is 5% higher than the base $\beta_{ij}$, and the third $\beta_{ij}$ is 5% higher than the second one, etc.

The checking cost and the base failure probability associated with each type of element are shown in Table 4.5. We assumed that the checking cost and the base failure probability of a medium workload element are 10% and 15% higher, respectively, than those for a low workload element and the checking cost and those corresponding to a high workload element are 10% and 15% higher, respectively, than those for a medium workload element.

Table 4.4: Probability distribution for element workload

| Stage | Steps that initiate elements | Probability of low workload elements | Probability of medium workload elements | Probability of high workload elements |
|---|---|---|---|---|
| Treatment Decision and Patient Assessment | A | 0.36 | 0.43 | 0.21 |
| | B | 0.09 | 0 | 0.91 |
| Simulation and Imaging for RT Planning | C | 0 | 0.25 | 0.75 |
| | D | 0 | 0 | 1 |
| Treatment Planning | E | 0 | 0.12 | 0.88 |
| | G | 0 | 0.17 | 0.83 |
| | H | 0 | 0 | 1 |
| | J | 0 | 0.2 | 0.8 |
| Pre-treatment Review and Verification | L | 0 | 0 | 1 |

Table 4.5: Checking cost and failure probability for each element workload

| Type of element workload | Checking cost (min) | Failure probability |
|---|---|---|
| Low Workload | 0.5 | 0.05 |
| Medium Workload | 0.55 | 0.058 |
| High Workload | 0.605 | 0.066 |

**Rework Cost**

To determine the rework costs, we consider the duration time of the process steps which are shown in Table 4.6. We assume each time an error is caught by a SB, all the steps between the step that generates the element and the SB must be repeated and therefore the amount of rework is equal to the sum of duration times for each of these steps. The rework cost for the patient represents the damage that an error can potentially cause when it reaches a patient. We assume the rework cost (i.e. penalty) for an error reaching the patient is fixed at 3000 hours regardless of the type of element. The penalty cost should be calibrated appropriately to reflect the decision makers' preferences. We can also obtain an efficient frontier by varying this cost. In Sections 4.6.2 and 4.6.3, we present computational results to demonstrate the effect of changing the penalty cost on algorithm performance and solution.

Table 4.6: Duration of each step by stage

| Stage | Steps | Duration Time(hour) |
|---|---|---|
| Treatment Decision and Patient Assessment | A | 2 |
| | B | 2 |
| Simulation and Imaging for RT Planning | C | 0.25 |
| | D | 1.5 |
| Treatment Planning | E | 1 |
| | F | 2 |
| | G | 2.5 |
| | H | 0.25 |
| | J | 0.5 |
| | K | 0.75 |
| Pre-treatment Review and Verification | L | 2 |

### 4.6.2 Performance Evaluation

We conducted a series of computational experiments to test the performance of the SnC algorithm. The algorithm is tested on several instances with 25 and 50 elements with the number of available SBs ranging from 5 to 20 and 5 to 40, respectively. We estimated the parameter values for each instance using the approach described in Section 4.6.1. The size of the RRTP corresponding to different values of $n$ (i.e., number of elements) and $b$ (i.e., number of available SBs) is shown in Table 4.7.

Table 4.7: Size of the RRTP as a function of elements and available SBs

| n | b | total # of decision variables | # of binary variables | # of continuous variables | # of constraints |
|---|---|---|---|---|---|
| 25 | 5 | 3,636 | 1,786 | 1,850 | 9,164 |
| 25 | 10 | 5,446 | 2,646 | 2,800 | 13,716 |
| 25 | 15 | 6,431 | 3,106 | 3,325 | 16,211 |
| 25 | 20 | 6,841 | 3,291 | 3,550 | 17,256 |
| 50 | 5 | 14,761 | 7,311 | 7,450 | 37,021 |
| 50 | 10 | 24,446 | 12,400 | 12,046 | 61,341 |
| 50 | 15 | 32,056 | 15,756 | 16,300 | 80,461 |
| 50 | 20 | 37,841 | 18,566 | 19,275 | 95,006 |
| 50 | 25 | 42,051 | 20,601 | 21,450 | 105,601 |
| 50 | 30 | 44,936 | 21,986 | 22,950 | 112,871 |
| 50 | 40 | 47,731 | 23,306 | 24,425 | 119,936 |

We ran SnC with $\epsilon$=0.005 up to 3,600 seconds. We coded the algorithm in Python. The parameter values for the SnC algorithm are summarized in Table 4.8.

Table 4.8: Parameters for SnC algorithm

| Parameter | Value |
|---|---|
| Optimality tolerance ($\epsilon$) | 0.005 |
| Neighborhood Size | 4 |
| Gap improvement tolerance $\delta$ | 0.0001 |
| Maximum number of consecutive iterations allowed at root node with gap improvement of less than $\delta$ | 3 |
| Number of iterations at child nodes | 1 |

To benchmark our results, we used Gurobi (version 7.0.1). Table 4.9 is a summary of the computational results. Columns "n" and "b" denote the number of elements and available SBs in each instance. The column "Gurobi gap" shows the optimality gap obtained by Gurobi and the "Gurobi time" column shows the CPU time limit or the time for the Gurobi gap to reach $\epsilon$. The "SnC gap" column denotes the SnC optimality gap which is calculated by $(UB-LB)/UB$ where LB and UB denote the best lower and upper bounds found by SnC within the specified time limit. The "SnC time" shows the maximum CPU time or the time required for the SnC gap to fall below $\epsilon$. Computational times are reported in seconds.

Table 4.9: SnC algorithm performance - Base Case

| n | b | Gurobi gap | Gurobi time (sec) | Snc gap | SnC time (sec) |
|---|---|---|---|---|---|
| 25 | 5 | 9% | 3600 | $\leq 0.5\%$ | 13 |
| 25 | 10 | 13% | 3600 | $\leq 0.5\%$ | 345 |
| 25 | 15 | 13% | 3600 | $\leq 0.5\%$ | 121 |
| 25 | 20 | 10% | 3600 | $\leq 0.5\%$ | 13 |
| 50 | 5 | 47% | 3600 | 5.78% | 3600 |
| 50 | 10 | 30% | 3600 | 8.62% | 3600 |
| 50 | 15 | 27% | 3600 | 4.97% | 3600 |
| 50 | 20 | 26% | 3600 | 3.23% | 3600 |
| 50 | 25 | 24% | 3600 | 1.60% | 3600 |
| 50 | 30 | 24% | 3600 | 0.97% | 3600 |
| 50 | 40 | 23% | 3600 | $\leq 0.5\%$ | 347 |

It can be seen that in all the instances, our algorithm outperformed Gurobi and resulted in significantly smaller gaps. To study the impact of key parameters on algorithm performance, we tested the algorithm under four different scenarios. In each scenario, we increased the value of one of the following parameters by 25%: (i) penalty for an error reaching the patient, (ii) failure probabilities, (iii) error probabilities and (iv) checking costs. Our analysis revealed that overall, the failure probabilities have the greatest influence on the performance of Gurobi and SnC. Tables 4.10 and 4.11 summarize the results for the first two scenarios : (i) increasing the penalty for an error reaching the patient by 25% and (ii) increasing the failure probabilities by 25%. The results for the error probabilities and checking costs are provided in Appendix A.1.

Table 4.10: SnC algorithm performance - increasing the penalty for an error reaching the patient by 25%

| n | b | Gurobi gap | Gurobi time (sec) | Snc gap | SnC time (sec) |
|---|---|---|---|---|---|
| 25 | 5 | 9% | 3600 | $\leq 0.5\%$ | 22 |
| 25 | 10 | 12% | 3600 | $\leq 0.5\%$ | 635 |
| 25 | 15 | 12% | 3600 | $\leq 0.5\%$ | 138 |
| 25 | 20 | 9% | 3600 | $\leq 0.5\%$ | 20 |
| 50 | 5 | 49% | 3600 | 9.67% | 3600 |
| 50 | 10 | 30% | 3600 | 8.74% | 3600 |
| 50 | 15 | 26% | 3600 | 5.09% | 3600 |
| 50 | 20 | 26% | 3600 | 3.41% | 3600 |
| 50 | 25 | 24% | 3600 | 1.60% | 3600 |
| 50 | 30 | 24% | 3600 | 0.95% | 3600 |
| 50 | 40 | 22% | 3600 | $\leq 0.5\%$ | 835 |

As shown in Table 4.10, increasing the penalty for an error reaching a patient had little impact on the Gurobi and SnC gaps. The results in Table 4.11 indicate that the problem is much harder when we increase failure probabilities. Although the SnC gaps are still significantly lower than the Gurobi gaps, in two instances we observe SnC gaps greater than 10%. This is due to the fact that the quality of the lower bounds generated by the LBP depends on the $\beta_{ij}$ values and increasing $\beta_{ij}$ results in looser bounds. In the LBP, we substitute $P_{ik}$ with $Q_{ik}$ which is an optimistic version of $P_{ik}$ based on the ordering of the $\beta_{ij}$s . As $\beta_{ij}$ increases, the difference between $Q_{ik}$ and the true value of $P_{ik}$ is likely to become greater, resulting in poor gaps.

Table 4.11: SnC algorithm performance - increasing failure probabilities by 25%

| n | b | Gurobi gap | Gurobi time (sec) | Snc gap | SnC time (sec) |
|---|---|---|---|---|---|
| 25 | 5 | 9% | 3600 | $\leq 0.5\%$ | 28 |
| 25 | 10 | 13% | 3600 | $\leq 0.5\%$ | 2019 |
| 25 | 15 | 12% | 3600 | $\leq 0.5\%$ | 677 |
| 25 | 20 | 9% | 3600 | $\leq 0.5\%$ | 22 |
| 50 | 5 | 62% | 3600 | 24.7% | 3600 |
| 50 | 10 | 32% | 3600 | 11.4% | 3600 |
| 50 | 15 | 29% | 3600 | 6.73% | 3600 |
| 50 | 20 | 26% | 3600 | 4.61% | 3600 |
| 50 | 25 | 26% | 3600 | 2.40% | 3600 |
| 50 | 30 | 26% | 3600 | 1.22% | 3600 |
| 50 | 40 | 22% | 3600 | $\leq 0.5\%$ | 325 |

According to the results shown in Table 4.11, the maximum SnC gap (24.7%) occurs for the problem instance with 50 elements and 5 safety barriers. To investigate the effect of the CPU time limit on the quality of gap for this problem, we increased the time limit from 1 hour to 6 hours and observed the improvement in the SnC and Gurobi gaps at 1 hour intervals. Figure 4.4 illustrates how the SnC and Gurobi gaps change over time. It can be seen that the Gurobi gap changes very little as we increase the time limit while the improvement in SnC gap is significant and the gap falls below $\epsilon$ (0.5%) after 6 hours.

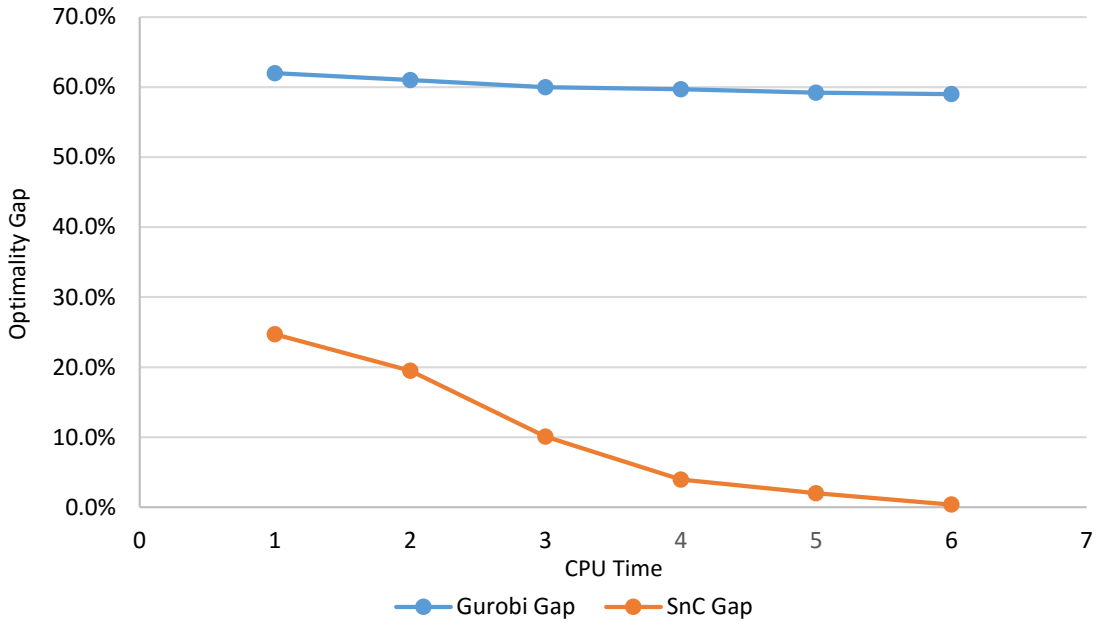


Figure 4.2: CPU Time - Solution Quality

In addition, we compared the performance of our algorithm against the total enumeration technique. In this technique, the optimization problem is solved by trying all possible combinations of discrete variables and the best result among the feasible choices is kept. Table 4.12 shows the results of the comparison for instances with 20 and 25 elements and $b = 5, 6, 7$ safety barriers. As shown in the table, the number of combinations grows exponentially as the size of the problem increases and therefore the total enumeration technique becomes inefficient. The SnC algorithm finds the optimal solution in less than a minute for all the instances in Table

4.12.

Table 4.12: Comparison between SnC Algorithm and total enumeration technique

| n | b | Total enumeration time (sec) | SnC time (sec) |
|---|---|---|---|
| 20 | 5 | 1,537 | 3 |
| 20 | 6 | 147,961 | 5 |
| 20 | 7 | $\geq$ 7 days | 17 |
| 25 | 5 | 5,389 | 13 |
| 25 | 6 | 646,800 | 29 |
| 25 | 7 | $\geq$ 7 days | 53 |

### 4.6.3 Policy Insights

In this section, we explore the optimal policies under the following scenarios: i) base case scenario, ii) increasing the penalty for an error reaching a patient by 25% and iii) increasing failure probabilities by 25% and discuss how they provide insights on the reliable design of SBs. For each scenario, we consider instances with 25 elements and 5, 10, 15 and 20 safety barriers. We first examine the total cost (i.e., SnC objective function value) breakdown for each instance. Table 4.13 shows the total cost, total rework cost, checking cost and number of checks under these scenarios. All costs are reported in minutes. We observe that rework cost accounts for the majority of the total cost. It also appears that increasing the penalty for an error reaching the patient or failure probabilities results in a greater total cost which is not surprising.

Further, it is interesting to examine the checking time for a single SB or element. Table 4.14 shows the average, minimum and maximum checking time per SB and the checking time per element for different scenarios. Note that the average checking time per SB decreases as the number of SBs increases. In addition, we observe that in all three scenarios increasing $b$ from 5 to 10 has resulted in a larger decrease in the average checking time per SB than increasing $b$ from 10 to 15 or from 15 to 20, meaning that the marginal value of opening more SBs diminishes as $b$ increases. On the other hand, the average checking time per element is not significantly affected by changes in $b$.

Finally, Table 4.15 shows the average, minimum and maximum number of checks per SB and number of checks per element. We observe a decrease in the number of checks per SB as

Table 4.13: Objective function value - cost elements

Base case

| n | b | total cost (min) | rework cost (min) | checking cost (min) | total # of checks |
|---|---|---|---|---|---|
| 25 | 5 | 190 | 135 | 55 | 92 |
| 25 | 10 | 148 | 94 | 54 | 91 |
| 25 | 15 | 138 | 84 | 54 | 91 |
| 25 | 20 | 135 | 82 | 53 | 90 |

Increasing the penalty for an error reaching the patient by 25%

| n | b | total cost (min) | rework cost (min) | checking cost (min) | total # of checks |
|---|---|---|---|---|---|
| 25 | 5 | 206 | 151 | 55 | 93 |
| 25 | 10 | 163 | 109 | 55 | 92 |
| 25 | 15 | 153 | 98 | 55 | 93 |
| 25 | 20 | 150 | 95 | 55 | 92 |

Increasing failure probabilities by 25%

| n | b | total cost (min) | rework cost (min) | checking cost (min) | total # of checks |
|---|---|---|---|---|---|
| 25 | 5 | 220 | 162 | 58 | 98 |
| 25 | 10 | 170 | 115 | 56 | 94 |
| 25 | 15 | 159 | 104 | 55 | 93 |
| 25 | 20 | 155 | 100 | 55 | 93 |

we increase $b$. It is interesting to note that all elements were checked at least once and some elements were checked up to 5 times. The effect of increasing $b$ on the number of checks per SB is also shown in Figure 4.3. In this figure, the rows correspond to the elements and the columns correspond to SBs. The shaded boxes represent the elements assigned to each SB. The last SB is the patient which includes all the elements highlighted in red. We can see that when we open more SBs, fewer elements will be assigned to each SB, i.e., the checking effort per SB decreases.

In addition, we investigated how the optimal policies are affected by changes in the error probabilities and checking cost. The detailed results for these two parameters are provided in Appendix A.2. For the scenario with $n = 25$ elements and $b = 10$ SBs, the results of the sensitivity analysis on different model parameters are summarized in Figures 4.4, 4.5 and 4.6. In these figures, we can observe how the model parameters influence the total, rework and checking costs. Figure 4.4 shows that the failure and error probabilities have the most influence on the total cost and rework cost while the checking cost is most impacted by variations in the checking cost parameters ($t_{ij}$).

The effect of changing model parameters on the solution policy is shown in Figure 4.7.

Table 4.14: Optimal policy - checking time

Base case

| n | b | checking time per SB (min) | | | checking time per element (min) | | |
|---|---|---|---|---|---|---|---|
| | | average | minimum | maximum | average | minimum | maximum |
| 25 | 5 | 10.91 | 7.04 | 14.20 | 2.18 | 0.60 | 3.03 |
| 25 | 10 | 5.40 | 2.97 | 9.02 | 2.16 | 0.60 | 2.42 |
| 25 | 15 | 3.60 | 1.11 | 7.26 | 2.16 | 0.60 | 2.42 |
| 25 | 20 | 2.67 | 0.50 | 4.24 | 2.14 | 0.60 | 2.42 |

Increasing the penalty for an error reaching the patient by 25%

| n | b | checking time per SB (min) | | | checking time per element (min) | | |
|---|---|---|---|---|---|---|---|
| | | average | minimum | maximum | average | minimum | maximum |
| 25 | 5 | 11.03 | 7.65 | 14.20 | 2.21 | 0.60 | 3.03 |
| 25 | 10 | 5.46 | 3.47 | 9.02 | 2.18 | 0.60 | 2.42 |
| 25 | 15 | 3.68 | 1.11 | 7.86 | 2.21 | 0.60 | 2.42 |
| 25 | 20 | 2.73 | 0.50 | 4.24 | 2.18 | 0.60 | 2.42 |

Increasing failure probabilities by 25%

| n | b | checking time per SB (min) | | | checking time per element (min) | | |
|---|---|---|---|---|---|---|---|
| | | average | minimum | maximum | average | minimum | maximum |
| 25 | 5 | 11.63 | 8.97 | 14.20 | 2.33 | 0.60 | 3.03 |
| 25 | 10 | 5.57 | 2.92 | 9.52 | 2.23 | 0.60 | 2.42 |
| 25 | 15 | 3.68 | 1.11 | 7.86 | 2.21 | 0.60 | 2.42 |
| 25 | 20 | 2.76 | 0.50 | 4.24 | 2.21 | 0.60 | 2.42 |

Overall, we can see that the policy is fairly robust to changes in the model parameters. The policy changes most in the cases where $b = 10$ and we increase the checking costs or failure probabilities by 25%.

Table 4.15: Optimal policy - number of checks

Base Case

| n | b | number of checks per SB | | | number of checks per element | | |
|---|---|---|---|---|---|---|---|
| | | average | minimum | maximum | average | minimum | maximum |
| 25 | 5 | 18.4 | 12 | 24 | 3.7 | 1 | 5 |
| 25 | 10 | 9.1 | 5 | 15 | 3.6 | 1 | 4 |
| 25 | 15 | 6.1 | 2 | 12 | 3.6 | 1 | 4 |
| 25 | 20 | 4.5 | 1 | 7 | 3.6 | 1 | 4 |

Increasing the penalty for an error reaching the patient by 25%

| n | b | number of checks per SB | | | number of checks per element | | |
|---|---|---|---|---|---|---|---|
| | | average | minimum | maximum | average | minimum | maximum |
| 25 | 5 | 18.6 | 13 | 24 | 3.7 | 1 | 5 |
| 25 | 10 | 9.2 | 6 | 15 | 3.7 | 1 | 4 |
| 25 | 15 | 6.2 | 2 | 13 | 3.7 | 1 | 4 |
| 25 | 20 | 4.6 | 1 | 7 | 3.7 | 1 | 4 |

Increasing failure probabilitie by 25%

| n | b | number of checks per SB | | | number of checks per element | | |
|---|---|---|---|---|---|---|---|
| | | average | minimum | maximum | average | minimum | maximum |
| 25 | 5 | 19.6 | 15 | 24 | 3.9 | 1 | 5 |
| 25 | 10 | 9.4 | 5 | 16 | 3.8 | 1 | 4 |
| 25 | 15 | 6.2 | 2 | 13 | 3.7 | 1 | 4 |
| 25 | 20 | 4.7 | 1 | 7 | 3.7 | 1 | 4 |

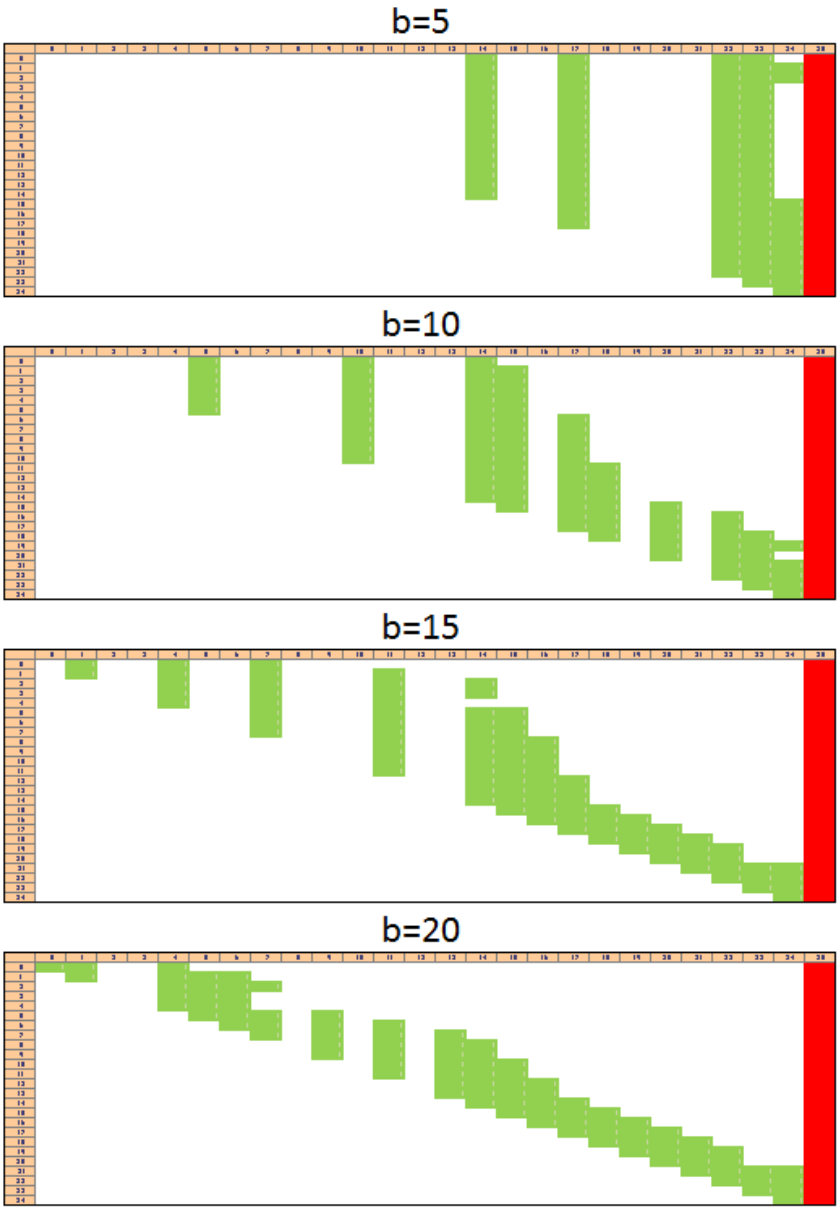


Figure 4.3: The effect of changing $b$ (number of available SBs) on optimal policy

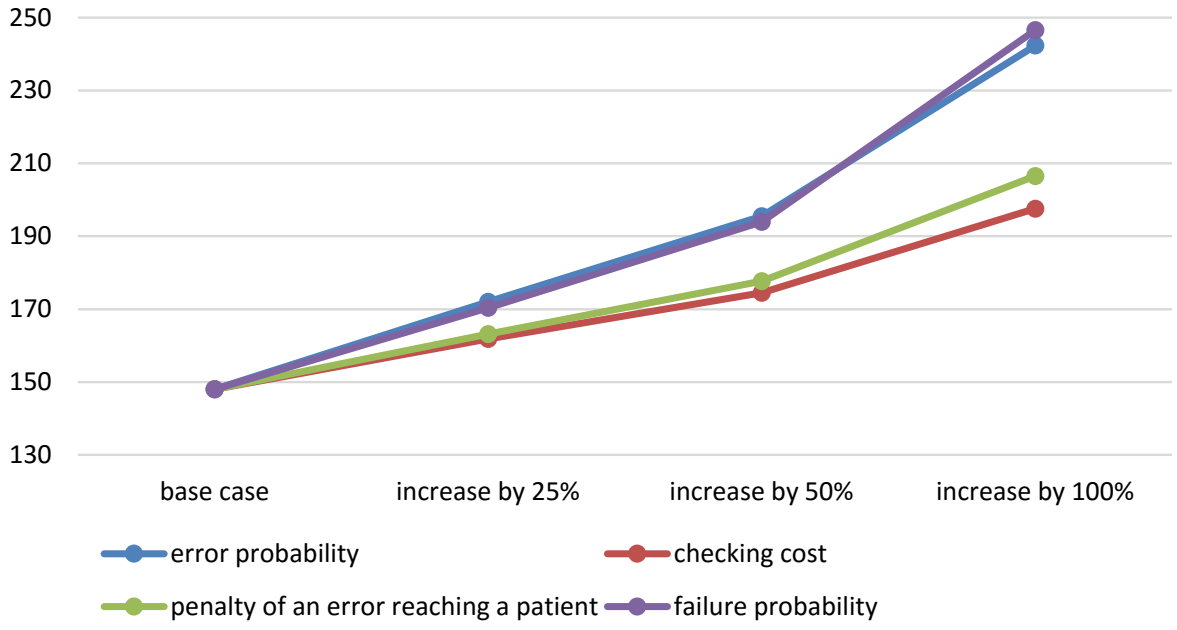


Figure 4.4: Sensitivity Analysis - Total Cost

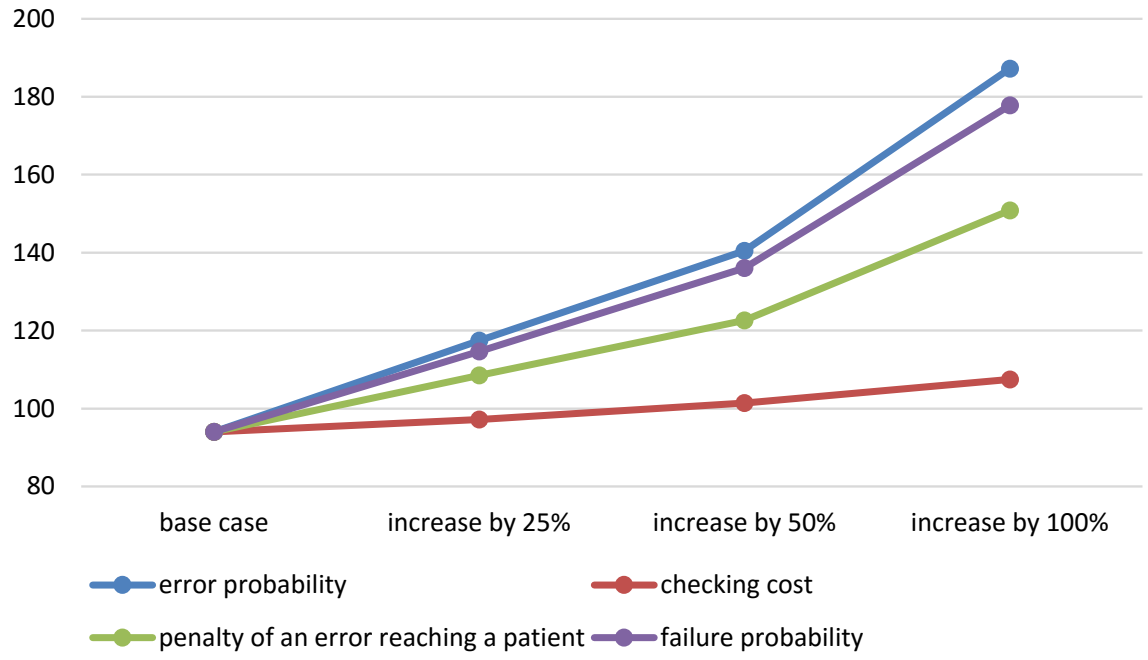


Figure 4.5: Sensitivity Analysis - Rework Cost

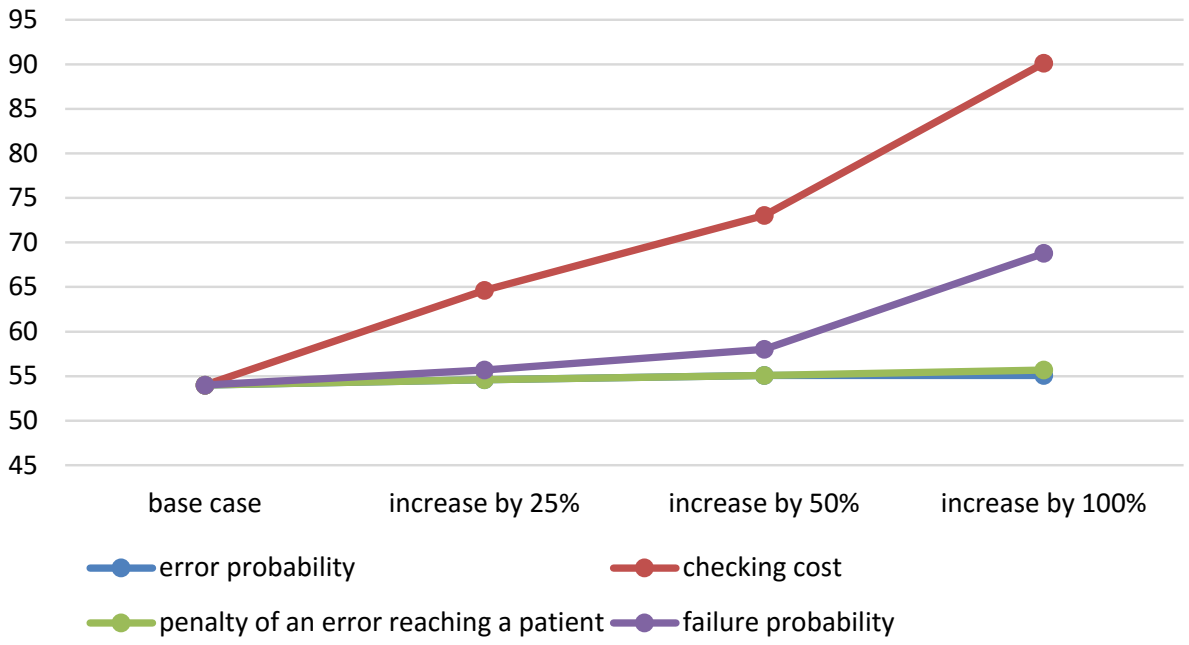


Figure 4.6: Sensitivity Analysis - Checking Cost

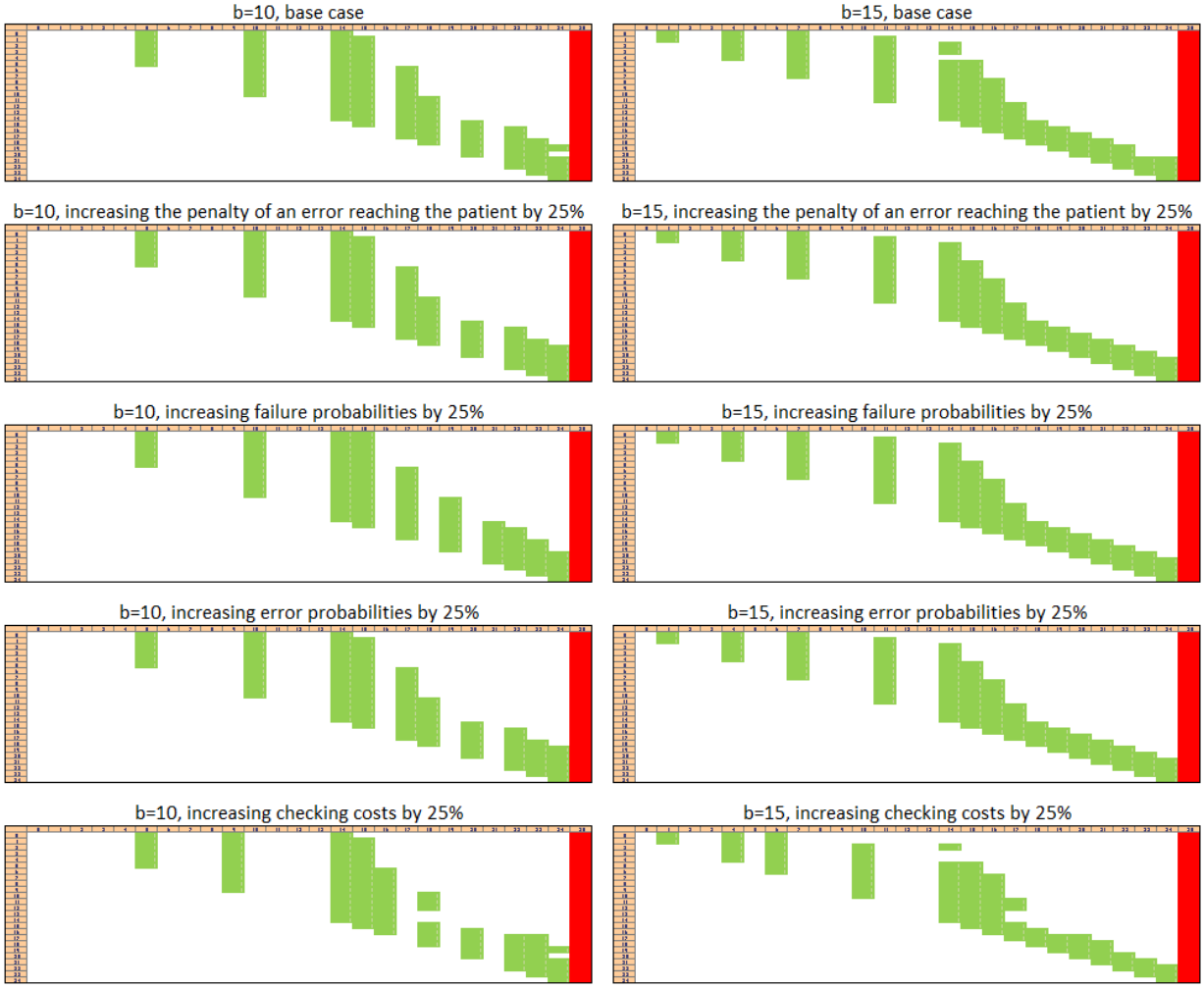


Figure 4.7: The effect of varying different parameters on optimal policy

Finally, to demonstrate how our model can improve error detection and patient safety, we compare the SB design obtained by our MIP model to the SB design used at UNC. Table 4.16 provides a summary of comparison between our solution and the SB design at UNC. The results indicate that our solution performs significantly better. We can observe that the checking time per SB and per element are higher in our solution. However, we can see that this additional checking time, results in much lower rework cost.

Table 4.16: Comparison between UNC SB design and our solution

| UNC Care Path | | | Our Solution | | |
|---|---|---|---|---|---|
| total cost | total rework cost | total checking cost | total cost | total rework cost | total checking cost |
| 8067.45 | 7980.37 | 87.08 | 304.48 | 147.80 | 156.68 |
| rework cost of errors that reach the patient | | | rework cost of errors that reach the patient | | |
| 3600 | | | 72 | | |
| expected number of errors that reach the patient | | | expected number of errors that reach the patient | | |
| 0.02 | | | 0.0004 | | |
| checking time per SB (min) | | | checking time per SB (min) | | |
| average | minimum | maximum | average | minimum | maximum |
| 10.94 | 4.2 | 23.90 | 15.67 | 5.34 | 29.68 |
| checking time per element (min) | | | checking time per element (min) | | |
| average | minimum | maximum | average | minimum | maximum |
| 1.34 | 0.6 | 3.63 | 2.41 | 0.60 | 3.30 |
| number of checks per SB | | | number of checks per SB | | |
| average | minimum | maximum | average | minimum | maximum |
| 19.23 | 8 | 41 | 26.9 | 9 | 51 |
| number of checks per element | | | number of checks per element | | |
| average | minimum | maximum | average | minimum | maximum |
| 2.34 | 1 | 7 | 4.14 | 1 | 6 |

In addition, we used the simulation model described in Chapter 3 to evaluate the solution obtained by the MIP model. We plugged our solution into the simulation model and compared the results to the UNC SB design. Using our solution, the expected number of errors that reach the patient is equal to 0.002 (half width = 0.0003) while under UNC SB design the expected number of errors that reach the patient is equal to 0.04 (half width = 0.001). These numbers

indicate that our solution greatly improves patient safety. It is important to note that the simulation results are different from those obtained from the MIP model. The main reason for this difference is related to the assumptions about rework. In the MIP model, we assume rework is perfect meaning no error occurs during the rework. In contrast, in the simulation model we assume errors may occur during the steps that are repeated when an error is caught. The second assumption is more realistic since in the RT process new errors may be generated during rework. However, for simplicity and modeling purposes we relaxed this assumption in the MIP model.

## 4.7 Conclusion and Future Work

In this chapter we presented a mixed integer programming model to determine the reliable design of SBs in RT. Our framework captures the trade-offs between SB implementation costs and their impact on patient safety. From the modeling perspective, our model is motivated by the reliable facility location problem; however there are several underlying differences that distinguish our problem from the RFLP. We developed a solution algorithm based on consecutive improvements of the lower and upper bounds to find efficient solutions. The proposed algorithm outperforms Gurobi reaching significantly smaller gaps on several random data sets. Furthermore, we examined the performance of the algorithm and the solution policies under different scenarios to assess the effect of key parameter values. The results show that the greatest effect on the performance and solution policy is related to changes in failure probabilities.

The main limitation of our model is the assumption that SBs have unlimited capacity which may be unrealistic from a practical perspective. One possible extension for the model would be adding constraints that limit the number of elements assigned to each SB. Another potential extension would be considering a weighted cost function. In the current model we assume that the checking cost and rework costs are equally important. One could explore incorporating weights in the objective function and apply a multi-objective framework. Finally, it would be worthwhile to improve the SnC algorithm and obtain better lower bounds for cases with large failure probabilities. As mentioned before, the quality of the lower bound solutions in our proposed approach, heavily depends on the failure probabilities and it would be an interesting

future research direction to explore other methods to obtain a lower bound for the RRTP.

Our study is focused on the reliable design of SBs in radiation therapy. However our modeling approach is general and applies to quality assurance problems in various different industries.

# Chapter 5

# A Two-Stage Stochastic Programming Model

In this chapter, we propose a stochastic programming model to capture uncertainties in the radiation therapy (RT) process. In Section 5.1, we provide an overview of stochastic programming. In Section 5.2, we propose a mean-risk two-stage stochastic programming model. In Section 5.3, we describe the sample average approximation method used to solve our model. In Section 5.4, we present the computational results and we conclude in Section 5.5.

## 5.1 Stochastic Programming

Stochastic programming (SP) incorporates uncertainty in optimization problems by assuming probability distributions on problem parameters. The general form of a stochastic programming problem is given by [73]:

$$\min_{x \in X} \{ f(x) := E[F(x, \xi(\omega))] \}, \tag{5.1a}$$

where $F(x, \xi)$ is real valued function of decision variable $x \in R^n$ and random variable $\xi \in R^d$, $X$ is a subset of $R^n$, and $\xi(\omega)$ is a random vector with probability distribution $P$ which is assumed to be known.

The most widely used SP models are two-stage models. In two-stage problems, there are two

types of decision variables: *first stage* and *second stage* variables. The decision maker decides on the first stage variables without knowing the realization of random parameters. After observing the realization of random parameters, the values of the second stage or *recourse* variables are selected. The objective is to find a policy that minimizes the sum of the first stage costs and the expected value of the second stage costs [5]. The standard form of a two-stage stochastic program with recourse is given by:

$$\min_{x \in X} \{f(x) := c^T x + E[Q(x, \xi(\omega))]\}, \tag{5.2a}$$

where

$$Q(x, \xi) = \min_{y \in Y} \{q^T y : Wy \geq h - Tx, y \geq 0\}. \tag{5.3a}$$

In the second stage, $\xi = (q, T, W, h)$ represents a particular realization of the random vector $\xi(\omega)$. The expectation in (5.2a) is with respect to the probability distribution of $\xi(\omega)$. The first stage consists of Problem (5.2a) with decision variables $x \in R^n$ that are selected before the realization of $\xi(\omega)$. Given the first stage decisions and realization $\xi$, the second stage consists of Problem 5.3a with variables $y \in R^n$ [5].

As explained in Chapter 4, there is a close relationship between our problem and the reliable facility location problem (RFLP) which considers supply chains that are subject to failure. In the literature, there are only a few studies that addressed stochastic reliable facility location problems. Jalali et al. [52] studied the bi-objective reliable capacitated facility location problem for a three-level supply chain that consists of plants, distribution centers (DC) and customers in which DCs are subject to random failures. They assumed that first stage decisions are related to DC and plant location and second stage decisions are related to demand allocation. They formulated a two-stage SP model and evaluated the effectiveness of the two stage model using two measures: (i) expected value of perfect information (EVPI) and (ii): value of stochastic solution (VSS).

Shen et al. [109] proposed a two-stage SP model for uncapacitated reliable facility location problem (URFLP) where the first stage decision is to determine which facilities to open without

knowing which facilities will be operational. In the second stage customers are assigned to operational facilities to minimize the total expected cost. Using a similar stochastic programming approach, Gade and Pohl [42] modeled the capacitated-facility location problem (CFLP) where the capacity of facilities is subject to uncertainty. They assumed a dummy facility that has a capacity to meet the entire customer demand. Aydin and Murat [12] formulated a capacitated reliable facility location problem (CRFLP) as a two-stage SP model. They developed a heuristic-based sample average approximation algorithm to reduce the computational burden resulted from a large number of scenario realizations.

All of the papers in the literature have developed SP models considering the uncertainty related to facility failures or their capacity. The major difference between our models and the existing models is that we take into account the randomness of error probabilities and rework costs as well as failure probabilities. Although, these parameters were assumed to be deterministic in the mixed-integer programming model in Chapter 4, they are not exactly known in most real-world situations, because estimating the parameters is difficult and sufficient data is not available.

## 5.2 A Two Stage Stochastic Programming Model for RRTP

Given a finite set of scenarios, where each scenario specifies a realization of random parameters, we formulate the reliable radiation therapy problem (RRTP) as a two-stage stochastic programming model. The first stage decisions determine the location of SBs and the elements assigned to each SB. In the second stage uncertain parameters: (i) error probabilities, (ii) failure probabilities; and (iii) rework costs are realized. Each uncertain parameter follows a continuous probability distribution which we will describe in Section 5.4. Upon realization of uncertain parameters, for each element $i$ rework cost is incurred based on the probability that it is caught by SB $j$. The objective is to minimize the total expected cost that includes the first stage cost and the expected second-stage cost.

We define $\xi$ as the vector of uncertain parameters i.e., error probabilities, failure probabilities and rework costs. Let $\xi^s = \{\gamma_i^s, \beta_{ij}^s, r_{ij}^s\}$ denote a given realization of uncertain parameters in the second stage.

Let $X$ denote the SB location vector and $Y$ denote the element assignment vector. Then the two-stage stochastic programming model called RRTP-SP can be formulated as :

$$\min f(X,Y) = \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{M}_i} \sum_{k\in\mathcal{K}_i:k\le\rho_{ij}} t_{ij}Y_{ijk} + E[Q(X,Y,\xi)] \tag{5.4a}$$

$$\text{s.t.} \sum_{j\in\mathcal{J}\setminus J} X_j = b \tag{5.4b}$$

$$\sum_{j\in\mathcal{M}_i:\rho_{ij}\ge k} Y_{ijk} + \sum_{s<k} Y_{iJs} = 1 \qquad k\in\mathcal{K}_i, i\in\mathcal{I} \tag{5.4c}$$

$$\sum_{k\in\mathcal{K}_i} Y_{ijk} = 1 \qquad i\in\mathcal{I} \tag{5.4d}$$

$$\sum_{k\in\mathcal{K}_i:k\le\rho_{ij}} Y_{ijk} \le X_j \qquad j\in\mathcal{M}_i, i\in\mathcal{I} \tag{5.4e}$$

$$Y_{ijk} \le \sum_{m\in\mathcal{M}_i:k-1\le\rho_{im}<\rho_{ij}} Y_{im,k-1} \qquad i\in\mathcal{I}, j\in\mathcal{M}_i, k\in\mathcal{K}_i : 1<k\le\rho_{ij} \tag{5.4f}$$

$$X_j, Y_{ijk} \in \{0,1\} \qquad i\in\mathcal{I}, j\in\mathcal{M}_i, k\in\mathcal{K}_i : k\le\rho_{ij} \tag{5.4g}$$

For any first-stage decision $(X,Y)$ and realization of $\xi$ denoted by $\xi^s$, $Q(X,Y,\xi^s)$ is defined as:

$$\min Q((X,Y),\xi^s) = \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{M}_i} \gamma_i r_{ij} Z_{ij} \tag{5.5a}$$

$$\text{s.t. } Z^s_{ij} = \sum_{k\in\mathcal{K}_i:k\le\rho_{ij}} a^s_{ik}Y_{ijk} \qquad i\in\mathcal{I}, j\in\mathcal{M}_i\setminus J \tag{5.5b}$$

$$Z^s_{iJ} = \sum_{k\in\mathcal{K}_i:k\le\rho_{iJ}} (\beta^s_i)^{k-1}Y_{iJk} \qquad i\in\mathcal{I} \tag{5.5c}$$

$$Z^s_{ij} \ge 0 \qquad i\in\mathcal{I}, j\in\mathcal{J} \tag{5.5d}$$

Parameter $a^s_{ik}$ is the probability that element $i$ needs to be checked $k$ times to be caught under realization $\xi^s$, in other words, the probability that the first $k-1$ SB fails to catch it, and $k^{th}$ SB catches it. We assume that failure probabilities, $(\beta_i)$, only depend on elements to facilitate the calculation of $a^s_{ik}$ in the second stage. Without this assumption, it would be necessary to incorporate the first-stage decisions $(X,Y)$ to calculate the $a^s_{ik}$. For each element $i\in\mathcal{I}$ at safety level $k\in\mathcal{K}_i$ we calculate $a^s_{ik} = (\beta^s_i)^{k-1}(1-\beta^s_i)$ under realization $\xi^s$. Note that $\sum_{k\in\mathcal{K}_i} a^s_{ik} \le 1$.

Table 5.1: Summary of Notation

| | |
|---|---|
| **Indices** | |
| $\mathcal{I}$ | set of elements to be checked, $\{1, \ldots, I\}$, |
| $\mathcal{J}$ | set of candidate safety barrier locations, $\{1, \ldots, J\}$, (SB $J$ is the patient) |
| $\mathcal{M}_i \subseteq \mathcal{J}$ | set of SBs that can include element $i \in \mathcal{I}$, |
| $\mathcal{K}_i$ | set of (safety) levels for element $i$, $\{1, \ldots, \lvert\mathcal{M}_i\rvert\}$ |
| $S$ | set of scenarios, $\{1, \ldots, \lvert S\rvert\}$ |
| **Parameters** | |
| $\gamma_i$ | the probability of having an error (error) in element $i \in \mathcal{I}$, |
| $t_{ij}$ | time required to check and verify element $i \in \mathcal{I}$ at SB $j \in \mathcal{M}_i$ , |
| $r_{ij}$ | rework cost for correcting error $i$ detected at SB $j \in \mathcal{M}_i$ |
| $\beta_{ij}$ | probability of failing to detect error $i$ at SB $j \in \mathcal{M}_i$ |
| $\rho_{ij}$ | the highest possible safety level for element $i \in \mathcal{I}$ in SB $j \in \mathcal{M}_i$ |
| **Decision Variables** | |
| $X_j$ | $\begin{cases} 1 & \text{if SB } j \in J \text{ is available} \\ 0 & \text{otherwise.} \end{cases}$ |
| $Y_{ijk}$ | $\begin{cases} 1 & \text{if element } i \in \mathcal{I} \text{ is included in SB } j \in \mathcal{M}_i \text{ at safety level } k \in \{1, \ldots, \rho_{ij}\} \\ 0 & \text{otherwise.} \end{cases}$ |
| $a^s_{ik}$ | probability that element $i$ needs to be checked $k$ times to be caught under realization $\xi^s$ |
| $Z^s_{ij}$ | probability that error $i$ is caught by SB $j$ under realization $\xi^s$ |

The second stage variable is $Z^s_{ij}$ the probability that error $i$ is caught by SB $j$ under realization $\xi^s$. Note that $Z^s_{iJ}$ is the probability that error $i$ reaches the patient under realization $\xi^s$. First-stage constraints (5.4b)-(5.4g) are the same as the constraints of the RRTP formulated in Chapter 4. Second-stage constraint (5.5b) shows the calculation of $Z^s_{ij}$ for $j \in \mathcal{M}_i \setminus J$. Under realization $\xi^s$, the probability that error $i$ is caught by SB $j$ before the patient will be equal to $a^s_{ik}$ given that element $i$ is checked by SB $j$ at $k$th level. Similarly, the second-stage constraint (5.5c) shows the calculation of $Z^s_{iJ}$. The probability that error in element $i$ reaches a patient, is equal to the probability that the first $k-1$ SBs before the patient fail to catch the error. Constraint (5.5d) restricts the $Z^s_{ij}$ variable to be non-negative. The objective (5.4a) minimizes the cost of first-stage decisions (total checking cost) plus the expected rework cost. Table 5.1 presents a summary of our notation.

### 5.2.1 Mean-risk Function

The stochastic programming model (5.4) is risk neutral as it is concerned with optimizing an expectation-based criterion [4]. However, decision makers may have different risk attitudes and preferences and therefore it would be helpful to incorporate risk measures in the model. A common approach to include risk in a stochastic programming model is to have a weighted mean-risk objective and measure risk using some dispersion statistic such as variance [4]. Using a mean-risk objective, the general form of a stochastic programming model (5.2a) is given by:

$$\min \{E[f(x,w)] + \lambda \mathcal{D}[f(x,w)] : x \in X\},$$

where $\mathcal{D}$ is a dispersion statistic and $\lambda$ is a non-negative weight coefficient for risk. The inclusion of risk can add to the complexity of solution, for example, it is known that using the classical mean-variance criterion in two-stage stochastic linear programming results in non-convexity [104, 119]. However, Shabbir Ahamed [4] showed a number of common risk statistics that preserve convexity and therefore more suitable for optimization. We use two of these risk measures to incorporate risk in our stochastic model: (i) *central semi-deviation* and (ii) *semi-deviation from a target*. In the following we describe each criterion.

#### (i) Central Semi-deviation

Given a random variable $Y$ belonging to the linear space $\chi_p$, the $p^{th}$ central semi-deviation [96, 4] is defined as :

$$\delta_p[Y] = (E[(Y - E[Y])_+^p])^{1/p}$$

It is known that the mean-semi-deviation objective:

$$g_{\lambda,\delta_p}[Y] = E[Y] + \lambda \delta_p[Y]$$

is convexity preserving for all $p \geq 1$ and $\lambda \in [0, 1]$.

**(ii) Semi-deviation From a Target**

For $Y \in \chi_p$ and a fixed target $T \in R$, the $p^{th}$ semi-deviation from $T$ [96, 4] is defined as :

$$S_{T,p}[Y] = (E[(Y - T)_+^p])^{1/p}$$

It is known that the mean-risk objective:

$$g_{\lambda,T,p}[Y] = E[Y] + \lambda S_{T,p}[Y]$$

is convexity preserving for all $p \geq 1$, $\lambda \geq 0$ and $T$.

Using the mean-risk approach with central semi-deviation as risk statistics, RRTP-SP is reformulated as :

$$\min\ f(X,Y) = \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{M}_i} \sum_{k\in\mathcal{K}_i:k\leq\rho_{ij}} t_{ij}Y_{ijk} + E[Q(X,Y,\xi)] \tag{5.11a}$$

$$\text{s.t. } (5.4b) - (5.4f)$$

$$X_j, Y_{ijk} \in \{0,1\} \qquad i \in \mathcal{I}, j \in \mathcal{M}_i, s \in \mathcal{S}, k \in \mathcal{K}_i : k \leq \rho_{ij}$$

For any first-stage decision $(X,Y)$ and realization of $\xi$ denoted by $\xi^s$, $Q(X,Y,\xi^s)$ is defined as:

$$\min\ Q((X,Y),\xi^s) = \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{M}_i} \gamma_i r_{ij} Z_{ij} + \lambda V^s \tag{5.12a}$$

$$\text{s.t. } Z_{ij}^s = \sum_{k\in\mathcal{K}_i:k\leq\rho_{ij}} a_{ik}^s Y_{ijk} \qquad i \in \mathcal{I}, j \in \mathcal{M}_i \setminus J \tag{5.12b}$$

$$Z_{iJ}^s = \sum_{k\in\mathcal{K}_i:k\leq\rho_{iJ}} (\beta_i^s)^{k-1} Y_{iJk} \qquad i \in \mathcal{I} \tag{5.12c}$$

$$V^s \geq \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{M}_i} \gamma_i^s r_{ij}^s Z_{ij}^s - E[Q(X,Y,\xi)] \tag{5.12d}$$

$$Z_{ij}^s \geq 0,\ V^s \geq 0 \qquad i \in \mathcal{I}, j \in \mathcal{J} \tag{5.12e}$$

Note $V^s$ is the semi-deviation from the mean under realization $\xi^s$. Constraint (5.12d) shows the calculation of $V^s$. The objective is to minimize the weighted mean-risk function. Varying the trade-off coefficient $\lambda$ allows the decision makers to evaluate different policies [90]. To use

*semi-deviation from a target* as the risk measure, we simply replace $E[Q(X, Y, \xi)]$ the right-hand side of (5.12d) with a target value $T$.

The major difficulty in solving the RRTP-SP is evaluating the objective function. For a given SB configuration, we have to calculate the expected value of the recourse function $Q((X, Y), \xi)$. For continuous distributions, computing the expectation is quite difficult because it requires taking multi-dimensional integrals. For discrete distributions, calculating the expectation involves solving the recourse problem with respect to all realizations of the uncertain parameters which may be prohibitive [105, 5]. In the next section we describe the sample average approximation (SAA) method that we use to solve the proposed stochastic program.

## 5.3 Sample Average Approximation (SAA) Method

In stochastic programming, the goal is to minimize the cost function taking into account the expected value of the recourse function based on all possible realizations of the uncertain parameters. However, as mentioned earlier, when the parameters uncertainty is characterized by a continuous probability distribution, we end up with an infinite scenario set and it is impossible to solve the stochastic model. To address this issue, we use the sample average approximation (SAA) method. SAA is a sampling-based procedure that has been used to solve stochastic programs with a large number of scenarios [61]. In the SAA approach, a number of scenarios $N$ are sampled each corresponding to a realization of uncertain parameters and an approximate problem called the SAA problem is formulated using these sampled scenarios. For a given sample $\xi^1, \xi^2, ..., \xi^N$ of $N$ realizations, the SAA problem has the same form as a two-stage SP with scenarios $\xi^n, n = 1, ...N$ each taken with the same probability $p^n = \frac{1}{N}$. To obtain the sample average approximation for RRTP-SP, we approximate the expected recourse costs $E[Q((X, Y), \xi)]$ by the sample average function $N^{-1} \sum_{n=1}^{N} Q((X, Y), \xi^n)$. The approximation of the objective function $f(.)$ based on a sample of $N$ realizations will be:

$$\min_{x \in X} \{\hat{f}_N(X, Y) := \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{M}_i} \sum_{k \in \mathcal{K}_i : k \leq \rho_{ij}} t_{ij} Y_{ijk} + \frac{1}{N} \sum_{n=1}^{N} Q((X, Y), \xi^n)\} \tag{5.13a}$$

Therefore the SAA problem is given by:

$$\min \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{M}_i}\sum_{k\in\mathcal{K}_i:k\le\rho_{ij}} t_{ij}Y_{ijk} + \frac{1}{N}\sum_{n=1}^{N}(\sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{M}_i} p^{s_n}\gamma_i^{s_n}r_{ij}^{s_n}Z_{ij}^{s_n} + \lambda\sum_{s_n\in\mathcal{S}_n} p^{s_n}V^{s_n}) \tag{5.14a}$$

$$\text{s.t.} \sum_{j\in\mathcal{J}\setminus J} X_j = b \tag{5.14b}$$

$$\sum_{j\in\mathcal{M}_i:\rho_{ij}\ge k} Y_{ijk} + \sum_{s<k} Y_{iJs} = 1 \qquad k\in\mathcal{K}_i, i\in\mathcal{I} \tag{5.14c}$$

$$\sum_{k\in\mathcal{K}_i} Y_{ijk} = 1 \qquad i\in\mathcal{I} \tag{5.14d}$$

$$\sum_{k\in\mathcal{K}_i:k\le\rho_{ij}} Y_{ijk} \le X_j \qquad j\in\mathcal{M}_i, i\in\mathcal{I} \tag{5.14e}$$

$$Y_{ijk} \le \sum_{m\in\mathcal{M}_i:k-1\le\rho_{im}<\rho_{ij}} Y_{im,k-1} \qquad i\in\mathcal{I}, j\in\mathcal{M}_i, k\in\mathcal{K}_i : 1<k\le\rho_{ij} \tag{5.14f}$$

$$Z_{ij}^{s_n} = \sum_{k\in\mathcal{K}_i:k\le\rho_{ij}} a_{ik}^{s_n}Y_{ijk} \qquad i\in\mathcal{I}, j\in\mathcal{M}_i\setminus J, s_n\in\mathcal{S}_n \tag{5.14g}$$

$$Z_{iJ}^{s_n} = \sum_{k\in\mathcal{K}_i:k\le\rho_{iJ}} (\beta_i^{s_n})^{k-1}Y_{iJk} \qquad i\in\mathcal{I}, s_n\in\mathcal{S}_n \tag{5.14h}$$

$$V^{s_n} \ge \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{M}_i}\gamma_i^{s_n}r_{ij}^{s_n}Z_{ij}^{s_n} - \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{M}_i}\sum_{w_n\in\mathcal{S}_n} p^{w_n}\gamma_i^{w_n}r_{ij}^{w_n}Z_{ij}^{w_n} \qquad s_n\in\mathcal{S}_n \tag{5.14i}$$

$$X_j, Y_{ijk}\in\{0,1\},\ Z_{ij}^{s_n}\ge 0,\ V^{s_n}\ge 0 \tag{5.14j}$$

$$i\in\mathcal{I}, j\in\mathcal{M}_i, s_n\in\mathcal{S}_n, k\in\mathcal{K}_i : k\le\rho_{ij}$$

Let $\hat{v}_N$ denote the optimal value of the SAA problem and $(\hat{X}_N, \hat{Y}_N)$ denote its optimal solution. The $\hat{f}_N(X,Y)$ is an unbiased estimator of $f(X,Y)$ for any $(X,Y)$[73]. If $\xi^i$ are $i.i.d$ samples, then $Var[\hat{f}_N(X,Y)] = \sigma^2(X,Y)/N$ where $\sigma^2(X,Y) := Var[Q((X,Y),\xi)]$ [73].

The convergence of the optimal solution depends on the choice of the sample size. The probability that $(\hat{X}_N, \hat{Y}_N)$ is equal to the optimal solution, approaches 1 exponentially fast in $N$. However, there is little to say about finding the optimal objective value of the true problem, i.e., $v^*$. Even with a large $N$ and when $(\hat{X}_N, \hat{Y}_N)$ is identical to the optimal solution of true problem $(X_N^*, Y_N^*)$, the corresponding optimal objective function values $\hat{v}_N$ may be different from the true optimum $v^*$ [73]. We use a sampling methodology to obtain statistical lower and upper bounds on the true optimal value $v^*$. The idea of the statistical lower bound and statistical

upper bound was introduced by [89] and was developed further by [76]. In the following, we describe the details on obtaining these bounds.

### 5.3.1 Lower Bound

Let $v^*$ and $\hat{v}_N$ denote the optimal values of the true and SAA problems respectively. It can be shown [73] that:

$$E[\hat{v}_N] \leq v^*$$

To estimate the expected value $E[\hat{v}_N]$, we solve the SAA problem $M$ times each with $N$ scenarios. Let $\hat{v}_N^1$, $\hat{v}_N^2$, ..., $\hat{v}_N^M$ be the corresponding objective values. We compute:

$$L_{N,M} = \frac{1}{M}\sum_{m=1}^{M} \hat{v}_N^m$$

which is the average of the objective values of the $M$ SAA problems. The $L_{N,M}$ is an unbiased estimator of $E[\hat{v}_N]$ and provides a statistical estimate of a lower bound of the true optimum [5, 63]. If the $M$ batches of $\xi^{1,j}, \xi^{2,j}, ..., \xi^{N,j}$, $j = 1, .., M$ are i.i.d, by Central Limit Theorem we will have [73]:

$$\sqrt{M}[L_{N,M} - E(\hat{v}_N)] \Rightarrow N(0, \sigma_L^2), \text{ as } M \to \infty$$

where $\sigma_L^2 := Var[\hat{v}_N]$ and "$\Rightarrow$" denotes convergence in distribution. The sample variance estimator of $\sigma_L^2$ is:

$$s_L^2(M) = \frac{1}{M-1}\sum_{m=1}^{M}(\hat{v}_N^m - L_{N,M})^2$$

Using $s_L(M)$, we can obtain an approximate $(1-\alpha)$-confidence interval for $E[\hat{v}_N]$ to be :

$$[L_{N,M} - \frac{z_{\alpha/2}s_L(M)}{\sqrt{(M)}}, L_{N,M} + \frac{z_{\alpha/2}s_L(M)}{\sqrt{(M)}}]$$

For small values of M, one can use $t_{\alpha/2,M-1}$ values instead of $z_{\alpha/2}$, which will produce slightly larger confidence intervals.

### 5.3.2 Upper Bound

For any feasible first-stage decisions $(\hat{X}, \hat{Y})$ obtained by solving the SAA problem, the value:

$$f(\hat{X}, \hat{Y}) = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{M}_i} \sum_{k \in \mathcal{K}_i : r \leq \rho_{ij}} t_{ij} \hat{Y}_{ijk} + E[Q((\hat{X}, \hat{Y}), \xi(\omega))]$$

is an upper bound on the true optimum $v^*$. We can obtain an estimate of an upper bound for $v^*$ by generating $M$ independent batches of samples of size $N'$ scenarios [73]. For each batch, we compute the corresponding $\hat{f}^m_{N'}(\hat{X}, \hat{Y})$ as :

$$\hat{f}^m_{N'}(\hat{X}, \hat{Y}) = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{M}_i} \sum_{k \in \mathcal{K}_i : k \leq \rho_{ij}} t_{ij} \hat{Y}_{ijk} + \frac{1}{N'} \sum_{n=1}^{N'} \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{M}_i} p^{s_n} \gamma_i^{s_n} r_{ij}^{s_n} \hat{Z}_{ij}^{s_n}$$

Then the average value which is defined as :

$$U_{N',M}(\hat{X}, \hat{Y}) = \frac{1}{M} \sum_{m=1}^{M} \hat{f}^m_{N'}(\hat{X}, \hat{Y})$$

is an unbiased estimator of $f(\hat{X}, \hat{Y})$ and provides a statistical estimate of an upper bound on the true optimum. Generating $M$ batches of $\xi^{1,j}, \xi^{2,j}, ..., \xi^{N',j}$, $j = 1, .., M$ that are i.i.d, by Central Limit Theorem we will have [73]:

$$\sqrt{M}[U_{N',M}(\hat{X}, \hat{Y}) - f(\hat{X}, \hat{Y})] \Rightarrow N(0, \sigma^2_U(\hat{X}, \hat{Y})) \text{ as } M \to \infty$$

where $\sigma^2_U(\hat{X}, \hat{Y}) := Var[\hat{f}_{N'}(\hat{X}, \hat{Y})]$. We estimate the $\sigma^2_U(\hat{X}, \hat{Y})$ by the sample variance estimator $s^2_U(\hat{X}, \hat{Y}, M)$ defined as:

$$s^2_U(\hat{X}, \hat{Y}, M) = \frac{1}{M-1} \sum_{m=1}^{M} (\hat{f}^m_{N'}(\hat{X}, \hat{Y}) - U_{N',M}(\hat{X}, \hat{Y}))^2$$

By replacing $\sigma_U^2(\hat{X},\hat{Y})$ with $s_U^2(\hat{X},\hat{Y},M)$, we can obtain an approximate $(1-\alpha)$-confidence interval for $f(\hat{X},\hat{Y})$:

$$[U_{N',M}(\hat{X},\hat{Y}) - \frac{z_{\alpha/2}s_U(\hat{X},\hat{Y},M)}{\sqrt(M)}, U_{N',M}(\hat{X},\hat{Y}) + \frac{z_{\alpha/2}s_U(\hat{X},\hat{Y},M)}{\sqrt(M)}]$$

For small values of M, one can use $t_{\alpha/2,M-1}$ values instead of $z_{\alpha/2,M-1}$, which will result in slightly larger confidence intervals [73].

### 5.3.3 Estimating the Gap

For a given feasible solution $(\hat{X},\hat{Y})$, we estimate the optimality gap $f(\hat{X},\hat{Y}) - v^*$ to assess the quality of the solution. Using the statistical estimates of upper and lower bounds, we can calculate an estimate for the gap according to :

$$Gap(\hat{X},\hat{Y}) := U_{N',M}(\hat{X},\hat{Y}) - L_{N,M}$$

The $Gap(\hat{X},\hat{Y})$ tends to $f(\hat{X},\hat{Y}) - v^*$ as $N, N'$ and $M$ tend to $\infty$. Three factors contribute to the error in the statistical estimator $Gap(\hat{X},\hat{Y})$ of the true gap $f(\hat{X},\hat{Y}) - v^*$. They are :

- (i) : variance of $U_{N',M}(\hat{X},\hat{Y})$,
- (ii) : variance of $L_{N,M}$, and
- (iii) : bias $v^* - E[\hat{v}_N]$

$U_{N',M}(\hat{X},\hat{Y})$ and $L_{N,M}$ are unbiased estimators of $f(\hat{x})$ and $E[\hat{v}_N]$. The variance of $U_{N',M}(\hat{X},\hat{Y})$ and $L_{N,M}$ can be estimated from the generated samples. The variance of the lower bound estimate is equal to :

$$\frac{S_M^2}{M} = \frac{1}{M(M-1)}\sum_{m=1}^{M}(\hat{v}_N^m - L_{N,M})$$

and the variance of the upper bound estimate is calculated as :

$$\frac{S_M^2(\hat{X},\hat{Y})}{M} = \frac{1}{M(M-1)}\sum_{m=1}^{M}(\hat{f}_N^m(\hat{X},\hat{Y}) - U_{N',M}(\hat{X},\hat{Y}))$$

Therefore, variance of the gap is calculated as :

$$Var(\hat{Gap}(\hat{X},\hat{Y})) = \frac{S_M^2}{M} + \frac{S_M^2(\hat{X},\hat{Y})}{M}$$

The variance of the lower and upper bound estimators may be reduced by increasing the batch and the sample size, i.e., $N, N'$ and $M$.

The $Gap(\hat{X},\hat{Y})$ overestimates the true gap $f(\hat{X},\hat{Y}) - v^*$ and has a bias of $v^* - E[\hat{v}_N]$ which is the most serious problem in estimating the gap [73]. This bias can be reduced by increasing the number of scenarios or by changing the sampling method (e.g., using Latin Hypercube Sampling). However, the SAA becomes computationally expensive as the number of sample scenarios increases. Increasing $M$ requires solving more instances and increasing $N$ results in solving larger problems which can significantly affect the computational time [73]. A major challenge with using the SAA method is the selection the appropriate number of instances to achieve a solution with reasonable accuracy in a reasonable time. In the next section, we will perform computational experimentation to find an adequate number of scenarios.

## 5.4 Computational Results

In this section we present the numerical results for the two-stage stochastic programming model for the reliable radiation therapy problem (RRTP-SP). The parameter values for the model are estimated based on the case study we conducted at the Department of Radiation Oncology of University of North Carolina (UNC) Lineberger Comprehensive Cancer Center. We use uniform distributions to describe the uncertainty in the parameters of RRTP-SP. The uncertain parameters in the second stage are : (i) error probabilities, (ii) failure probabilities, and (iii) rework costs. We assume the checking cost and the penalty for an error reaching the patient are fixed. To generate the scenarios for the SAA problem, we sample the value for each parameter from a uniform distribution where the mean value of the distribution is calculated based on the UNC RT process using the approach described in Section 1.6.1 of Chapter 4. For each uncertain parameter we consider a uniform distribution on an interval between 30% below and 30% above the calculated mean.

Recall that the SAA method requires $M$ batches of the approximating stochastic problem each with $N$ scenarios [105]. For each batch, the candidate solution is evaluated using $N'$ sample scenarios. In our experiments, we consider problem instances with 25 and 50 elements with 10 safety barriers (i.e., $b = 10$). We use $N = 25,\ 50,\ 100,\ 150,\ 200$, $M = 50$ and $N' = 1000$. The results show that these parameters were sufficient to obtain good quality solutions and confidence intervals.

Table 5.2 shows the size of the SAA problem with mean-risk objective function corresponding to different values of $N$. Note that if we do not include a risk measure in the objective function, the complexity of the problem will not be affected by the number of scenarios. This is because we do not have second-stage decision variables so the size of the problem will remain the same regardless of the number of scenarios.

Table 5.2: Size of the SAA Problem

| n | N | total # of decision variables | # of binary variables | # of continuous variables | # of constraints |
|---|---|---|---|---|---|
| 25 | 25 | 11,396 | 2,621 | 8,775 | 11,651 |
| 25 | 50 | 11,421 | 2,621 | 8,800 | 11,676 |
| 25 | 100 | 11,471 | 2,621 | 8,850 | 11,726 |
| 25 | 150 | 11,521 | 2,621 | 8,900 | 11,776 |
| 25 | 200 | 11,571 | 2,621 | 8,950 | 11,826 |
| 50 | 25 | 45,146 | 11,996 | 33,150 | 45,701 |
| 50 | 50 | 45,171 | 11,996 | 33,175 | 45,726 |
| 50 | 100 | 45,221 | 11,996 | 33,225 | 45,776 |
| 50 | 150 | 45,271 | 11,996 | 33,275 | 45,826 |
| 50 | 200 | 45,321 | 11,996 | 33,325 | 45,876 |

The SAA problem instances were solved with Gurobi solver version 7.0.1. Each problem instance is run on a 64 -bit workstation on Windows with 5 core 3.4 GHz CPU and 32 GB of RAM. All CPU times are reported in seconds. Table 5.3 shows the computational time required for different number of scenarios, $N$, for problems with $n = 25$ and $n = 50$ elements and $b = 10$ safety barriers. We reported the computational times for the model with no risk measure, the model with *central semi-deviation* criterion and the model with criterion *semi-deviation from target*. The difficulty of solving the risk-neutral problem does not depend on

the number of scenarios. We only see a small increase in time as $N$ increases because the calculation of expected cost is over a larger number of scenarios. We also see that increasing $N$ has a relatively small effect on the computational effort when we use semi-deviation from target as a risk measure. However the results indicate that in the central semi-deviation model, increasing $N$ can significantly increase the computational time.

Table 5.3: Computational Time for the SAA Problem

| n | N | time per replication (sec) | | |
|---|---|---|---|---|
| | | no risk measure | semi-deviation from target | semi-deviation from mean |
| 25 | 25 | 23.7 | 34.8 | 75.4 |
| 25 | 50 | 23.9 | 37.5 | 206.2 |
| 25 | 100 | 27.6 | 45.7 | 775.8 |
| 25 | 150 | 32.4 | 51.5 | 1,507.4 |
| 25 | 200 | 33.7 | 59.5 | 2,619.9 |
| 50 | 25 | 107.9 | 162.8 | 353.2 |
| 50 | 50 | 108.2 | 157.7 | 961.8 |
| 50 | 100 | 132.5 | 207.1 | 3,403.9 |
| 50 | 150 | 141.4 | 233.2 | 7,356.4 |
| 50 | 200 | 167.5 | 269.8 | >18,000 |

Solving the SAA problem provides statistical estimates of the lower and upper bounds on the true optimum $v^*$. Using the estimates, we can obtain a statistical estimator for the optimality gap as shown in Section 5.3.3. Table 5.4 summarizes the computational results. Initially, we set the risk coefficient $\lambda$ to zero, therefore our objective function only includes the expected cost. In Table 5.4, we provide the CPU time, estimates for lower bound, upper bound and gap as well as variance, coefficient of variation (CV), confidence interval (CI) and half width for the gap. As seen from Table 5.4, some of the gap estimates are negative. This is not unexpected because both lower bound and upper bound are random variables [124]. However, because of the difficulty of interpreting the negative gaps and confidence intervals, we present two modifications to improve our gap estimates and avoid the negative ones:

(i) In the first modification, we examine the individual gaps corresponding to the $M$ batches of SAA problem and replace the negative gaps with zero [76]. Then we take the average of the individual gaps to obtain a gap estimator for the true problem. This approach guarantees that

Table 5.4: Computational Results, $\lambda = 0$

n=25, b=10

| N | LB | UB | gap | variance | CV | CI | half width |
|---|---|---|---|---|---|---|---|
| 25 | 236.99 | 238.02 | 1.02 | 0.58 | 0.74 | (-0.47, 2.52) | 1.49 |
| 50 | 237.17 | 237.92 | 0.75 | 0.27 | 0.69 | (-0.27, 1.76) | 1.01 |
| 100 | 238.25 | 238.14 | -0.11 | 0.13 | -3.27 | (-0.81, 0.59) | 0.70 |
| 150 | 238.17 | 237.97 | -0.20 | 0.11 | -1.69 | (-0.86, 0.46) | 0.66 |
| 200 | 237.38 | 238.02 | 0.65 | 0.08 | 0.44 | (0.09, 1.20) | 0.55 |

n=50, b=10

| N | LB | UB | gap | variance | CV | CI | half width |
|---|---|---|---|---|---|---|---|
| 25 | 349.27 | 349.49 | 0.22 | 0.68 | -1.39 | (-1.39, 1.83) | 1.61 |
| 50 | 348.39 | 349.56 | 1.17 | 0.34 | 0.04 | (0.04, 2.31) | 1.14 |
| 100 | 348.32 | 349.33 | 1.01 | 0.12 | 0.32 | (0.32, 1.70) | 0.69 |
| 150 | 349.40 | 349.27 | -0.13 | 0.15 | -0.90 | (-0.90, 0.63) | 0.77 |
| 200 | 348.71 | 349.37 | 0.66 | 0.10 | 0.00 | (0.00, 1.29) | 0.64 |

no negative gaps will arise and the variance will decrease.

(ii) Another possible modification is to generate $M$ batches of SAA with non-negative gaps. This means that we do not limit the number of iterations (i.e., batches) to $M$ and we keep running SAA until we obtain $M$ batches with non-negative gaps. Then, we take the average of these $M$ gaps as a statistical estimator for true gap. Again, this ensures non-negative gap estimates and results in variance reduction.

The results associated with these two approaches are shown in Table 5.5 and Table 5.6. As we expected, with no negative gaps or confidence interval that include negative values, there is less variation in the CV values. In addition, it can be seen that the values for variance are smaller compared to the values in Table 5.4. Note that Table 5.5 presents two values for the lower bound estimator and the upper bound estimator because for iterations with negative gap (i.e. lower bound is higher is than the upper bound), the bounds have been adjusted so that they have zero gap. We can achieve this either by setting the upper bound equal to the lower bound or setting the lower bound equal to the upper bound. In Table 5.5, LB1 and UB1 correspond to the first case and LB2 and UB2 correspond to the second case.

Although the proposed modifications result in more reasonable and interpretable gaps and confidence intervals, there are some disadvantages associated with using these two approaches. In the first approach, we replace negative gaps with zero which causes an artificial decrease in

the gap estimator. We can avoid this issue, by applying the second approach, but we should note that this approach may greatly increase the computational effort as it requires running the model for additional iterations. In Table 5.6, the second column shows the number of iterations required to obtain $M = 50$ batches with non-negative gaps. It can be seen that in most cases the computational time has increased by more than 50%. To choose between these methods, one must consider the trade-off between the accuracy of gap estimator and the required computational time. Since we were able to afford the computational time associated with the second approach, we choose it as the primary method for obtaining the gap estimator.

Note that in Table 5.5 and Table 5.6, the gap estimator improves as the number of scenarios $N$, is increased. The maximum improvement is achieved by increasing $N$ from 25 to 50. Therefore, we choose 50 as the number of scenarios for remaining numerical experiments.

Table 5.5: Computational Results - Modification 1, $\lambda = 0$

n=25, b=10

| N | LB1 | UB1 | LB2 | UB2 | gap | variance | CV | CI | half width |
|---|---|---|---|---|---|---|---|---|---|
| 25 | 237.05 | 239.25 | 235.82 | 238.02 | 2.20 | 0.18 | 0.19 | (1.37, 3.02) | 0.826 |
| 50 | 237.32 | 239.45 | 235.75 | 237.88 | 2.13 | 0.12 | 0.17 | (1.44, 2.82) | 0.690 |
| 100 | 237.51 | 238.86 | 236.48 | 237.84 | 1.35 | 0.05 | 0.17 | (0.90, 1.81) | 0.456 |
| 150 | 237.67 | 238.68 | 237.06 | 238.08 | 1.02 | 0.04 | 0.19 | (0.64, 1.39) | 0.374 |
| 200 | 238.16 | 238.90 | 237.01 | 237.75 | 0.74 | 0.02 | 0.21 | (0.43, 1.04) | 0.302 |

n=50, b=10

| N | LB1 | UB1 | LB2 | UB2 | gap | variance | CV | CI | half width |
|---|---|---|---|---|---|---|---|---|---|
| 25 | 348.21 | 351.49 | 346.05 | 349.34 | 3.28 | 0.30 | 0.17 | (2.21, 4.36) | 1.078 |
| 50 | 348.86 | 350.65 | 347.53 | 349.31 | 1.79 | 0.13 | 0.20 | (1.07, 2.50) | 0.717 |
| 100 | 349.01 | 350.47 | 347.82 | 349.28 | 1.46 | 0.07 | 0.19 | (0.93, 1.99) | 0.530 |
| 150 | 349.40 | 350.36 | 348.32 | 349.27 | 0.95 | 0.03 | 0.19 | (0.60, 1.30) | 0.352 |
| 200 | 348.71 | 349.92 | 348.15 | 349.37 | 1.22 | 0.04 | 0.17 | (0.80, 1.63) | 0.413 |

### 5.4.1 Sensitivity Analysis

In this section, we present the results of sensitivity analysis on the parameter values. We focus on the effect of changing error probabilities, the penalty cost for an error reaching the patient, failure probabilities and checking cost. We solve the SAA problem with 25 elements and 10 safety barriers and we increase the values corresponding to each parameter by 50% and 100%.

Table 5.6: Computational Results - Modification 2, $\lambda = 0$

n=25, b=10

| N | M | LB | UB | gap | variance | CV | CI | half width |
|---|---|---|---|---|---|---|---|---|
| 25 | 82 | 233.60 | 238.11 | 4.51 | 0.19 | 0.10 | (3.66, 5.36) | 0.848 |
| 50 | 100 | 235.67 | 238.15 | 2.47 | 0.10 | 0.13 | (1.85, 3.10) | 0.626 |
| 100 | 103 | 235.89 | 238.16 | 2.27 | 0.06 | 0.11 | (1.77, 2.76) | 0.493 |
| 150 | 116 | 236.25 | 238.27 | 2.03 | 0.05 | 0.12 | (1.57, 2.48) | 0.458 |
| 200 | 106 | 236.32 | 237.98 | 1.66 | 0.04 | 0.12 | (1.26, 2.07) | 0.405 |

n=50, b=10

| N | M | LB | UB | gap | variance | CV | CI | half width |
|---|---|---|---|---|---|---|---|---|
| 25 | 94 | 344.21 | 349.49 | 5.28 | 0.25 | 0.10 | (4.29, 6.26) | 0.988 |
| 50 | 78 | 346.27 | 349.48 | 3.21 | 0.16 | 0.12 | (2.43, 3.98) | 0.775 |
| 100 | 88 | 347.06 | 349.56 | 2.49 | 0.09 | 0.12 | (1.89, 3.09) | 0.600 |
| 150 | 88 | 347.41 | 349.40 | 1.98 | 0.05 | 0.12 | (1.52, 2.44) | 0.459 |
| 200 | 89 | 347.69 | 349.57 | 1.88 | 0.06 | 0.13 | (1.40, 2.37) | 0.483 |

The rest of the parameters are $N = 50$, $N' = 1000$ and $M = 50$. The results are summarized in Tables 5.7, 5.8, 5.9, and 5.10. Comparing the four tables, we can see that changing failure probabilities had the greatest impact on the statistical lower bound, upper bound and variance while changing checking costs had the least effect on them. This is in line with the sensitivity analysis that we presented in Chapter 4. We observed that the total cost and the performance of our proposed algorithm were most impacted by changes in the failure probabilities.

Table 5.7: sensitivity analysis - increasing checking costs

| checking cost | LB | UB | gap | variance | CV | CI | half width |
|---|---|---|---|---|---|---|---|
| base | 234.79 | 238.00 | 3.21 | 0.14 | 0.12 | (2.48, 3.95) | 0.736 |
| increase by 50% | 263.45 | 266.58 | 3.13 | 0.13 | 0.12 | (2.42, 3.84) | 0.708 |
| increase by 100% | 291.10 | 294.14 | 3.04 | 0.13 | 0.12 | (2.33, 3.75) | 0.706 |

Table 5.8: sensitivity analysis - increasing error probabilities

| error probability | LB | UB | gap | variance | CV | CI | half width |
|---|---|---|---|---|---|---|---|
| base | 234.79 | 238.00 | 3.21 | 0.14 | 0.12 | (2.48, 3.95) | 0.736 |
| increase by 50% | 319.42 | 324.02 | 4.60 | 0.29 | 0.12 | (3.55, 5.66) | 1.058 |
| increase by 100% | 403.57 | 409.65 | 6.08 | 0.52 | 0.12 | (4.67, 7.49) | 1.407 |

Table 5.9: sensitivity analysis - increasing the penalty for an error reaching the patient

| penalty cost | LB | UB | gap | variance | CV | CI | half width |
|---|---|---|---|---|---|---|---|
| base | 234.79 | 238.00 | 3.21 | 0.14 | 0.12 | (2.48, 3.95) | 0.736 |
| increase by 50% | 299.78 | 304.35 | 4.57 | 0.29 | 0.12 | (3.52, 5.62) | 1.050 |
| increase by 100% | 364.29 | 370.30 | 6.01 | 0.51 | 0.12 | (4.62, 7.40) | 1.393 |

Table 5.10: sensitivity analysis - increasing failure probabilities

| failure probability | LB | UB | gap | variance | CV | CI | half width |
|---|---|---|---|---|---|---|---|
| base | 234.79 | 238.00 | 3.21 | 0.14 | 0.12 | (2.48, 3.95) | 0.736 |
| increase by 50% | 351.69 | 356.67 | 4.98 | 0.36 | 0.12 | (3.80, 6.15) | 1.175 |
| increase by 100% | 504.58 | 512.60 | 8.02 | 0.82 | 0.11 | (6.24, 9.80) | 1.779 |

### 5.4.2 Risk Measures

As described in Section 5.2.1, we consider a mean-risk model to allow decision makers to incorporate their risk preferences into the stochastic programming model [90]. We use two different criteria to measure the risk: (i) central semi-deviation or semi-deviation from mean and (ii) semi-deviation from a target. Each risk measure is weighed by a trade-off parameter $\lambda$. When $\lambda = 0$, we will have the risk neutral two-stage stochastic programming formulation which only considers the expected cost as the preference criterion.

In order to demonstrate the effect of incorporating the risk measure, we evaluate the objective function value using different values of $\lambda$. Table 5.11 shows the results for the model with central semi-deviation criterion. We can observe that increasing $\lambda$ increases the gap and variance. Table 5.12 shows the decomposition of the cost function for the lower and upper bound. The cost function consists of the expected cost plus the risk measure which is calculated by multiplying central semi-deviation by $\lambda$. When $\lambda$ is zero (i.e., the risk neutral model), the central semi-deviation is equal to 10.18 and 10.71 for lower and upper bound estimates, respectively. From the table, we can also see that as we increase $\lambda$, the mean increases and the central semi-deviation decreases meaning that larger $\lambda$ values would result in more conservative (risk-averse) policies. This is because when $\lambda$ is large we are giving more weight to worse

scenarios. The effect of varying $\lambda$ on the mean is shown in Figure 5.1.

Table 5.11: n=25, b=10, central semi-deviation - impact of $\lambda$ on SAA performance

| $\lambda$ | LB | UB | gap | variance | CV | CI | half width |
|---|---|---|---|---|---|---|---|
| 0 | 234.79 | 238.00 | 3.21 | 0.14 | 0.12 | (2.48, 3.95) | 0.736 |
| 1 | 245.15 | 248.64 | 3.49 | 0.18 | 0.12 | (2.65, 4.34) | 0.843 |
| 5 | 284.67 | 291.43 | 6.77 | 0.56 | 0.11 | (5.31, 8.23) | 1.461 |
| 10 | 332.53 | 344.94 | 12.41 | 1.39 | 0.09 | (10.10, 14.72) | 2.310 |
| 15 | 380.78 | 400.23 | 19.45 | 2.99 | 0.09 | (16.06, 22.84) | 3.388 |
| 20 | 430.95 | 457.11 | 26.16 | 6.89 | 0.10 | (21.02, 31.31) | 5.145 |

Table 5.12: n=25, b=10, central semi-deviation - cost elements

| $\lambda$ | LB mean | LB risk | LB semi-deviation | UB mean | UB risk | UB semi-deviation |
|---|---|---|---|---|---|---|
| 0 | 234.79 | 0.00 | 10.18 | 238.00 | 0.00 | 10.71 |
| 1 | 235.21 | 9.94 | 9.94 | 237.98 | 10.66 | 10.66 |
| 5 | 236.77 | 47.89 | 9.58 | 238.20 | 53.24 | 10.65 |
| 10 | 237.45 | 95.08 | 9.51 | 238.53 | 106.41 | 10.64 |
| 15 | 239.22 | 141.57 | 9.44 | 240.22 | 160.02 | 10.67 |
| 20 | 242.35 | 188.60 | 9.43 | 243.08 | 214.04 | 10.70 |

Table 5.13 shows the results for the case where semi-deviation from a target is used as the risk measure. The target value $T$ is set to 240 minutes. Again we observe that as we increase $\lambda$ the gap and variance estimates increase. The composition of total cost is shown in Table 5.14. For the risk-neutral model, the semi-deviation from target is equal to 0.04. The risk coefficient $\lambda$ balances the trade-off between the mean and the risk criterion. Therefore we observe that larger values for $\lambda$ result in lower semi-deviation from the target and higher mean values.

Figures 5.2 and 5.3 illustrate the effect of varying $\lambda$ on solution policies corresponding to the two risk-measures discussed above. In the figures, the rows correspond to the elements and columns correspond to the SBs. The shaded boxes represent the elements assigned to each SB. The last SB is the patient which includes all the elements highlighted in red. We can see that the effect of risk on solution policies is minimal suggesting policies are fairly robust to risk. Although the policies are more sensitive to the risk for the central semi-deviation case particularly for $\lambda$ between 5 and 10, compared to smaller or larger values of $\lambda$.

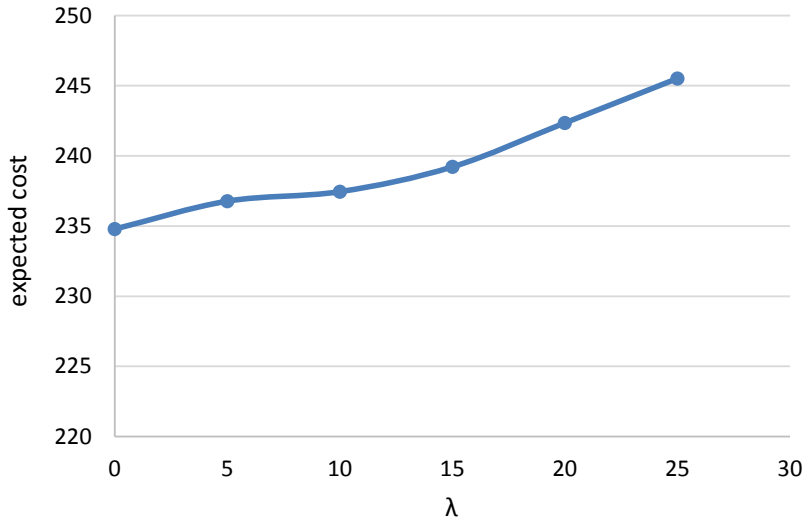


Figure 5.1: Effect of risk parameter $\lambda$ on expected cost- central semi-deviation

Table 5.13: n=25, b=10, T=240, semi-deviation from a target - impact of $\lambda$ on SAA performance

| $\lambda$ | LB | UB | gap | variance | CV | CI | half width |
|---|---|---|---|---|---|---|---|
| 0 | 234.79 | 238.00 | 3.21 | 0.14 | 0.12 | (2.48, 3.95) | 0.736 |
| 1 | 237.59 | 247.62 | 10.04 | 0.32 | 0.06 | (8.92, 11.15) | 1.113 |
| 5 | 237.69 | 286.30 | 48.61 | 0.46 | 0.01 | (47.28, 49.94) | 1.329 |
| 10 | 237.78 | 334.67 | 96.89 | 0.81 | 0.01 | (95.13, 98.65) | 1.761 |
| 15 | 237.86 | 383.05 | 145.19 | 1.35 | 0.01 | (142.91, 147.46) | 2.273 |
| 20 | 237.94 | 431.45 | 193.51 | 2.09 | 0.01 | (190.67, 196.34) | 2.833 |

Table 5.14: n=25, b=10, T=240, semi-deviation from a target - cost elements

| $\lambda$ | LB mean | LB risk | LB semi-deviation | UB mean | UB risk | UB semi-deviation |
|---|---|---|---|---|---|---|
| 0 | 234.79 | 0.00 | 0.040 | 238.00 | 0.00 | 9.67 |
| 1 | 237.55 | 0.03 | 0.033 | 237.96 | 9.67 | 9.67 |
| 5 | 237.59 | 0.10 | 0.020 | 237.96 | 48.33 | 9.67 |
| 10 | 237.61 | 0.17 | 0.017 | 237.97 | 96.70 | 9.67 |
| 15 | 237.62 | 0.24 | 0.016 | 237.98 | 145.07 | 9.67 |
| 20 | 237.62 | 0.32 | 0.016 | 237.98 | 193.46 | 9.67 |

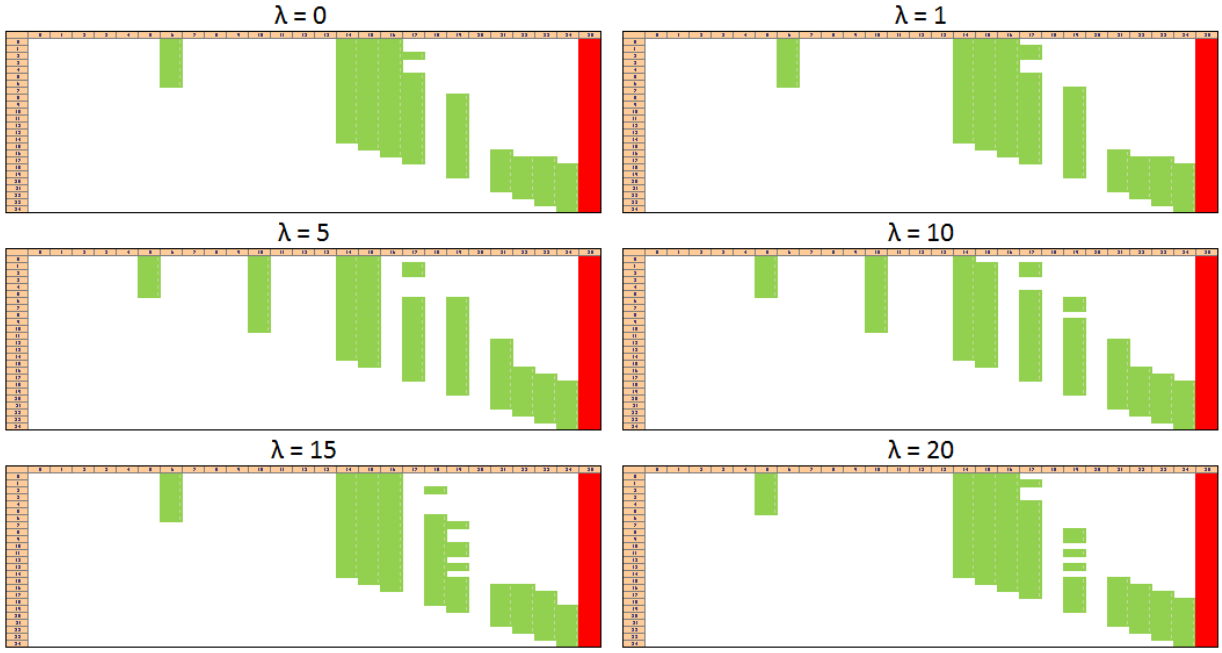


Figure 5.2: Effect of $\lambda$ on solution policy - central semi-deviation

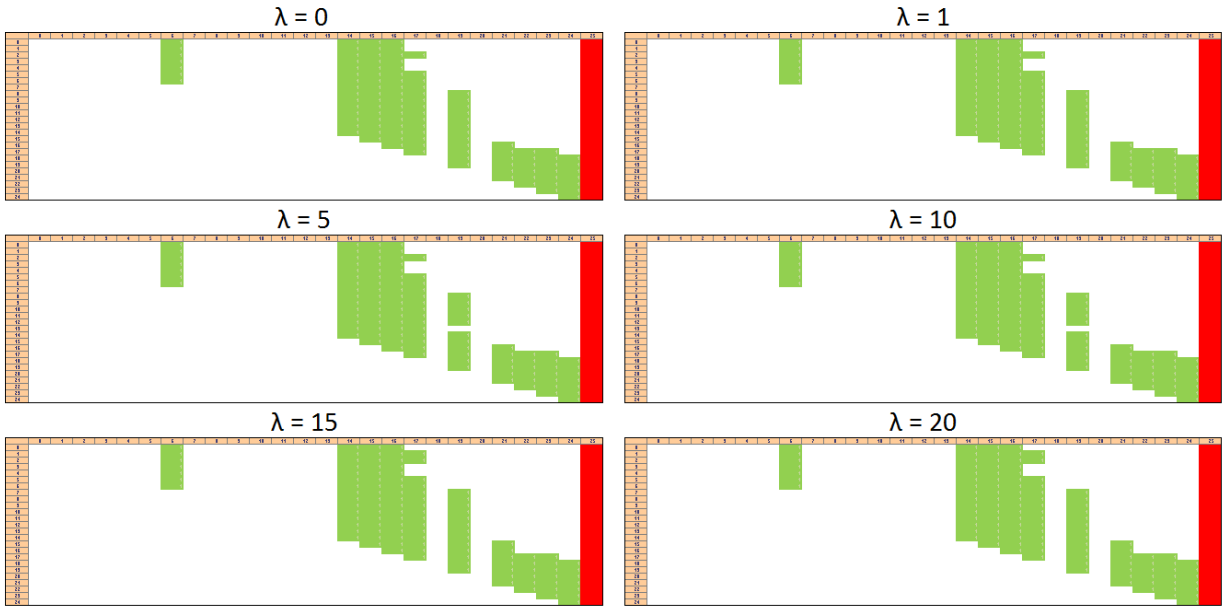


Figure 5.3: Effect of $\lambda$ on solution policy - semi-deviation from a target

To illustrate the effect of target value on the risk-mean function, we varied $T$ between 180 and 300 and solved the corresponding problems. The results for $\lambda = 5$ are summarized in Table 5.15 and Table 5.16. The results indicate that as we increase the target value, both the mean and risk components decrease and we obtain smaller gaps. Figure 5.4 shows the effect of changing $T$ on the mean. We can see that as the target value $T$ goes up, the mean decreases and then remains constant after $T = 252$ minutes. This is because for $T = 252$ minutes and above, the risk is equal to zero.

Table 5.15: n=25, b=10, $\lambda = 5$, semi-deviation from a target - impact of $T$ on SAA performance

| T | LB | UB | gap | variance | CV | CI | half width |
|---|---|---|---|---|---|---|---|
| 180 | 272.04 | 529.94 | 257.89 | 3.46 | 0.01 | (254.25, 261.54) | 3.65 |
| 192 | 255.62 | 470.45 | 214.83 | 2.00 | 0.01 | (212.05, 217.60) | 2.77 |
| 204 | 245.89 | 412.99 | 167.10 | 1.24 | 0.01 | (164.92, 169.29) | 2.18 |
| 216 | 240.61 | 360.88 | 120.27 | 0.80 | 0.01 | (118.51, 122.02) | 1.76 |
| 228 | 238.40 | 318.27 | 79.87 | 0.56 | 0.01 | (78.40, 81.34) | 1.47 |
| 240 | 237.69 | 286.30 | 48.61 | 0.46 | 0.01 | (47.28, 49.94) | 1.33 |
| 252 | 237.55 | 264.47 | 26.92 | 0.40 | 0.02 | (25.68, 28.16) | 1.24 |
| 264 | 237.55 | 250.82 | 13.27 | 0.36 | 0.05 | (12.10, 14.44) | 1.17 |
| 276 | 237.55 | 243.23 | 5.68 | 0.27 | 0.08 | (5.60, 7.62) | 1.01 |
| 288 | 237.55 | 239.72 | 2.17 | 0.19 | 0.11 | (3.17, 4.86) | 0.84 |
| 300 | 237.55 | 238.39 | 0.84 | 0.15 | 0.12 | (2.57, 4.07) | 0.75 |

Table 5.16: n=25, b=10, $\lambda = 5$, semi-deviation from a target - cost elements

| T | LB mean | LB risk | LB semi-deviation | UB mean | UB risk | UB semi-deviation |
|---|---|---|---|---|---|---|
| 180 | 238.18 | 33.86 | 6.77 | 238.32 | 291.62 | 58.32 |
| 192 | 238.17 | 17.45 | 3.49 | 238.31 | 232.14 | 46.43 |
| 204 | 238.06 | 7.83 | 1.57 | 238.22 | 174.77 | 34.95 |
| 216 | 237.84 | 2.77 | 0.55 | 238.05 | 122.83 | 24.57 |
| 228 | 237.67 | 0.73 | 0.15 | 237.98 | 80.29 | 16.06 |
| 240 | 237.59 | 0.10 | 0.02 | 237.96 | 48.33 | 9.67 |
| 252 | 237.55 | 0 | 0 | 237.96 | 26.51 | 5.30 |
| 264 | 237.55 | 0 | 0 | 237.96 | 12.86 | 2.57 |
| 276 | 237.55 | 0 | 0 | 237.96 | 5.27 | 1.05 |
| 288 | 237.55 | 0 | 0 | 237.96 | 1.76 | 0.35 |
| 300 | 237.55 | 0 | 0 | 237.96 | 0.44 | 0.09 |

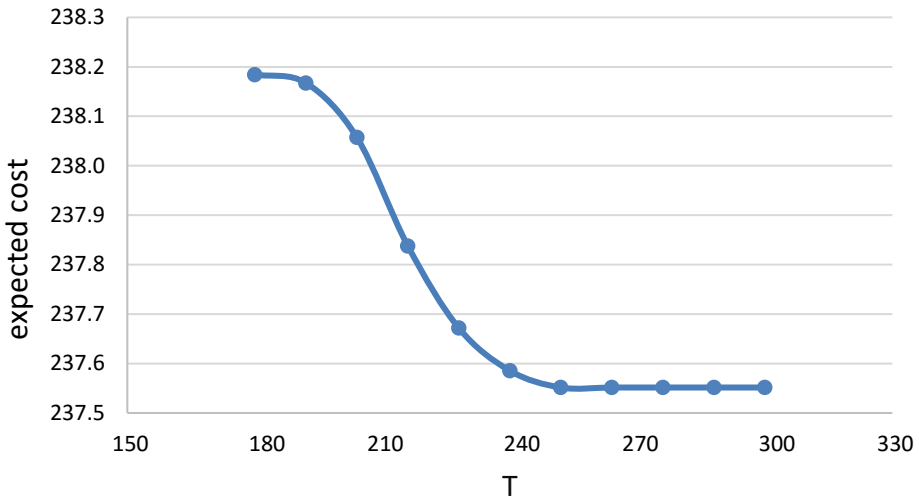


Figure 5.4: Effect of target mean $T$ on expected cost

Finally, we evaluated the impact of removing the mean or risk component from the objective function on solution policies. Figures 5.5 and 5.6 compare the solution policy for the risk-neutral model to solution policy for the models where the objective function only considers risk, i.e. the risk term has a very large weight. It is evident from the figures that considering a risk-only model significantly impacts the solution. The risk-only policy focuses on checking more elements later and more often focusing on reducing the chance the error reaches the patient. While the mean-only policy checks early and seems to consider rework cost more than the risk-only policies. The risk-averse solution for the central semi-deviation case, results in a 35% increase in expected cost (from 234.79 to 382.52) while there is an 18% reduction for the central semi-deviation (from 10.18 to 8.28) when compared to the risk-neutral solution ($\lambda = 0$). In the case with the semi-deviation from a target risk with $T = 240$, the risk-averse solution results in a 7% increase in expected cost (from 234.79 to 253.18) compared to a 75% reduction for the semi-deviation from the target (from 0.04 to 0.01) when compared to the risk-neutral solution. However, it is important to explore how the percentages may change for lower or higher values of $T$.

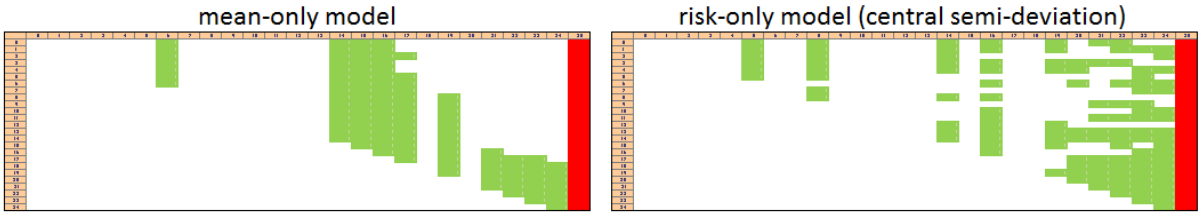


Figure 5.5: Comparing the risk neutral model to the central semi-deviation model

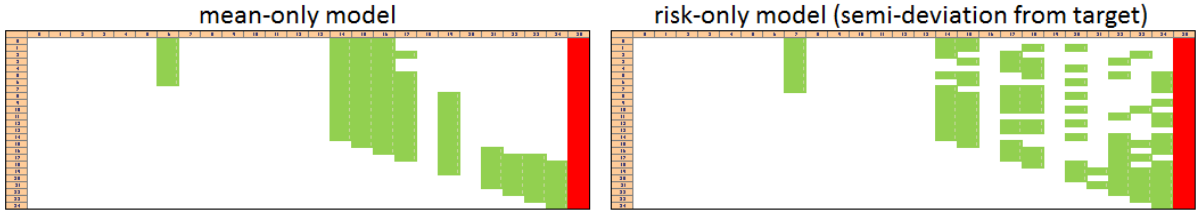


Figure 5.6: Comparing the risk neutral model to the semi-deviation from a target model

## 5.5 Conclusion and Future Work

In this chapter we presented a two-stage stochastic programming model to address uncertainty in reliable radiation therapy problem. Our proposed framework includes expected cost and a risk measure to characterize different risk preferences. Given the continuous distribution of uncertain parameters, it is impossible to exactly evaluate the objective function of the stochastic model. Therefore we used the sample average approximation method to solve the stochastic model and obtained statistical lower and upper bounds on true optimum. We proposed two modifications to avoid the negative gap estimates in the SAA approach. In addition, we considered two risk criteria and tested the effect of changing risk coefficient in the mean-risk function. The numerical results indicate that different risk measures and risk parameters can significantly impact the gap estimate and its variance as well as the computational effort. However, the policy is relatively robust to risk.

One potential extension to our model would be to consider other risk measures such as measures that involve worst case scenarios. Another possible future work would be to develop efficient algorithms for solving the risk-mean stochastic model with larger number of scenarios. Furthermore, it would be interesting to investigate a two-stage stochastic programming approach with second stage decision variables to allow adjustments in the design of SBs.

# Chapter 6

# Conclusion and Future Work

About half of all cancer patients receive radiation therapy during the course of their treatment [53]. The goal is to deliver the proper dose of radiation to tumor while minimizing the exposure to nearby healthy tissues. Given the risks associated with any radiation therapy method, it is critical that the treatment is planned and delivered safely and accurately. Despite the use of quality assurance procedures in radiation oncology clinics, errors still occur and sometimes result in devastating consequences including fatal injuries. Statistics show that RT errors occur for up to 5% of patients and the rate of lethal errors is about 1/10,000 patients in the U.S. [80, 21]. These numbers highlight the need for improving patient safety in radiation therapy.

Safety barriers (SB) are used throughout the RT process to detect potential process deviations and errors that may affect patients. An effective combination of safety barriers prevents errors from propagating and eventually reaching patients. Despite the critical role of SBs in providing patients with a safe treatment, very little research effort has been focused on potential costs, impact or optimal implementation of SBs. Adding a SB after each step in the patient care pathway could significantly improve the process reliability and patient safety. However each SB increases the cost and duration of treatment and causes extra workload that may result in additional errors. In this dissertation, I developed mathematical models to determine the optimal design of SBs in terms of patient safety while considering the implementation costs. This is a challenging problem mainly because (i) the objectives of improving patient safety and minimizing SB implementation costs are conflicting objectives and (ii) the RT process is a highly complex process that involves multiple steps, constraints and interdependences that

need to be addressed. Our proposed framework captures the complexities of RT process and inform the design of SBs to assure process reliability and patient safety.

In Chapter 3, we used computer simulation to assess the reliability of the RT process. We used actual data from the error reporting system at the Department of Radiation Oncology of University of North Carolina (UNC) Lineberger Cancer Center to estimate the error rates at different steps in radiotherapy pathway. The simulation model proved to be an effective tool to evaluate the reliability of the process under different scenarios and identify potential enhancements. We investigated the effect of SB design modifications such as adding more SBs or changing their composition. Our analysis highlighted the trade-off between process reliability and the effort required for performing SBs. Furthermore, the simulation model provided valuable insights on how error-detection can be improved through changes in the SB design.

In Chapter 4, we developed deterministic mathematical models to find the optimal design of SBs. Our modeling approach is motivated by the reliable facility location problem (RFLP). However there are still key differences between our problem and RFLP and these differences introduce additional challenges in modeling and solving the problem. We formulated the problem as a mixed integer program which aims to find the optimal position and composition of a given number of SBs to minimize checking and rework costs. We proposed an efficient solution algorithm. We presented results using data from the UNC Department of Radiation Oncology. Analyzing different parameters of the model, we found that SB failure probabilities are the most influential on model results and the performance of the proposed algorithm. Finally, we compared our proposed SB design to the current configuration of SBs in the UNC RT process. The comparison revealed that although our proposed design involves more checks, it results in a significantly lower expected cost.

In Chapter 5, we formulated a two-stage stochastic programming model to address uncertainties in the reliable SB design problem. We applied a sample average approximation (SAA) as a solution method and obtained statistical lower and upper bound estimates for the true optimal objective value. To avoid negative SAA gaps, we proposed two modifications and demonstrated how they affect the SAA results. Finally, we considered a mean-risk objective to incorporate different risk preferences in the stochastic model. We used two different risk measures and evaluated their impact on total cost and solution policies. Our findings indicate that incorporating

a risk measure has limited impact on solution policy.

## 6.1 Future Work

Future research can be conducted in a few directions. The main limitation of the proposed models is the assumption that SBs are uncapacitated meaning that there is no limit on the number of elements they could check. This assumption may not hold in practice. Future research should consider a capacity limit for each SB.

Another assumption that might not hold in some situations is that SB failure probability for a given element only depends on the workload of that individual element and is not affected by the workload of other elements checked in the same SB. However, when the overall workload of a SB high, it is more likely to fail, even if the individual elements have low workload. It would be an interesting future direction to incorporate the effect of overall workload on SB failure probabilities for individual elements.

In addition, future work might explore other risk-aversion criteria in the two-stage stochastic programming model. Examples include criteria that involve reaching a certain level of reliability or minimizing the cost in the worst case scenario. Another potential extension to our stochastic programming model is to incorporate the second-stage decision variables that allow adjustments in SB design after the realization of uncertain parameters. In the current model, there is no remedy for the first-stage decisions regarding the position of SBs and the elements assigned to them. In reality, it might be necessary to make modifications to the design of SBs based of different realizations of error probabilities, SB failure probabilities or rework costs.

Although the focus of this dissertation is on improving patient safety in RT process, the findings have implications beyond radiation oncology. The proposed framework can be applied to inform the design of SBs and in general QA systems in any field where the poorly designed processes raise safety concerns.

# REFERENCES


[1] Robert Aboolian, Tingting Cui, and Zuo-Jun Max Shen. An efficient approach for solving reliable facility location models. *INFORMS Journal on Computing*, 25(4):720–729, 2012.

[2] Sajjad Ahmad and John Billimek. Estimating the health impacts of tobacco harm reduction policies: a simulation modeling approach. *Risk Analysis*, 25(4):801–812, 2005.

[3] Sajjad Ahmad and John Billimek. Limiting youth access to tobacco: Comparing the long-term health impacts of increasing cigarette excise taxes and raising the legal smoking age to 21 in the united states. *Health Policy*, 80(3):378–391, 2007.

[4] Shabbir Ahmed. Convexity and decomposition of mean-risk stochastic programs. *Mathematical Programming*, 106(3):433–446, 2006.

[5] Shabbir Ahmed, Alexander Shapiro, and Er Shapiro. The sample average approximation method for stochastic programs with integer recourse. *Submitted for publication*, pages 1–24, 2002.

[6] American Society of Radiation Oncology. ASTRO participates in FDA meeting on radiation safety. http://www.eurekalert.org/pub$_r eleases/2010 - 06/asfr - api061610.php. Accessed : 2016 - 06 - 15$.

[7] Yu An, Bo Zeng, Yu Zhang, and Long Zhao. Reliable p-median facility location problem: two-stage robust models and algorithms. *Transportation Research Part B: Methodological*, 64:54–72, 2014.

[8] B Antonijevic, Christophe Matthys, Isabelle Sioen, Maaike Bilau, John Van Camp, JL Willems, and Stefaan De Henauw. Simulated impact of a fish based shift in the population n- 3 fatty acids intake on exposure to dioxins and dioxin-like compounds. *Food and Chemical Toxicology*, 45(11):2279–2286, 2007.

[9] Carlos A Arboleda, Dulcy M Abraham, and Robert Lubitz. Simulation as a tool to assess the vulnerability of the operation of a health care facility. *Journal of performance of constructed facilities*, 21(4):302–312, 2007.

[10] Anthony Arnold, Geoff P Delaney, Lynette Cassapi, and Michael Barton. The use of categorized time-trend reporting of radiation oncology incidents: a proactive analytical approach to improving quality and safety over time. *International Journal of Radiation Oncology* Biology* Physics*, 78(5):1548–1554, 2010.

[11] Rifat A Atun, Reda M Lebcir, Martin McKee, Jarno Habicht, and Richard J Coker. Impact of joined-up hiv harm reduction and multidrug resistant tuberculosis control programmes in estonia: System dynamics simulation model. *Health Policy*, 81(2):207–217, 2007.

[12] Nezir Aydin and Alper Murat. A swarm intelligence based sample average approximation algorithm for the capacitated reliable facility location problem. *International Journal of Production Economics*, 145(1):173–183, 2013.

[13] Paul Barach, SD Small, and H Kaplan. Designing a confidential safety reporting system: in depth review of thirty major medical incident reporting systems, and near-miss safety reporting systems in the nuclear, aviation, and petrochemical industries. *Anesthesiology*, 91:A1209, 1999.

[14] Paul Barach and Stephen D Small. Reporting and preventing medical mishaps: lessons from non-medical near miss reporting systems. *British Medical Journal*, 320(7237):759, 2000.

[15] Joseph Barjis, Alexander Verbraeck, Sung J Shim, and Arun Kumar. Simulation for emergency care process reengineering in hospitals. *Business Process Management Journal*, 16(5):795–805, 2010.

[16] Debbie Barnard, Marilyn Dumkee, Balvir Bains, and Brenda Gallivan. Implementing a good catch program in an integrated health system. *Healthc Q*, 9:22–27, 2006.

[17] O Berman, T Drezner, Z Drezner, and GO Wesolowsky. A defensive maximal covering problem on a network. *International Transactions in Operational Research*, 16(1):69–86, 2009.

[18] Oded Berman, Dmitry Krass, and Mozart BC Menezes. Facility reliability issues in network p-median problems: strategic centralization and co-location effects. *Operations Research*, 55(2):332–350, 2007.

[19] Allyson M Best, Cinnamon A Dixon, W David Kelton, Christopher J Lindsell, and Michael J Ward. Using discrete event computer simulation to improve patient flow in a ghanaian acute care hospital. *The American journal of emergency medicine*, 32(8):917–922, 2014.

[20] Jean-Pierre Bissonnette and Gaylene Medlam. Trend analysis of radiation therapy incidents over seven years. *Radiotherapy and Oncology*, 96(1):139–144, 2010.

[21] Walt Bogdanich. Radiation offers new cures, and ways to do harm. *The New York Times*, 23, 2010.

[22] James P Borgstede and Marie D Zinninger. Radiology and patient safety 1. *Academic radiology*, 11(3):322–332, 2004.

[23] Thomas Bortfeld, Timothy CY Chan, Alexei Trofimov, and John N Tsitsiklis. Robust management of motion uncertainty in intensity-modulated radiation therapy. *Operations Research*, 56(6):1461–1473, 2008.

[24] Sally C Brailsford, VA Lattimer, P Tarnaras, and JC Turnbull. Emergency and on-demand health care: modelling a large complex system. *Journal of the Operational Research Society*, 55(1):34–42, 2004.

[25] Riccardo Calandrino, Giovanni Mauro Cattaneo, Claudio Fiorino, Barbara Longobardi, Paola Mangili, and Patrizia Signorotto. Detection of systematic errors in external radiotherapy before treatment delivery. *Radiotherapy and Oncology*, 45(3):271–274, 1997.

[26] Centers for Disease Control and Prevention. Number of deaths for leading causes of death. http://www.cdc.gov/nchs/fastats/leading-causes-of-death.htm. Accessed: 2016-12-07.

[27] Timothy CY Chan, Thomas Bortfeld, and John N Tsitsiklis. A robust approach to imrt optimization. *Physics in medicine and biology*, 51(10):2567, 2006.

[28] Timothy CY Chan, Houra Mahmoudzadeh, and Thomas G Purdie. A robust-cvar optimization approach with application to breast cancer therapy. *European Journal of Operational Research*, 238(3):876–885, 2014.

[29] Tingting Cui, Yanfeng Ouyang, and Zuo-Jun Max Shen. Reliable facility location design under the risk of disruptions. *Operations research*, 58(4-part-1):998–1011, 2010.

[30] Mark S Daskin. *Network and discrete location: models, algorithms, and applications*. John Wiley & Sons, 2011.

[31] Djohan Djohan, Jimmy Yu, Des Connell, and Elizabeth Christensen. Health risk assessment of chlorobenzenes in the air of residential houses using probabilistic techniques. *Journal of Toxicology and Environmental Health, Part A*, 70(19):1594–1603, 2007.

[32] Lane F Donnelly, Julie M Dickerson, Martha A Goodfriend, and Stephen E Muething. Improving patient safety in radiology: concepts for a comprehensive patient safety program. In *Seminars in Ultrasound, CT and MRI*, volume 31, pages 67–70. Elsevier, 2010.

[33] Neil D'Souza, Lori Holden, Sheila Robson, Kathy Mah, Lisa Di Prospero, C Shun Wong, Edward Chow, and Jacqueline Spayne. Modern palliative radiation treatment: do complexity and workload contribute to medical errors? *International Journal of Radiation Oncology* Biology* Physics*, 84(1):e43–e48, 2012.

[34] MA Ebert, W Li, L Jennings, R Kearvell, and S Bydder. Utilitarian prioritization of radiation oncology patients based on maximization of population tumour control. *Physics in medicine and biology*, 58(12):4013, 2013.

[35] Jason M Etchegaray, Eric J Thomas, Jane M Geraci, Debora Simmons, and Sharon K Martin. Differentiating close calls from errors: A multidisciplinary perspective. *Journal of Patient Safety*, 1(3):133–137, 2005.

[36] Alexander Finkelshtein, Yale T Herer, Tzvi Raz, and Irad Ben-Gal. Economic optimization of off-line inspection in a process subject to failure and recovery. *IIE Transactions*, 37(11):995–1009, 2005.

[37] EC Ford, L Fong de Los Santos, T Pawlicki, S Sutlief, and P Dunscombe. Consensus recommendations for incident learning database structures in radiation oncology. *Medical physics*, 39(12):7272–7290, 2012.

[38] Eric C Ford and Stephanie Terezakis. How safe is safe? risk in radiotherapy. *International Journal of Radiation Oncology* Biology* Physics*, 78(2):321–322, 2010.

[39] Eric C Ford, Stephanie Terezakis, Annette Souranis, Kendra Harris, Hiram Gay, and Sasa Mutic. Quality control quantification (qcq): A tool to measure the value of quality control checks in radiation oncology. *International Journal of Radiation Oncology* Biology* Physics*, 84(3):e263–e269, 2012.

[40] Benedick A Fraass. Errors in radiotherapy: motivation for development of new radiotherapy quality assurance paradigms. *International Journal of Radiation Oncology* Biology* Physics*, 71(1):S162–S165, 2008.

[41] Benedick A Fraass, Kathy L Lash, Gwynne M Matrone, Susan K Volkman, Daniel L Mcshan, Marc L Kessler, and Allen S Lichter. The impact of treatment complexity and computer-control delivery technology on treatment delivery errors. *International Journal of Radiation Oncology* Biology* Physics*, 42(3):651–659, 1998.

[42] D Gade and EA Pohl. Sample average approximation applied to the capacitated-facilities location problem with unreliable facilities. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 223(4):259–269, 2009.

[43] Alberto Garcia-diaz, Joseph W Foster, and Manuel Bonyuet. Dynamic programming analysis of special multi-stage inspection systems. *IIE Transactions*, 16(2):115–126, 1984.

[44] Jerry H Gurwitz, Terry S Field, Jerry Avorn, Danny McCormick, Shailavi Jain, Marie Eckler, Marcia Benser, Amy C Edmondson, and David W Bates. Incidence and preventability of adverse drug events in nursing homes. *The American journal of medicine*, 109(2):87–94, 2000.

[45] Bernard J Healey and Michele McGowan. The enormous cost of medical errors. *Academy of Health Care Management Journal*, 6(1), 2010.

[46] Alejandro Heredia-Langner, Douglas C Montgomery, and WM Carlyle. Solving a multistage partial inspection problem using genetic algorithms. *International journal of production research*, 40(8):1923–1940, 2002.

[47] Ola Holmberg and Brendan McClean. Preventing treatment errors in radiotherapy by identifying and evaluating near misses and actual incidents. *Journal of Radiotherapy in Practice*, 3(01):13–25, 2002.

[48] Grace Huang, Gaylene Medlam, Justin Lee, Susan Billingsley, Jean-Pierre Bissonnette, Jolie Ringash, Gabrielle Kane, and David C Hodgson. Error in the delivery of radiation therapy: results of a quality assurance review. *International Journal of Radiation Oncology* Biology* Physics*, 61(5):1590–1595, 2005.

[49] Armann Ingolfsson, Erhan Erkut, and Susan Budge. Simulation of single start station for edmonton ems. *Journal of the Operational Research Society*, 54(7):736–746, 2003.

[50] Robert R Inman, Dennis E Blumenfeld, Ningjian Huang, Jingshan Li, and Jing Li. Survey of recent advances on the interface between production system design and quality. *IIE Transactions*, 45(6):557–574, 2013.

[51] Brian R Jackson, Alun Thomas, Karen C Carroll, Frederick R Adler, and Matthew H Samore. Use of strain typing data to estimate bacterial transmission rates in healthcare settings. *Infection Control*, 26(07):638–645, 2005.

[52] Sajjad Jalali, Mehdi Seifbarghy, Javad Sadeghi, and Samad Ahmadi. Optimizing a bi-objective reliable facility location problem with adapted stochastic measures using tuned-parameter multi-objective algorithms. *Knowledge-Based Systems*, 2015.

[53] Chandra Prakash Joshi. Patient safety in an environment of rapidly advancing technology in radiation therapy. *Journal of medical physics/Association of Medical Physicists of India*, 39(2):61, 2014.

[54] Vivek Kakade, Jorge F Valenzuela*, and Jeffrey S Smith. An optimization model for selective inspection in serial manufacturing systems. *International journal of production research*, 42(18):3891–3909, 2004.

[55] John A Kalapurakal, Aleksandar Zafirovski, Jeffery Smith, Paul Fisher, Vythialingam Sathiaseelan, Cynthia Barnard, Alfred W Rademaker, Nick Rave, and Bharat B Mittal. A comprehensive quality assurance program for personnel and procedures in radiation oncology: Value of voluntary error reporting and checklists. *International Journal of Radiation Oncology* Biology* Physics*, 86(2):241–248, 2013.

[56] T Kapamara, K Sheibani, D Petrovic, O Haas, and C Reeves. A simulation of a radiotherapy treatment system: A case study of a local cancer centre. In *ORP3 meeting*, 2007.

[57] HS Kaplan and B Rabin Fastman. Organization of event reporting data for sense making and system improvement. *Quality and Safety in Health Care*, 12(suppl 2):ii68–ii72, 2003.

[58] Mikael Karlsson, Anders Ahnesjö, Dietmar Georg, and Tufve Nyholm. Independent dose calculations, concepts and models. 2010.

[59] Korina Katsaliaki and Navonil Mustafee. Applications of simulation within the healthcare context. *Journal of the Operational Research Society*, 62(8):1431–1451, 2011.

[60] Marieke Kessels-Habraken, Tjerk Van der Schaaf, Jan De Jonge, and Christel Rutte. Defining near misses: towards a sharpened definition based on empirical data about error handling processes. *Social science & medicine*, 70(9):1301–1308, 2010.

[61] Sujin Kim, Raghu Pasupathy, and Shane G Henderson. A guide to sample average approximation. In *Handbook of simulation optimization*, pages 207–243. Springer, 2015.

[62] Immo Kleinschmidt, Arthi Ramkissoon, Natashia Morris, Zonke Mabude, Bronwyn Curtis, and Mags Beksinska. Mapping indicators of sexually transmitted infection services in the south african public health sector. *Tropical Medicine & International Health*, 11(7):1047–1057, 2006.

[63] Anton J Kleywegt, Alexander Shapiro, and Tito Homem-de Mello. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization*, 12(2):479–502, 2002.

[64] Tommy Knöös. Qa procedures needed for advanced rt techniques and its impact on treatment outcome. In *Journal of Physics: Conference Series*, volume 573, page 012001. IOP Publishing, 2015.

[65] Linda T Kohn, Janet M Corrigan, Molla S Donaldson, et al. *To Err Is Human:: Building a Safer Health System*. National Academies Press, 2000.

[66] Gerald J Kutcher, Lawrence Coia, Michael Gillin, William F Hanson, Steven Leibel, Robert J Morton, Jatinder R Palta, James A Purdy, Lawrence E Reinstein, Goran K Svensson, et al. Comprehensive qa for radiation oncology: report of aapm radiation therapy committee task group 40. *MEDICAL PHYSICS-LANCASTER PA-*, 21:581–581, 1994.

[67] Dora LW Kwong, Jonathan ST Sham, Daniel TT Chua, Damon TK Choy, Gordon KH Au, and PMl Wu. The effect of interruptions and prolonged treatment time in radiotherapy for nasopharyngeal carcinoma. *International Journal of Radiation Oncology* Biology* Physics*, 39(3):703–710, 1997.

[68] David C Lane, Camilla Monefeldt, and JV Rosenhead. Looking in the wrong place for healthcare improvements: A system dynamics study of an accident and emergency department. *Journal of the operational Research Society*, pages 518–531, 2000.

[69] Thomas H Lee and James J Mongan. *Chaos and organization in health care*. MIT Press Cambridge, MA, 2009.

[70] Qingwei Li, Bo Zeng, and Alex Savachkin. Reliable facility location design under disruptions. *Computers & Operations Research*, 40(4):901–909, 2013.

[71] Xiaopeng Li and Yanfeng Ouyang. A continuum approximation approach to reliable facility location design under correlated probabilistic disruptions. *Transportation research part B: methodological*, 44(4):535–548, 2010.

[72] Michael Lim, Mark S Daskin, Achal Bassamboo, and Sunil Chopra. A facility reliability problem: formulation, properties, and algorithm. *Naval Research Logistics (NRL)*, 57(1):58–70, 2010.

[73] Jeff Linderoth, Alexander Shapiro, and Stephen Wright. The empirical behavior of sampling methods for stochastic programming. *Annals of Operations Research*, 142(1):215–241, 2006.

[74] G Lonati, S Cernuschi, M Giugliano, and M Grosso. Health risk analysis of pcdd/f emissions from msw incineration: comparison of probabilistic and deterministic approaches. *Chemosphere*, 67(9):S334–S343, 2007.

[75] Mengshi Lu, Lun Ran, and Zuo-Jun Max Shen. Reliable facility location design under uncertain correlated disruptions. *Manufacturing & Service Operations Management*, 17(4):445–455, 2015.

[76] Wai-Kei Mak, David P Morton, and R Kevin Wood. Monte carlo bounding techniques for determining solution quality in stochastic programs. *Operations Research Letters*, 24(1):47–56, 1999.

[77] Martin A Makary and Michael Daniel. Medical errorthe third leading cause of death in the us. *BMJ*, 353:i2139, 2016.

[78] Sampatraj S Mandroli, Abhishek K Shrivastava, and Yu Ding. A survey of inspection strategy and sensor distribution studies in discrete-part manufacturing processes. *IIE Transactions*, 38(4):309–328, 2006.

[79] Haider R Mannan, Matthew Knuiman, and Michael Hobbs. A markov simulation model for analyzing and forecasting the number of coronary artery revascularization procedures in western australia. *Annals of epidemiology*, 17(12):964–975, 2007.

[80] Lawrence B Marks, Marianne Jackson, Liyi Xie, Sha X Chang, Katharin Deschesne Burkhardt, Lukasz Mazur, Ellen L Jones, Patricia Saponaro, Dana LaChapelle, Dee C Baynes, et al. The challenge of maximizing safety in radiation oncology. *Practical radiation oncology*, 1(1):2–14, 2011.

[81] Lawrence B Marks, Kim L Light, Jessica L Hubbs, Debra L Georgas, Ellen L Jones, Melanie C Wright, Christopher G Willett, and Fang Fang Yin. The impact of advanced

technologies on treatment deviations in radiation treatment delivery. *International Journal of Radiation Oncology* Biology* Physics*, 69(5):1579–1586, 2007.

[82] Lukasz M Mazur, Prithima R Mosaly, Lesley M Hoyle, Ellen L Jones, Bhishamjit S Chera, and Lawrence B Marks. Relating physicians workload with errors during radiation therapy planning. *Practical radiation oncology*, 4(2):71–75, 2014.

[83] Lukasz M Mazur, Prithima R Mosaly, Marianne Jackson, Sha X Chang, Katharin Deschesne Burkhardt, Robert D Adams, Ellen L Jones, Lesley Hoyle, Jing Xu, John Rockwell, et al. Quantitative assessment of workload and stressors in clinical radiation oncology. *International Journal of Radiation Oncology* Biology* Physics*, 83(5):e571–e576, 2012.

[84] Douglas C Montgomery. *Statistical quality control*. John Wiley, 1985.

[85] Lorenzo Moreno, Rosa María Aguilar, CA Martin, José D Piñeiro, JI Estevez, José F Sigut, and José L Sánchez. Patient-centered simulation to aid decision-making in hospital management. *Simulation*, 74(5):290–304, 2000.

[86] AJ Munro and S Potter. Waiting times for radiotherapy treatment: not all that mysterious and certainly preventable. *Clinical Oncology*, 6(5):314–318, 1994.

[87] Navonil Mustafee, Korina Katsaliaki, and Simon JE Taylor. Profiling literature in healthcare simulation. *Simulation*, 2010.

[88] Y Narahari and LM Khan. Modeling reentrant manufacturing systems with inspection stations. *Journal of Manufacturing Systems*, 15(6):367–378, 1996.

[89] Vladimir I Norkin, Georg Ch Pflug, and Andrzej Ruszczyński. A branch and bound method for stochastic global optimization. *Mathematical programming*, 83(1-3):425–450, 1998.

[90] Nilay Noyan. Risk-averse two-stage stochastic programming with an application to disaster management. *Computers & Operations Research*, 39(3):541–559, 2012.

[91] S Noyan Ogulata, M Oya Cetik, Esra Koyuncu, and Melik Koyuncu. A simulation approach for scheduling patients in the department of radiation oncology. *Journal of medical systems*, 33(3):233–239, 2009.

[92] Jesse R OHanley and Richard L Church. Designing robust coverage networks to hedge against worst-case facility losses. *European Journal of Operational Research*, 209(1):23–36, 2011.

[93] Eduardo Ortiz, Gregg Meyer, and Helen Burstin. Clinical informatics and patient safety at the agency for healthcare research and quality. *Journal of the American Medical Informatics Association*, 9(Supplement 6):S2–S7, 2002.

[94] Peng Peng, Lawrence V Snyder, Andrew Lim, and Zuli Liu. Reliable logistics networks design with facility disruptions. *Transportation Research Part B: Methodological*, 45(8):1190–1211, 2011.

[95] Michal Penn and Tal Raviv. A polynomial time algorithm for solving a quality control station configuration problem. *Discrete Applied Mathematics*, 156(4):412–419, 2008.

[96] R Burr Porter. Semivariance and stochastic dominance: a comparison. *The American Economic Review*, 64(1):200–204, 1974.

[97] Peter Mark Pruzan and JT Ross Jackson. A dynamic programming application in production line inspection. *Technometrics*, 9(1):73–81, 1967.

[98] Lian Qi and Zuo-Jun Max Shen. A supply chain design model with unreliable supply. *Naval Research Logistics (NRL)*, 54(8):829–844, 2007.

[99] Gad Rabinowitz and Hamilton Emmons. Optimal and heuristic inspection schedules for multistage production systems. *IIE transactions*, 29(12):1063–1071, 1997.

[100] H Rau and Y-H Chu. Inspection allocation planning with two types of workstation: Wvd and wad. *The International Journal of Advanced Manufacturing Technology*, 25(9-10):947–953, 2005.

[101] James Reason. Human error: models and management. *Bmj*, 320(7237):768–770, 2000.

[102] JT Reason, J Carthey, and MR De Leval. Diagnosing vulnerable system syndrome: an essential prerequisite to effective risk management. *Quality in health care*, 10(suppl 2):ii21–ii25, 2001.

[103] Ranjit Rebello, Alessandro Agnetis, and Pitu B Mirchandani. Specialized inspection problems in serial production systems. *European Journal of Operational Research*, 80(2):277–296, 1995.

[104] Andrzej Ruszczyński and Alexander Shapiro. Stochastic programming models. *Handbooks in operations research and management science*, 10:1–64, 2003.

[105] Tjendera Santoso, Shabbir Ahmed, Marc Goetschalckx, and Alexander Shapiro. A stochastic programming approach for supply chain network design under uncertainty. *European Journal of Operational Research*, 167(1):96–115, 2005.

[106] José A Sepúlveda, William J Thompson, Felipe F Baesler, María I Alvarez, and Lonnie E Cahoon III. The use of simulation for process improvement in a cancer treatment center. In *Proceedings of the 31st conference on Winter simulation: Simulation—a bridge to the future-Volume 2*, pages 1541–1548. ACM, 1999.

[107] Tait D Shanafelt, Lotte N Dyrbye, Christine Sinsky, Omar Hasan, Daniel Satele, Jeff Sloan, and Colin P West. Relationship between clerical burden and characteristics of the electronic environment with physician burnout and professional satisfaction. In *Mayo Clinic Proceedings*, volume 91, pages 836–848. Elsevier, 2016.

[108] Steven M Shechter, Cindy L Bryce, Oguzhan Alagoz, Jennifer E Kreke, James E Stahl, Andrew J Schaefer, Derek C Angus, and Mark S Roberts. A clinically based discrete-event simulation of end-stage liver disease and the organ allocation process. *Medical Decision Making*, 25(2):199–209, 2005.

[109] Zuo-Jun Max Shen, Roger Lezhou Zhan, and Jiawei Zhang. The reliable facility location problem: Formulations, heuristics, and approximation algorithms. *INFORMS Journal on Computing*, 23(3):470–482, 2011.

[110] David M Shepard, Michael C Ferris, Gustavo H Olivera, and T Rockwell Mackie. Optimizing the delivery of radiation therapy to cancer patients. *Siam Review*, 41(4):721–744, 1999.

[111] Ali G Shetwan, Valentin I Vitanov, and Benny Tjahjono. Allocation of quality control stations in multistage manufacturing systems. *Computers & Industrial Engineering*, 60(4):473–484, 2011.

[112] Y-R Shiau. Inspection allocation planning for a multiple quality characteristic advanced manufacturing system. *The International Journal of Advanced Manufacturing Technology*, 21(7):494–500, 2003.

[113] Yau-Ren Shiau. Inspection resource assignment in a multistage manufacturing system with an inspection error model. *International Journal of Production Research*, 40(8):1787–1806, 2002.

[114] Yau-Ren Shiau, Meng-Hung Lin, and Wen-Chieh Chuang. Concurrent process/inspection planning for a customized manufacturing system based on genetic algorithm. *The International Journal of Advanced Manufacturing Technology*, 33(7-8):746–755, 2007.

[115] Nagesh Shukla, Dariusz Ceglarek, and Manoj K Tiwari. Key characteristics-based sensor distribution in multi-station assembly processes. *Journal of intelligent manufacturing*, 26(1):43–58, 2015.

[116] Lawrence V Snyder and Mark S Daskin. Reliability models for facility location: the expected failure cost case. *Transportation Science*, 39(3):400–416, 2005.

[117] Society of Nuclear Medicine and Molecular Imaging. Consistency, accuracy, responsibility, and excellence in medical imaging and radiation therapy (CARE) act, 2016. http://interactive.snm.org/index.cfm?PageID=9705. Accessed: 2016-06-15.

[118] James R Swisher, Sheldon H Jacobson, J Brian Jun, and Osman Balci. Modeling and analyzing a physician clinic environment using discrete-event (visual) simulation. *Computers & operations research*, 28(2):105–125, 2001.

[119] Samer Takriti and Shabbir Ahmed. On robust optimization of two-stage systems. *Mathematical Programming*, 99(1):109–126, 2004.

[120] Michal Tamuz and Michael I Harrison. Improving patient safety in hospitals: contributions of high-reliability theory and normal accident theory. *Health services research*, 41(4p2):1654–1676, 2006.

[121] SJ Thomas. Capacity and demand models for radiotherapy treatment machines. *Clinical Oncology*, 15(6):353–358, 2003.

[122] Sofie Van Volsem, Wout Dullaert, and Hendrik Van Landeghem. An evolutionary algorithm and discrete event simulation for optimizing inspection strategies for multi-stage processes. *European Journal of Operational Research*, 179(3):621–633, 2007.

[123] Michael H Veatch. Inspection strategies for multistage production systems with time-varying quality. *International Journal of Production Research*, 38(4):837–853, 2000.

[124] Bram Verweij, Shabbir Ahmed, Anton J Kleywegt, George Nemhauser, and Alexander Shapiro. The sample average approximation method applied to stochastic routing problems: a computational study. *Computational Optimization and Applications*, 24(2-3):289–333, 2003.

[125] Junwen Wang, Jingshan Li, Anne T Campbell, Kim Prather, Barbara Johnson, and Kay Ross. Analysis of patient flow in radiation oncology department of central baptist hospital using discrete event simulations. In *IIE Annual Conference. Proceedings*, page 1. Institute of Industrial Engineers-Publisher, 2012.

[126] Greg Werker, Antoine Sauré, John French, and Steven Shechter. The use of discrete-event simulation modelling to improve radiation therapy planning processes. *Radiotherapy and Oncology*, 92(1):76–82, 2009.

[127] World Health Organization. Radiotherapy risk prole, 2008. www.who.int/patientsafety/activities/technical/radiotherapy risk prole.pdf. Accessed: 2016-06-15.

[128] Tai Keung Yeung, Karen Bortolotto, Scott Cosby, Margaret Hoar, and Ernst Lederer. Quality assurance in radiotherapy: evaluation of errors and incidents recorded over a 10 year period. *Radiotherapy and Oncology*, 74(3):283–291, 2005.

[129] Shaohui Zheng. Dynamic quality control in assembly systems. *IIE transactions*, 32(9):797–806, 2000.

# APPENDIX

# Appendix A

# Computational Results

## A.1 Effect of increasing error probability and checking cost on SnC algorithm performance

Table A.1: SnC algorithm performance - increasing the error probabilities by 25%

| n | b | Gurobi gap | Gurobi time (sec) | Snc gap | SnC time (sec) |
|---|---|---|---|---|---|
| 25 | 5 | 7% | 3600 | $\leq 0.5\%$ | 12 |
| 25 | 10 | 12% | 3600 | $\leq 0.5\%$ | 351 |
| 25 | 15 | 10% | 3600 | $\leq 0.5\%$ | 102 |
| 25 | 20 | 8% | 3600 | $\leq 0.5\%$ | 18 |
| 50 | 5 | 47% | 3600 | 5.41% | 3600 |
| 50 | 10 | 29% | 3600 | 8.32% | 3600 |
| 50 | 15 | 27% | 3600 | 4.90% | 3600 |
| 50 | 20 | 24% | 3600 | 3.25% | 3600 |
| 50 | 25 | 23% | 3600 | 1.56% | 3600 |
| 50 | 30 | 22% | 3600 | 0.88% | 3600 |
| 50 | 40 | 21% | 3600 | $\leq 0.5\%$ | 371 |

Table A.2: SnC algorithm performance - increasing the checking costs by 25%

| n | b | Gurobi gap | Gurobi time (sec) | Snc gap | SnC time (sec) |
|---|---|---|---|---|---|
| 25 | 5 | 11% | 3600 | $\leq 0.5\%$ | 15 |
| 25 | 10 | 14% | 3600 | $\leq 0.5\%$ | 1503 |
| 25 | 15 | 13% | 3600 | $\leq 0.5\%$ | 1767 |
| 25 | 20 | 10% | 3600 | $\leq 0.5\%$ | 405 |
| 50 | 5 | 47% | 3600 | 8.34% | 3600 |
| 50 | 10 | 31% | 3600 | 9.91% | 3600 |
| 50 | 15 | 28% | 3600 | 6.05% | 3600 |
| 50 | 20 | 28% | 3600 | 4.19% | 3600 |
| 50 | 25 | 25% | 3600 | 2.67% | 3600 |
| 50 | 30 | 25% | 3600 | 1.78% | 3600 |
| 50 | 40 | 24% | 3600 | $\leq 0.5\%$ | 390 |

## A.2 Effect of increasing error probabilities and checking cost on objective function value and optimal policy

Table A.3: Objective function value - cost elements

Base case

| n | b | total cost (min) | rework cost (min) | checking cost (min) | total # of checks |
|---|---|---|---|---|---|
| 25 | 5 | 190 | 135 | 55 | 92 |
| 25 | 10 | 148 | 94 | 54 | 91 |
| 25 | 15 | 138 | 84 | 54 | 91 |
| 25 | 20 | 135 | 82 | 53 | 90 |

Increasing error probabilities by 25%

| n | b | total cost (min) | rework cost (min) | checking cost (min) | total # of checks |
|---|---|---|---|---|---|
| 25 | 5 | 224 | 169 | 55 | 93 |
| 25 | 10 | 172 | 117 | 55 | 92 |
| 25 | 15 | 159 | 104 | 55 | 93 |
| 25 | 20 | 156 | 101 | 55 | 92 |

Increasing checking costs by 25%

| n | b | total cost (min) | rework cost (min) | checking cost (min) | total # of checks |
|---|---|---|---|---|---|
| 25 | 5 | 204 | 135 | 68 | 92 |
| 25 | 10 | 162 | 97 | 65 | 87 |
| 25 | 15 | 151 | 90 | 62 | 83 |
| 25 | 20 | 148 | 91 | 56 | 76 |

Table A.4: Optimal policy - checking time

Base Case

| n | b | checking time per SB (min) | | | checking time per element (min) | | |
|---|---|---|---|---|---|---|---|
| | | average | minimum | maximum | average | minimum | maximum |
| 25 | 5 | 10.91 | 7.04 | 14.20 | 2.18 | 0.60 | 3.03 |
| 25 | 10 | 5.40 | 2.97 | 9.02 | 2.16 | 0.60 | 2.42 |
| 25 | 15 | 3.60 | 1.11 | 7.26 | 2.16 | 0.60 | 2.42 |
| 25 | 20 | 2.67 | 0.50 | 4.24 | 2.14 | 0.60 | 2.42 |

Increasing error probabilities by 25%

| n | b | checking per SB (min) | | | checking time per element (min) | | |
|---|---|---|---|---|---|---|---|
| | | average | minimum | maximum | average | minimum | maximum |
| 25 | 5 | 11.03 | 7.65 | 14.20 | 2.21 | 0.60 | 3.03 |
| 25 | 10 | 5.46 | 3.47 | 9.02 | 2.18 | 0.60 | 2.42 |
| 25 | 15 | 3.68 | 1.11 | 7.86 | 2.21 | 0.60 | 2.42 |
| 25 | 20 | 2.73 | 0.50 | 4.24 | 2.18 | 0.60 | 2.42 |

[t]

Increasing checking costs by 25%

| n | b | checking per SB (min) | | | checking time per element (min) | | |
|---|---|---|---|---|---|---|---|
| | | average | minimum | maximum | average | minimum | maximum |
| 25 | 5 | 13.63 | 8.80 | 17.74 | 2.73 | 0.76 | 3.78 |
| 25 | 10 | 6.46 | 3.64 | 11.28 | 2.59 | 0.76 | 3.03 |
| 25 | 15 | 4.11 | 1.38 | 8.32 | 2.46 | 0.76 | 3.03 |
| 25 | 20 | 2.82 | 1.38 | 4.41 | 2.25 | 0.76 | 3.03 |

Table A.5: Optimal policy - number of checks

Base case

| n | b | number of checks per SB | | | number of checks per element | | |
|---|---|---|---|---|---|---|---|
| | | average | minimum | maximum | average | minimum | maximum |
| 25 | 5 | 18.4 | 12 | 24 | 3.7 | 1 | 5 |
| 25 | 10 | 9.1 | 5 | 15 | 3.6 | 1 | 4 |
| 25 | 15 | 6.1 | 2 | 12 | 3.6 | 1 | 4 |
| 25 | 20 | 4.5 | 1 | 7 | 3.6 | 1 | 4 |

Increasing error probabilities by 25%

| n | b | checking per SB (min) | | | checking time per element (min) | | |
|---|---|---|---|---|---|---|---|
| | | average | minimum | maximum | average | minimum | maximum |
| 25 | 5 | 18.6 | 13 | 24 | 3.72 | 1 | 5 |
| 25 | 10 | 9.2 | 6 | 15 | 3.68 | 1 | 4 |
| 25 | 15 | 6.2 | 2 | 13 | 3.72 | 1 | 4 |
| 25 | 20 | 4.6 | 1 | 7 | 3.68 | 1 | 4 |

Increasing checking costs by 25%

| n | b | checking per SB (min) | | | checking time per element (min) | | |
|---|---|---|---|---|---|---|---|
| | | average | minimum | maximum | average | minimum | maximum |
| 25 | 5 | 18.4 | 12 | 24 | 3.68 | 1 | 5 |
| 25 | 10 | 8.7 | 5 | 15 | 3.48 | 1 | 4 |
| 25 | 15 | 5.5 | 2 | 11 | 3.32 | 1 | 4 |
| 25 | 20 | 3.8 | 2 | 6 | 3.04 | 1 | 4 |