Purdue University
Purdue e-Pubs

Open Access Dissertations | Theses and Dissertations

January 2016

MODELS AND OPTIMIZATION FOR ELECTIVE SURGERY SCHEDULING UNDER UNCERTAINTY CONSIDERING PATIENT HEALTH CONDITION

Joonyup Eun
Purdue University

Graduate School Form
30 Updated 12/26/2015

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Joonyup Eun

Entitled
MODELS AND OPTIMIZATION FOR ELECTIVE SURGERY SCHEDULING UNDER UNCERTAINTY CONSIDERING PATIENT HEALTH CONDITION

For the degree of Doctor of Philosophy

Is approved by the final examining committee:

Yuehwern Yih
Chair

Seokcheon Lee

Andrew Lu Liu

Masha Shunko

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy of Integrity in Research" and the use of copyright material.

Approved by Major Professor(s): Yuehwern Yih

Approved by: Abhijit Deshmukh 7/20/2016

Head of the Departmental Graduate Program Date

MODELS AND OPTIMIZATION FOR ELECTIVE SURGERY SCHEDULING UNDER UNCERTAINTY CONSIDERING PATIENT HEALTH CONDITION

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Joonyup Eun

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 2016

Purdue University

West Lafayette, Indiana

ACKNOWLEDGMENTS

Fist of all, I would like to express my sincere gratitude to my advisor, Dr. Yuehwern Yih, for her support and guidance during my years at Purdue. I was lucky to meet her and to be her student. She opened my eyes on healthcare operations management.

I would also like to thank my doctoral advisory committee members, Dr. Seokcheon Lee, Dr. Andrew L. Liu, and Dr. Masha Shunko, for their assistance and advice to make this dissertation rich.

I am grateful to Dr. Sang-Phil Kim and Dr. Sangbok Lee for their helpful comments on my research and life.

I should not forget to thank my IE friends at Purdue: Sukwon, Yul, Chulhun, Eddie, Jeongjun, Seokhoon, Kihyung, Bomi, Jieun, Harsh, Varun, Bobby, Kang-yu, Joanna, Shan, Hongyue, and Tian. I was enjoyable at Purdue and I will miss Purdue life because they were with me.

Last but certainly not least, I am really thankful to my wife, Keunyoung, and my lovely son, Aiden, for their selfless support and love. They are everything to me.

TABLE OF CONTENTS

LIST OF TABLES

Table Page

LIST OF FIGURES

ABSTRACT

Eun, Joonyup Ph.D., Purdue University, August 2016. Models and Optimization for Elective Surgery Scheduling under Uncertainty Considering Patient Health Condition. Major Professor: Yuehwern Yih.

The managerial aspects to run a healthcare system are becoming increasingly important for patient safety. More than one patients are competing each other to be treated using limited medical resources in a healthcare system. The limited medical resources include surgeons, physicians, anesthesiologists, nurses, operating rooms, wards, etc. Therefore, patient safety is related to how to run a healthcare system with the limited resources.

Surgery scheduling, one of the managerial aspects to run a healthcare system, can contribute to improving patient safety. Diseases exacerbate patient health condition with the increase of waiting time for surgery. Therefore, surgeons and patients may want to schedule their surgeries as early as possible in order to escape from the risk of patients' deaths or the risk of turning the current diseases into more severe diseases. However, the needs may not be satisfied due to the limited medical resources.

This research incorporates deteriorating patient health condition in elective surgery scheduling to improve patient safety. Two different models are presented: elective surgery scheduling models with 1) linearly deteriorating patient health condition and 2) step- deteriorating patient health condition. In this research, the basis to manage uncertainties in surgery durations and/or patient health condition is sample average approximation. However, in general, the sample average approximation algorithm is time consuming. Therefore, a fastest local search and a tabu search are also developed to solve large-size problems.

1. INTRODUCTION

1.1 Background

Healthcare service providers have paid increased attention to patient safety in recent decades [1]. To improve patient safety, medical science focuses on improving the reporting and prevention of medical errors, the technical skills of surge, and the quality of medication [2–5]. In addition, even though it is not only for patient safety but also for human safety, infection control has drawn much attention recently due to the propagation of fatal infectious viruses such as ebola and swine flu. Likewise, most of approaches to improve patient safety focus on improving medical skills. Those medical skills play important roles for patient safety. However, patient safety cannot be improved without a wide range of tools that identify the sources of patient risk [6]. For example, patients may not be treated due to the lack of operating rooms (ORs) or wards in a hospital even though the hospital has many competent doctors. Therefore, beyond the improvements of medical skills, other aspects of a healthcare system should also be investigated.

The managerial aspects to run a healthcare system are becoming increasingly important for patient safety [7]. Generally, more than one people are competing each other to use limited resources in a system [8]. Likewise, more than one patients are competing each other to be treated using limited medical resources in a healthcare system. The limited medical resources include surgeons, physicians, anesthesiologists, nurses, ORs, wards, etc. Therefore, patient safety is also related to how to run a healthcare system with the limited resources. That is why operations research and management science are needed to run a healthcare system.

Surgery scheduling, one of the managerial aspects to run a healthcare system, can contribute to improving patient safety. Diseases exacerbate the health conditions of patients with the increase of waiting time for surgeries. Therefore, surgeons and patients may want to perform the surgeries as early as possible in order to escape from the risk of patients' deaths or turning the current diseases into more severe diseases. However, the resource limitation on surgeons, anesthesiologists, nurses, ORs, and post-anesthesia care units (PACUs) may not satisfy the needs of surgeons and patients who ask for early surgeries. Therefore, surgical scheduler (anesthesiologists or surgeons) need to schedule surgeries considering patient health condition. Roughly speaking, urgent patients should be scheduled first for their surgeries while patients with good health conditions need to be scheduled later than the urgent patients. In reality, surgical schedulers may often consider patient health condition in scheduling surgeries. However, to the best of my knowledge, there is no known surgery scheduling models which systematically consider patient health condition. A new systematic mathematical model to incorporate patient health condition in surgery scheduling needs to be developed.

On the other hand, reducing the cost for healthcare service is also important. The healthcare expenditure in the U.S. is 17.5% of Gross Domestic Product (GDP) in 2014 and projected to reach 19.6% of GDP by 2024 [9]. It means that people in the U.S. spends a great portion of their incomes for healthcare service and continuously need to spend more portion of their incomes to be healthy. Therefore, a wide range of efforts to reduce the cost for healthcare service is needed. Figure 1.1 illustrates the healthcare expenditure in the U.S. from 2000 to 2014 [10]. The healthcare expenditure in 2014 is more than twice the healthcare expenditure in 2000. It is rapidly increasing and expected to be rapidly increasing.

OR is the most cost intensive area in hospitals. Surgery operations comprise more than 40% of the expenses of hospitals [11–14]. Therefore, hospitals are under pressure to control their surgical cost. With regard to surgery scheduling problems, minimizing overutilization and underutilization of ORs can be considered to reduce the surgical costs of hospitals. The schedule considering patient health condition prevents underutilization to some degree.

Overutilization of ORs has a great impact on the cost of hospitals. 100 U.S. hospitals found, in a 2005 study, that it cost $62 on average to use an operating room for a minute [15]. Since the cost of running surgery staff (surgeons, anesthesiologists, and nurses) may

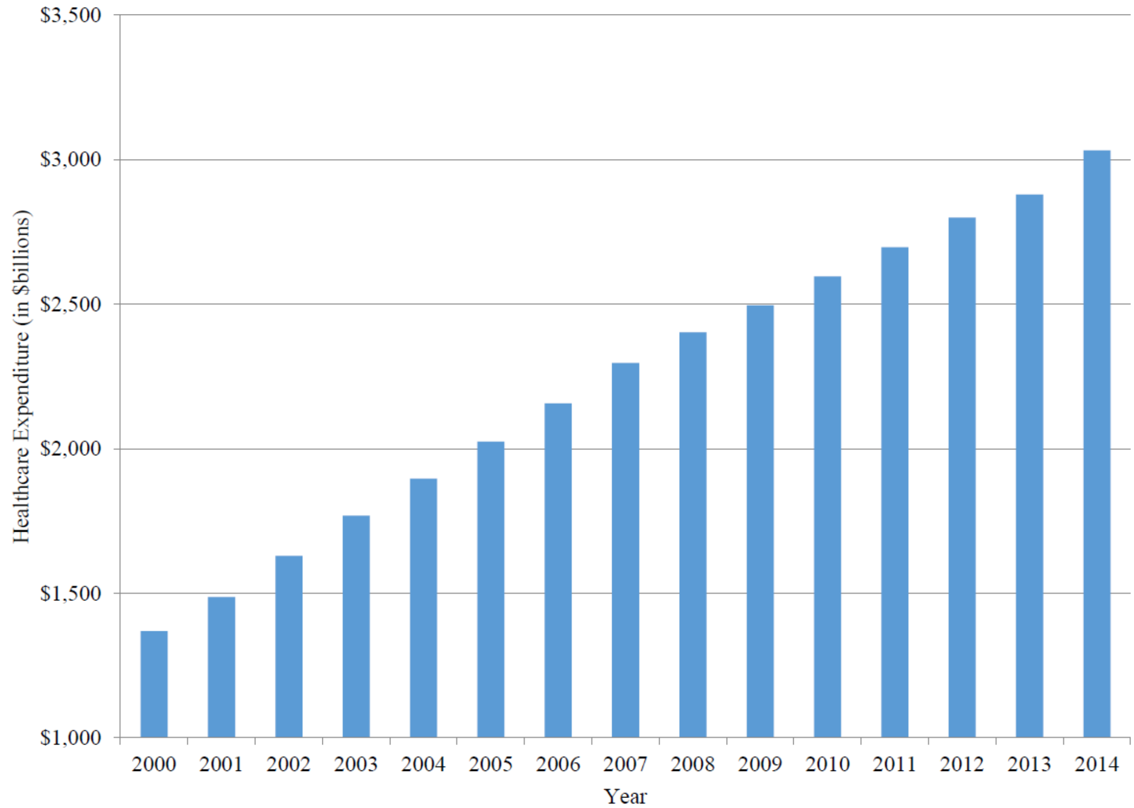


Figure 1.1. Healthcare expenditure from 2000 to 2014.

be more expensive during overtime, the cost to use an operating room during overtime is expected more expensive than $62 per minute. Therefore, surgery scheduling problems significantly need to deal with OR overutilization issues.

1.2 Objectives

The dissertation focuses on obtaining a schedule that incorporates the idea of maximizing the minimum patient health condition, which improves critical patient safety, and analyze the schedule. The minimum patient health condition is defined as the health condition of the most critical patient. While the schedule maximizing total patient health condition can delay the surgery of the most critical patient for the benefits of the entire patients, the schedule maximizing the minimum patient health condition prevents the risk of sentinel events. The idea of maximizing the minimum patient health condition has not been discussed in previous work.

In addition, the dissertation contributes to solution methodologies to solve the elective surgery scheduling problems. The uncertainty in surgery durations is considered in Chapters 3 and 4, and that in patient health condition is considered in Chapter 5. The basis of solution approaches to manage the uncertainties is the sample average approximation (SAA) method [16]. However, the algorithm that implements the SAA method is time consuming for a problem described in Chapter 3. To make up for the time inefficiency of the SAA method, heuristics with good performance (i.e., fastest ascent local search and tabu search) are developed to solve large problems in Chapter 4. On the other hand, the SAA method applied to a problem described in Chapter 5 provide effective solution within short computation times.

1.3 Organization of the Dissertation

The remainder of this dissertation is organized as follows. In Chapter 2, a broad review of the literature on surgery scheduling and a brief review of the SAA method. Chapter 3 describes an elective surgery scheduling problem with linearly deteriorating patient health condition. The problem is formulated as a stochastic mixed integer program (MIP) and solved using the SAA method. However, since the SAA method is very time consuming for the problem, two heuristics are proposed for the same problem in Chapter 4. Chapter 5 presents an elective surgery scheduling problem with step-deteriorating patient health condition. The problem is also formulated as a stochastic MIP and solved using the SAA method. The SAA method applied to the problem is efficient and effective. In addition to the solution methodology, the analyses on the SAA solutions are presented in Chapter 5. Finally, Chapter 6 discusses the conclusion of this dissertation and future research.

2. LITERATURE REVIEW

Surgery scheduling problems have been extensively studied in the literature. Magerlein and Martin [17], Przasnyski [18], Blake and Carter [19], Cardoen et al. [7], May et al. [20], and Hulshof et al. [21] provide detailed literature reviews on surgery scheduling problems.

Different from the existing literature reviews, this chapter focuses more on the literature related to consideration of OR overutilization since the overutilization of ORs is considered in the problems of this dissertation.

In addition, surgery scheduling problems prioritizing patients and imposing waiting time-dependent costs are presented in Section 2.2. A similar concept to patient health condition presented in the dissertation is to classify patients into each priority group when they are admitted to the waiting list and, according to the priority, impose patient-related costs that increase in waiting time.

Finally, a brief review on the SAA method and the surgery scheduling literature using the SAA method is presented in Section 2.3.

2.1 Overutilization of Operating Rooms

This section provides the surgery scheduling literature that considers the overutilization of ORs. “Integer program” in this section means either “mixed integer program” or “pure integer program”.

Adan and Vissers [22] decided the number of patients admitted to a specialty and the combination of patients in terms of specialties. They considered the underutilization and overutilization of ORs, wards, and intensive care units. They developed an integer program to solve the proposed problem.

Bowers and Mould [23] focused on the surgery demand of orthopaedic trauma patients. They incorporated the uncertainty in the surgery demand. They uses a simulation model to maximize the utilization of ORs. In addition, the overutilization of ORs were also examined using the simulation model.

Bruni et al. [24] suggested three different recourse strategies, which can be adopted by surgical schedulers selectively. Those recourse strategies are overtime of ORs, swapping surgeries, and rescheduling surgeries. They considered the uncertainties in the arrival of emergency patients and surgery durations. Tailored heuristics were developed to solve their models.

Cardoen and Demeulemeester [25] modeled the patient flow in a hospital from admission to discharge. They observed the patient length of stay in each suite (pre-surgical consultaion, surgery, and post-surgical consultation) of hospital and the utilization of each suite using a discrete-event simulation and suggested the reasonable numbers of wards and operating rooms. They used actual data of Belgian hospitals.

Chaabane et al. [26] suggested two surgery scheduling models. The first model is to obtain a surgery schedule which minimizes the gap between the total supply and the weekly requests of the surgical specialty. Another model is to obtain a surgery schedule which minimizes the sum of the surgery operating cost and patient waiting time-dependent cost. Integer programs are employed to solve those two models. They compared two models by the rate of occupancy time of ORs, percentage of total surgeries performed, percentage of planned surgical cases that actually performed, and overtime of ORs. Even though they used real data of a Belgian university hospital, they estimated the surgery durations as the means of the real data.

Denton and Gupta [27] suggested a surgery scheduling model which minimizes the weighed sum of patient waiting time, undertime (underutilization) of an OR, and tardiness of an OR. They used the L-shaped method [28] to solve the proposed problem. Denton et al. [11] solved the same problem using the SAA method and several heuristics. They compared the performances of the SAA method and the heuristics in terms of computation time and the quality of solutions.

Denton et al. [29] proposed two types of surgery scheduling models. The first one is the model to minimize the sum of the overtime cost of ORs and the fixed cost of opening

ORs. In the problem, a surgical scheduler can decide whether to open each OR or not. They developed bounds on the optimal solution in integer programs. Easy-to-implement heuristics were also introduced and compared in the experiment.

Dexter [30] studied a decision to decide whether to move the last surgery of the day in an OR to another empty OR to reduce overtime cost. They simulated two decision: performing the surgery scheduled the last of the day and moving the surgery to another empty OR having penalties for moving the surgery. The result that they found was depending on parameters they set between overtime cost and moving penalties. They gave the range of parameters that yields the decision.

Dexter and Macario [31] examined when to released allocated OR time for surgical services to improve OR efficiency (underutilization and overutilization). They found that postponing the decision whether to perform the surgery until early morning before surgery reduces overtime at most 15 minutes compared to the case to fix the surgery schedule 5 days before surgery. They obtained this result from actual implementation.

Dexter and Traub [32] came up with two surgery scheduling strategies: earliest start time and latest start time. They wanted to compare the OR efficiency between the two scheduling rules. The surgery is scheduled into an OR as early as possible in the earliest start time strategy, and the surgery is scheduled into an OR as late as possible in the latest start time strategy. They simulated the two strategies under several scenarios. They found that there are not significant differences of OR efficiency between two strategies.

Dexter et al. [33] studied about releasing the allocated OR time to another surgeries. "Releasing OR time" means that assign one specialty's allocated OR time for another specialty's surgery which was already scheduled to other OR. Therefore, the OR that is releasing it's OR time may increase overutiliztion and the OR that lost the surgery previously scheduled may experience underutilization. They observed the pattern of underutilization and overutilization using actual implementation in a hospital.

Dexter et al. [34] experimented whether computer recommendations and status displays to surgical staff helps to reduce overutilization and underutilization of ORs. They are three types of decision aids to facilitate decision-making in ORs: passive status displays, active status displays, and/or command displays. They experimented the scenarios with/without the decision aids. There were no big difference of OR efficiency with/without computer recommendations and status displays. However, command displays increased the quality of decisions in ORs.

Epstein and Dexter [35] tested that the error in OR information system data that tells wrong information for surgeries and the operating rooms in which the surgeries should be performed. In common sense, the error in OR information system data increases the undertime and overtime of ORs and, thus, increases surgical staffing costs. It states that expensive and time-consuming data-cleaning of the OR information system is needed. However, using a simulation, they concluded that the error of the OR information system data did not significantly increase the undertime and overtime of ORs.

Fei et al. [36] suggested a surgery scheduling problem in which a set of surgeries are assign to several multi-functional ORs. The objective the the problem was to minimize the weighted sum of undertime and overtime cost of ORs. They used the Dantzig-Wolf [37] decomposition to reformulate an original integer problem proposed as a set partitioning problem. A branch-and-price algorithm which combines a branch-and-bound algorithm and the column generation method had been developed. They showed that the solution approach is capable of solving large problems.

Fügener et al. [38] considered multiple downstream units such as intensive care units and general wards in a surgery scheduling problem. They focused on minimizing the fixed cost, overcapacity cost, staffing cost, and weekend staffing cost of intensive care units and general wards. They considered ORs as suppliers of patients. They did not include the undertime or overtime cost of ORs in their objective function. However, since undertime and overtime of ORs affect the cost of downstream units, This paper is included in this literature review. They implemented a branch-and-bound algorithm after formulated the problem as an integer problem. In addition, they suggested two heuristics and those were compared to the branch-and-bound algorithm.

Gladish et al. [39] explained how to incorporate multi-objctive criteria in an integer problem rather than specifying each objective. Like Lee and Yih [14], they used fuzzy sets for generating surgery durations. They applied possibility theory [40] to construct an integer program. Using simulation, they demonstrated the decreasing number of patients in the waiting list as their solutions applied to scheduling surgeries.

Guinet and Chaabane [41] proposed a surgery scheduling problem which minimizes the sum of hospitalization cost, i.e., patient length of stay, and OR overtime cost. they considered human resource limitation for surgeons. The showed that their problem is NP-hard and developed a constructive heuristic which adds constraints to an integer problem while the algorithm is being implemented.

Gupta [42] described various surgical operations and associated decision problems. He divided the process of surgery scheduling into two parts. First, he suggested a elective surgery booking control which was formulated as an integer program to consider down-stream units such as PACUs, intensive care units, and wards. Then, he proposed a surgery scheduling formulation which focused on the operations of ORs to minimize the weighted sum of overutilization, underutilization, and tardiness of ORs.

Hans et al. [43] came up with a set of multi-objectives: minimizing overtime of ORs, maximizing free OR-days, and maximizing utilization of ORs. They did not try to find the solutions which give the optimal trade-off between the three objectives. They ranked the objectives in terms of importance. Several constructive and local search heuristics were proposed and those are compared. They used a historical data of an academic university in the Netherlands.

Herring and Herrmann [44] focused on a single-day surgery scheduling problem because scheduling elective surgeries should be dynamic. Therefore, they solved a single-day surgery scheduling problems sequentially as time elapses. They proposed a dynamic programming formulation to minimize the weighted sum of patient blocking and deferral cost and utilization cost of ORs. They analyzed the problem and suggested a stochastic dynamic programming formulation. In addition, several heuristics were proposed. The solution quality of the heuristics were extensively investigated.

Jebali et al. [45] introduced a two stage approach to schedule surgeries. The first stage is to assign surgeries to operating rooms. The second stage is to sequence the assign surgeries to minimize the sum of patient hospitalization , OR undertime, and OR overtime cost. There compared two strategies. The first strategy is surgery assignment decided in the first stage is not reconsidered. The second strategy is surgery assignment decided in the first stage is reconsidered to make it less constrained. They used integer program for both first and second stage. They concluded that, as expected, the second strategy performed better since it improves the utilization of ORs.

Lamiri et al. [46] prioritized two groups of patients: elective and emergency surgery patients. It is assumed that surgeries for emergency patients should be performed on the day of arrival. The sum of elective patient-related cost (e.g., hospitalization cost, patient waiting time cost, etc.) and overtime cost of ORs were minimized in an integer program. Since they assumed that surgery durations are stochastic, they generated the realization of surgery durations using Monte Carlo simulation and incorporated it in the integer program.

Lamiri et al. [47] made an slight difference in the problem description of the Lamiri et al.'s [46] problem. In addition to elective patient-related cost overtime cost of ORs, they added undertime of ORs. The proposed problem were formulated as an integer program and a column generation approached is proposed to solve the integer program. A dynamic programming approach and local search algorithms that improve the solutions after obtaining solutions from the column generation and dynamic programming were also introduced and compared over several combinations.

Lebowitz [48] argued that surgeries with short surgery durations should be scheduled first. He simulated several scenarios in term of surgery durations and observed patient and surgeon waiting time, and overutilization and underutilization of ORs. In all measures he observed, the SPT rule for surgery durations proved outstanding performance. He stated that less inherent variability of short surgery durations improve the performance measures.

Lee and Yih [14] considered PACUs in a surgery scheduling problem. he formulated it as a flexible job shop models and assumed fuzzy sets of surgery durations. Performance measure they used were waiting times of patient and surgeons, undertime of ORs, and completion time of all the surgical processes until discharged from PACUs. He solved the proposed problem using a generic algorithm and adding a solution-improving algorithm after obtaining the solutions of the generic algorithm.

Mannino et al. [49] focused on two variants arisen from a large hospital in Norway. The first is balancing the number of patients waiting for different specialties and the second is minimizing the overtime of ORs. They suggested two integer programs with each objective. For the second problem, they developed a light robustness approach. Deterministic surgery durations are used in the integer program. Their formulations were solved by XPRESS-MP 19.00 and solutions were tested by simulation.

Marcon and Dexter [50] observed impact of surgery schedule on PACU staffing and overutilization of ORs resulting from admission delays in PACUs. Seven surgery scheduling rules in term of surgery duration were compared: (1) random assignment, (2) longest surgeries first, (3) shortest surgeries first, (4) Jonhson's scheduling rule [51], (5) half increase and half decrease, (6) half decrease and half increase, and (7) mix of (5) and (6). They used simulation to compare those seven scheduling rules. The best rules that minimizes PACU staffing cost and overutilization of ORs were (5) and (7).

Marcon et al. [52] saw the surgery-related operations as "the fruit of negotiation of different actors of the block such as surgeons, anesthetists, nurses, managerial staff, etc." Therefore, they developed an integer problem focusing on the negotiation relationship. The integer program minimizes the weighted sum of the difference of workload between ORs and the risk of no surgery-realization. They solve the integer program and tested the solutions of the integer program using simulation.

Marques et al. [53] assumed that the overtime of ORs is not permitted. Since there used deterministic surgery durations, they could incorporate the constraint in an integer program. The objective of the integer program was to maximize the utilization of ORs (i.e., minimize the underutilization of ORs). Real data from a hospital in Portugal were used to evaluate the solution of the integer program.

Min and Yih [13] proposed a surgery scheduling problem with consideration of PACUs to minimize the weighed sum of patient-related cost (e.g., hospitalization cost depending on waiting time) and overtime costs. They formulated the proposed problem as a two-stage stochastic model and solved it using the SAA method. The solutions of the SAA method were compared to those of the expected value model and test data of simulations. Due to the uni-modularity of the integer program proven in this paper, the SAA method provided the solutions within reasonable computation times.

Min and Yih [54] considered the priorities of patients. Patients were divided by several priority groups tagged as recommended surgery timings. Once surgeries passes their recommended surgery timings, associated costs that increase depending on waiting time for surgeries are imposed. They analyzed the problem and those properties found were implemented in a dynamic programming. Three priority groups were actually tested in the experiment.

Ogulata and Erol [55] introduced three stages of surgery scheduling: patient acceptance planning, assignment to surgeon groups, and sequencing surgeries. To make a decision at each stage, each stage was formulated as an integer program. Performance measures selected were the underutilization and overutilization of ORs, balanced workload between specialties, and patient waiting times. At each stage, appropriate performance measures were incorporated in each integer program.

Ozkarahan [56] used a goal programming to consider the underutilization and overutilization of ORs, waiting time of patients, and preferences of surgeons. Those performance measures were represented as various terms in a formulation. The formulation allows a surgical scheduler to adjust the weight of each term so that the surgical scheduler are able to produce a surgery schedule easily under uncertain circumstance. This paper provided extensive surgery duration statistics of various types of surgeries.

Pham and Klinkert [12] formulated a scheduling problem as a generalized job shop scheduling problem. They considered the entire flow of patients in integrated hospitals: preoperative stage (preoperative holding units), perioperative stage (ORs), and postoperative stage (PACUs and intensive care units). An extensive integer program was proposed to describe the entire flow of patients. However, assuming the deterministic surgery durations, they can prevent more complex formulation. The objective was designed to minimize the makespan (i.e., discharge time of last patient) and overtime of ORs was forced to be minimized in the constraints.

Saadouli et al. [57] studied a elective surgery scheduling problem in an orthopedic surgery division. They aimed to minimize the makespan of the operating rooms while maintaining appropriate utilization of ORs. They used the 85th percentile of surgery durations in their MIP formulation to calculate the makespan.

Sciomachen et al. [58] ran simulation models for three scheduling rules: the longest waiting time first, the longest surgery time first, and the shortest surgery time first. Throughput (i.e., number of patients treated in ORs), utilization of ORs, number of overruns in ORs, and overtime of ORs were observed under each rule. They used four different simulation models changing the usage of input data collected.

Testi et al. [59] used a three phase approach for surgery scheduling: determining the number of sessions to be scheduled weekly, assigning ward and OR on a specific day, and determining sequence of surgeries within a day. First and second stages were formulated as an integer programs, and the sequence decision on third stage was determined by simulation. They focused on improving overtime and throughput of ORs, and on reducing waiting list.

Tyler et al. [60] was motivated to answer about what the optimum utilization of ORs. They set two operational goals: surgeries should start within 15 minutes of the scheduled time and surgeries should end no more than 15 minutes past the planned end of the day. A simulation model was used to observe the utilization of ORs. Within the operational goals they set, they observed that 85% to 90% was the highest utilization rate of ORs.

Van Huele and Vanhoucke [61] studied on integrating the physician rostering problem and the surgery scheduling problem. Their objective is to minimize overtime of the ORs. They used a deterministic mixed integer program to formulate the problem and solve it with CPLEX 12.5. They analyzed the solutions varying the number of ORs, time periods, the number of days for surgeries, and the number of physicians.

Wullink et al. [62] studied on determining the best way to reserve OR time for emergency surgeries. Two strategies for reserving emergency OR time were proposed: designating emergency-dedicated ORs and sharing OR capacities with elective surgeries. Patient waiting time, OR overtime, and OR utilization were evaluated. They simulated two strategies and observed that, when emergency and elective patients shared OR capacities, OR overtime, utilization, and patient waiting time for both elective cases and emergency cases were significantly improved.

2.1.1 Performance Measures

This subsection summarizes performance measures in the literature introduced in Section 2.1. See Tables 2.1, 2.2, and 2.3.

2.1.2 Solution Approaches and Consideration of Uncertainty in Surgery Durations

This subsection summarizes solution approaches and consideration of uncertain surgery durations in the literature introduced in Section 2.1. See Tables 2.4, 2.5, and 2.6.

Table 2.1
Performance measures in literature.

	Waiting Time			Underutilization			Overutilization						
Paper	PT	SN	TP	OR	WD	ICU	OR	WD	ICU	PACU	MKSN	PR/D	Other
Adan and Vissers [22]				X	X	X	X	X	X				
Bowers and Mould [23]				X			X					X	
Bruni et al. [24]	X						X						
Cardoen and Demeulemeester [25]	X		X		X		X					X	
Chaabane et al. [26]	X						X						X
Denton and Gupta [27]	X	X					X						
Denton et al. [11]	X	X					X						
Denton et al. [29]							X						X
Dexter [30]							X						
Dexter [63]				X			X						
Dexter and Epstein [64]				X			X						
Dexter and Macario [31]				X			X						
Dexter and Traub [32]				X			X						
Dexter et al. [33]				X			X						
Dexter et al. [34]				X			X						X

PT: patient,
SN: surgeon,
TP: throughput,
OR: operating room,
WD: ward,
ICU: intensive care unit,
PACU: post-anesthesia care unit,
MKSN: makespan,
PR/D: patient reject/deferral.

Table 2.2
Performance measures in literature (cont'd).

	Waiting Time			Underutilization			Overutilization						
Paper	PT	SN	TP	OR	WD	ICU	OR	WD	ICU	PACU	MKSN	PR/D	Other
Epstein and Dexter [35]				X			X						
Fei et al. [36]				X			X						
Fügener et al. [38]				X			X	X	X				
Gladish et al. [39]	X			X			X						
Guinet and Chaabane [41]	X						X						
Gupta [42]	X	X					X						X
Hans et al. [43]				X			X						
Herring and Herrmann [44]				X			X				X		
Jebali et al. [45]	X			X			X						
Lamiri et al. [46]							X						X
Lamiri et al. [47]				X			X						X
Lebowitz [48]	X	X		X			X						X
Lee and Yih [14]	X	X		X			X			X			
Mannino et al. [49]	X						X						

PT: patient,
SN: surgeon,
TP: throughput,
OR: operating room,
WD: ward,
ICU: intensive care unit,
PACU: post-anesthesia care unit,
MKSN: makespan,
PR/D: patient reject/deferral.

Table 2.3
Performance measures in literature (cont'd).

	Waiting Time			Underutilization			Overutilization						
Paper	PT	SN	TP	OR	WD	ICU	OR	WD	ICU	PACU	MKSN	PR/D	Other
Marcon and Dexter [50]							X				X		X
Marcon et al. [52]							X						
Marques et al. [53]				X			X						
Min and Yih [13]				X			X						
Min and Yih [54]	X						X						
Ogulata and Erol [55]	X			X			X						X
Ozkarahan [56]				X			X		X				X
Pham and Klinkert [12]							X				X		X
Sciomachen et al. [58]	X		X				X						
Saadouli et al. [57]	X			X			X				X		
Testi et al. [59]	X		X				X				X		X
Tyler et al. [60]				X			X						X
Van Huele and Vanhoucke [61]							X						
Wullink et al. [62]	X						X						

PT: patient,
SN: surgeon,
TP: throughput,
OR: operating room,
WD: ward,
ICU: intensive care unit,
PACU: post-anesthesia care unit,
MKSN: makespan,
PR/D: patient reject/deferral.

Table 2.4
Solution approaches and consideration of uncertain surgery durations in literature.

	Solution Approach					Surgery Durations	
Paper	MP (MIP)	MP (DP)	HEU	SIM	AI	Deterministic	Stochastic
Adan and Vissers [22]	X					X	
Bowers and Mould [23]				X			X
Bruni et al. [24]			X				X
Cardoen and Demeulemeester [25]				X			X
Chaanane et al. [26]	X					X	
Denton and Gupta [27]	X						X
Denton et al. [11]	X		X				X
Denton et al. [29]	X		X				X
Dexter [30]				X			X
Dexter [63]				X			X
Dexter and Epstein [64]				X			X
Dexter and Macario [31]					X		
Dexter and Traub [32]				X			X
Dexter et al. [33]					X		
Dexter et al. [34]					X		

MP: mathematical programming,
HEU: heuristic
SIM: simulation,
AI: actual implementation,
MIP: mixed integer programming (including any techniques based on MIP),
DP: dynamic programming,

Table 2.5
Solution approaches and consideration of uncertain surgery durations in literature (cont'd).

	Solution Approach					Surgery Durations	
Paper	MP (MIP)	MP (DP)	HEU	SIM	AI	Deterministic	Stochastic
Epstein and Dexter [35]				X			X
Fei et al. [36]	X					X	
Fügener et al. [38]	X		X			X	
Gladish et al. [39]	X						X
Guinet and Chaabane [41]			X				
Gupta [42]	X						X
Hans et al. [43]			X	X			X
Herring and Herrmann [44]		X	X				X
Jebali et al. [45]	X					X	
Lamiri et al. [46]	X						X
Lamiri et al. [47]	X						X
Lebowitz [48]				X			X
Lee and Yih [14]			X				X
Mannino et al. [49]	X					X	

MP: mathematical programming,
HEU: heuristic
SIM: simulation,
AI: actual implementation,
MIP: mixed integer programming (including any techniques based on MIP),
DP: dynamic programming,

Table 2.6
Solution approaches and consideration of uncertain surgery durations in literature (cont'd).

	Solution Approach					Surgery Durations	
Paper	MP (MIP)	MP (DP)	HEU	SIM	AI	Deterministic	Stochastic
Marcon and Dexter [50]				X			X
Marcon et al. [52]	X			X			X
Marques et al. [53]	X					X	
Min and Yih [13]	X						X
Min and Yih [54]		X					X
Ogulata and Erol [55]	X					X	
Ozkarahan [56]	X					X	
Pham and Klinkert [12]	X					X	
Saadouli et al. [57]	X					X	
Sciomachen et al. [58]				X			X
Testi et al. [59]	X			X		X	X
Tyler et al. [60]				X			X
Van Huele and Vanhoucke [61]	X					X	
Wullink et al. [62]				X			X

MP: mathematical programming,
HEU: heuristic
SIM: simulation,
AI: actual implementation,
MIP: Mixed integer programming (including any techniques based on MIP),
DP: dynamic programming,

2.2 Penalizing Waiting Time based on Group Priority

Cardoen et al. [7] classified the literature according to performance measures that evaluate surgery scheduling procedures; the performance measures are, for instance, waiting time of patients, utilization of operating rooms, leveling of resources (e.g., operating rooms, wards, and post-anesthesia care units), makespan (i.e., completion time of the last patient's recovery), number of patient deferrals or refusals, cost savings, and preferences of the different parties involved in the surgery scheduling.

However, to the best of my knowledge, there is a lack of surgery scheduling literature that considers time-dependent patient health condition as a performance measure.

In the literature, a similar concept to patient health condition presented in this research is to classify patients into each priority group when they are admitted to the waiting list and, according to the priority, impose patient-related costs that increase with respect to waiting time (e.g., hospitalization costs and penalties for passing the optimal surgery date).

When elective surgery patients and emergency surgery patients share the operation room capacity, two groups of patients with different priorities are naturally considered. Gerchak et al. [65] suggested an elective surgery scheduling problem under the assumption that operating rooms can be utilized by emergency surgeries as well as elective surgeries, and the number of emergency cases arriving each day is uncertain. Elective surgeries that are unable to be scheduled for the requested day, due to the full capacity of the operating rooms, are rescheduled penalizing the profit margin of a hospital. They decided the number of elective surgeries scheduled in a day to maximize the expected profit for the hospital. Lamiri et al. [46] dealt with a similar problem. The difference in the problem setting is the use of a multi-period model, while Gerchak et al. [65] used a mono-period model. They designed patient-related costs as a stepwise increasing function with respect to waiting time in a multi-period model and obtained an elective surgery schedule to minimize the sum of patient-related and overtime costs. In Gerchak et al. [65] and Lamiri et al. [46], patients were prioritized into two groups: elective and emergency cases.

There are a couple of papers that consider multiple patient groups to represent patient priorities for elective surgery. Gupta [42] suggested an elective surgery booking control with multiple patient groups. He divided all surgery types into multiple classes with each group having a maximum delay the surgery must be performed by. Overtime costs that are incurred to satisfy the maximum delays penalized the hospital's profit which is the objective function of his problem. Similarly, Min and Yih [54] used multiple patient groups for their elective surgery scheduling problem. They did not specify the maximum delays. However, they designed patient-related costs as a stepwise increasing function in a multi-period setting and obtained an elective surgery schedule to minimize the sum of patient-related costs and overtime costs like Lamiri et al. [46].

The problems that minimize total patient-related costs, in the previous work, can be converted to those that maximize total patient health condition.

However, this study uses a different approach to formulate an elective surgery problem. While the previous approaches can delay the surgery for the most critical patient to maximize total patient health condition and, therefore, expose the patient at risk of death, this study focuses on improving critical-patient safety by maximizing minimum patient health condition.

2.3 Sample Average Approximation Method

In order to discuss a methodology to solve a stochastic MIP presented in the dissertation, the SAA method needs to be addressed. The general idea to tackle stochastic MIPs is to convert the stochastic MIPs to the corresponding "deterministic equivalent" [66] models and then solve the deterministic equivalent models using well-established techniques for deterministic models. Most of the solution approaches that are currently available to tackle stochastic MIPs, such as the integer L-shaped method [67], stochastic branch-and-bound method [68], and SAA method, are implemented based on this general idea. A thorough review of the SAA method is given in Kleywegt et al. [16]. "The basic idea of the SAA method is simple indeed" [16]. A deterministic equivalent model with a set of scenarios is solved to approximate the objective function of the corresponding stochastic MIP. If the deterministic equivalent model is solved several times changing the set of scenarios each time, the quality (i.e., optimality gap) of solutions can be statistically obtained.

The SAA method was adopted in several papers to deal with surgery scheduling problems under uncertain surgery durations. Denton et al. [11] formulated an elective surgery scheduling problem as an SAA model (i.e., deterministic equivalent model of a stochastic MIP to implement the SAA method) to minimize the weighted sum of waiting time, idling time, and overtime costs. They also proposed simple heuristics. Min and Yih [13] presented an SAA model to minimize the weighted sum of patient-related and overtime costs. They proved the total unimodularity of their SAA model so that some integer variables were able to be relaxed as continuous variables. Mancilla and Storer [69] dealt with a similar SAA model to that of Denton et al. [11]. They incorporated the Benders' decomposition [70] to the SAA method.

Since the algorithm that implements the SAA method is time consuming and, thus, limited to solve large-size problems, the previous work developed heuristics [11], relaxed integer variables as continuous variables [13], or added a decomposition technique to the SAA method [69].

In a similar way to the above referenced work, this research develops heuristics in Chapter 4 to provide effective solutions for large size problems, in addition to implementing the SAA method to solve the proposed stochastic MIP.

3. SAMPLE AVERAGE APPROXIMATION APPROACH TO ELECTIVE SURGERY SCHEDULING WITH LINEARLY DETERIORATING PATIENT HEALTH CONDITION

3.1 Introduction

Hospitals are under increasing pressure to improve patient safety [71]. Surgery scheduling without considering patient health condition exposes patients at risk of death or decreases their safety.

This study incorporates deteriorating patient health condition in elective surgery scheduling to improve patient safety. Diseases may exacerbate patient health condition with the increase of waiting time for surgery. Abdominal disease is a good example for these diseases. Abdominal disease is caused by inflammation, calculi, ulcers, abscesses, or malignant tumors which exacerbate patient health condition over time [72].

In addition to patient health condition, overutilization of an operating room (OR) is considered in this study. Since more than 40% of a hospital's expenses have been estimated for surgery operation [11, 12], hospitals are under pressure to control their surgical costs. With regard to OR planning problems, minimizing overutilization and/or underutilization of ORs can be considered to reduce a hospital's surgical costs. Since the schedule considering patient health condition already prevents underutilization to some degree, this study deals with overutilization. Overutilization causes additional resource costs for surgeons, anesthetists, nurses, and ORs [12].

The main purpose of this study is to obtain a schedule which provides the optimal trade-off between two objectives: maximizing the minimum patient health condition and minimizing total overtime. The minimum patient health condition is the health condition of the most critical patient. A schedule maximizing the minimum patient health condition prevents patients to be at risk of sentinel events. However, the schedule may not be used by hospitals since the schedule cannot restrict overtime. Therefore, the trade-off of the two objectives, under uncertain surgery durations, is considered in this study.

The remainder of this chapter is organized as follows. The next section describes the problem and formulates it as a stochastic mixed integer program (MIP). Section 3.3 presents the SAA method. The computational study for the SAA algorithm is presented in Section 3.4. Section 3.5 concludes this chapter.

3.2 Problem Description

To schedule a surgery, surgical schedulers must know the time frame within which the surgery should be performed [73]. The time frame is decided appropriately by clinicians considering patient health condition [74, 75]. In other words, clinicians assess the health condition of a patient when he/she is diagnosed and, according to his/her health condition, set the critical time point by which the surgery should be performed.

This study uses the information about the current patient health condition and the critical time point to represent time-dependent patient health condition. The current patient health condition can be determined by clinicians using generic measures (e.g., expected life durations, impairments, and psychological/physical functions), disease-specific measures (e.g., dyspnea index for lung disease [76], Karnofsky grade for cancer [77], and model for end-stage liver disease score [78]), or some combination of them [79]. This study employs the line connecting the two points (i.e., current patient health condition and critical time point) as an approximation for deteriorating patient health condition.

A set of patients $J = \{1,\dots,n\}$ is to be scheduled for surgery in an OR on a day in the set of available days for surgeries $L = \{1,\dots,m\}$. Each patient $j \in J$ is characterized by its random surgery duration S_j and its health condition $HC_j(t)$ which is given as a linearly decreasing function depending on time t. Let b_j be the initial health condition of patient j at the beginning of day 1 and a_j be the time by which the surgery for patient j should start. If the surgery for patient j does not start by a_j, the health condition of patient j can become very critical or patient j may become deceased. It is assumed that physicians or surgeons are able to decide a_j and b_j. The health condition of patient j is defined in the problem as follows:

$$HC_j(t) := -\frac{b_j}{a_j}t + b_j \qquad a_j, b_j > 0 \tag{3.1}$$

where t is 0 at the beginning of day 1.

Available time duration of an OR on day $l \in L$ in which surgeries can be performed without overtime is represented by d_l, and time duration between d_l and d_{l+1} is represented by d_l'. Since an OR may not be continuously run 24 hours due to staffing or maintenance, allowable maximum overtime on day l, denoted by c_l, is introduced. Figure 3.1 illustrates the relationship among d_l, d_l', c_l, and d_{l+1} over time t. Surgery for patient j can be scheduled to the kth surgery , $k \in K$ (being K the set of surgery sequence index on a day), on day l. Total number of surgeries on day l is not fixed but an output of the model. The cardinality of set J is the same as that of set K because the problem considers the possibility that all patients are scheduled on a day. Therefore, the surgery assignment decision variable used in the model is:

$$x_j^{kl} = \begin{cases} 1, \text{ if a patient } j \text{ is assigned to the } k\text{th surgery on day } l \\ 0, \text{ otherwise} \quad (j \in J,\ k \in K, l \in L) \end{cases}$$

Let O_l be overtime on day l which exceeds the available time duration d_l. Let T_j be the surgery start time for patient j. The objective value of the problem has two components:

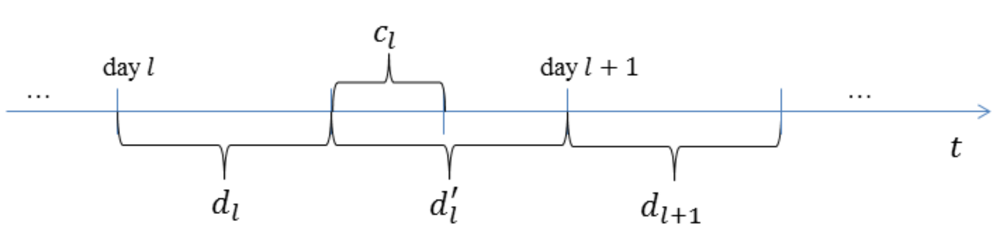


Figure 3.1. Relationship among d_l, d_l', c_l, and d_{l+1}.

the minimum patient health condition and total overtime. Since the problem considers uncertain surgery durations affecting the surgery start time of patient j and overtime on day l, $HC_j(T_j)$ and O_l are also uncertain even in a fixed surgical schedule. Thus, the expected value of the minimum patient health condition, $E[\min_j\{HC_j(T_j)\}]$, and the expected value of total overtime, $E[\sum_{l=1}^{m} O_l]$, are considered in the objective function. Note that the two components of the objective function have different units. Therefore, coefficient δ is used to compensate for the two different units, and $E[\min_j\{HC_j(T_j)\}] - \delta E[\sum_{l=1}^{m} O_l]$ is maximized. δ can be decided by surgical schedulers.

Let s_j^{μ} be the mean surgery duration of patient j and M be a sufficiently large number. A stochastic MIP used to formulate the problem of this study is as follows:

$$\max \ E[Y - \delta \sum_{l=1}^{m} O_l] \tag{3.2}$$

$$\text{s.t.} \ \sum_{k=1}^{n}\sum_{l=1}^{m} x_j^{kl} = 1, \qquad j = 1,...,n \tag{3.3}$$

$$\sum_{j=1}^{n} x_j^{kl} \leq 1, \qquad k = 1,...,n; l = 1,...m \tag{3.4}$$

$$\sum_{j=1}^{n} x_j^{kl} \geq \sum_{j=1}^{n} x_j^{k+1,l}, \qquad k = 1,...,n-1; l = 1,...,m \tag{3.5}$$

$$\sum_{k=1}^{n}\sum_{j=1}^{n} s_j^{\mu} x_j^{kl} - d_l \leq c_l, \qquad l = 1,...,m \tag{3.6}$$

$$\sum_{k=1}^{n}\sum_{j=1}^{n} S_j x_j^{kl} - d_l \leq O_l, \qquad l = 1,...,m \tag{3.7}$$

$$-\frac{b_j}{a_j}\left[\sum_{i=1}^{l-1}\left(d_i + d_i^{'}\right)x_j^{kl} + \sum_{g=1}^{k-1}\sum_{h=1}^{n} S_h x_h^{gl} - M\left(1 - x_j^{kl}\right)\right] + b_j \geq Y,$$
$$j = 1,...,n; k = 1,...,n; l = 1,...,m \tag{3.8}$$

$$x_j^{kl} \in \{0,1\}, \qquad j = 1,...,n; k = 1,...,n; l = 1,...,m \tag{3.9}$$

$$Y \ \text{free} \tag{3.10}$$

$$O_l \geq 0, \qquad l = 1,...,m \tag{3.11}$$

Y in the objective function (3.2) is a variable for the minimum patient health condition because the left-hand side of constraint (3.8) forces Y to become the minimum health condition of the most critical patient. If x_j^{kl} = 1 in constraint (3.8), the left-hand side yields the health condition of patient j at the start time of patient j's surgery. $\sum_{i=1}^{l-1}\left(d_i + d_i^{'}\right) x_j^{kl}$ determines the start time of day l and $\sum_{g=1}^{k-1}\sum_{h=1}^{n} S_h x_h^{gl}$ determines total durations of preceding surgeries on day l. Even though x_j^{kl} = 0, $\sum_{g=1}^{k-1}\sum_{h=1}^{n} S_h x_h^{gl}$ still remains. Therefore, M is subtracted if x_j^{kl} = 0. Constraint (3.3) imposes every patient to be scheduled. Constraint (3.4) ensures that more than one surgery cannot be operated simultaneously. Constraint (3.5) guarantees that any following surgery can be scheduled only after the preceding surgery is assigned. Constraint (3.6) states that surgeries are able to be scheduled only if total mean duration of surgeries fit within the time available for surgeries. This constraint is widely used for the block scheduling system [11, 13]. Overtime for each day is obtained in constraint (3.7).

Note that solving the (deterministic) MIP for every realization of the uncertain surgery durations is needed for the exact evaluation of the objective function in the stochastic MIP, which is "computationally prohibitive" (see Ahmed and Shapiro [80] for detailed description of difficulties in solving stochastic MIPs). This chapter suggests the SAA method in the following section.

3.3 Sample Average Approximation Method

This section constructs an SAA algorithm referring to Kleywegt et al. [16], Ahmed and Shapiro [80], and Bayraksan and Morton [81].

Let ω_r be the rth scenario that defines the rth realization of the random surgery duration vector $\vec{S} = (S_1, S_2, \ldots, S_n)$ and $s_j(\omega_r)$ be the element of scenario ω_r that defines the surgery duration of patient j. Note that Y and O_l used for the stochastic MIP are approximated by the objective function of the following SAA model. $y(\omega_r)$ and $o_l(\omega_r)$ are used to represent the realizations of Y and O_l respectively under scenario ω_r. It is assumed that total number of scenarios is q (i.e., the set of scenarios is $\{\omega_r | r = 1, \ldots, q\}$) to present an SAA model as follows:

$$\max \ \frac{1}{q}\sum_{r=1}^{q}\left[y(\omega_r) - \delta\sum_{l=1}^{m} o_l(\omega_r)\right] \tag{3.12}$$

$$\text{s.t.} \ \sum_{k=1}^{n}\sum_{l=1}^{m} x_j^{kl} = 1, \qquad j = 1,...,n \tag{3.13}$$

$$\sum_{j=1}^{n} x_j^{kl} \leq 1, \qquad k = 1,...,n; l = 1,...,m \tag{3.14}$$

$$\sum_{j=1}^{n} x_j^{kl} \geq \sum_{j=1}^{n} x_j^{k+1,l}, \qquad k = 1,...,n-1; l = 1,...,m \tag{3.15}$$

$$\sum_{k=1}^{n}\sum_{j=1}^{n} s_j^{\mu} x_j^{kl} - d_l \leq c_l, \qquad l = 1,...,m \tag{3.16}$$

$$\sum_{k=1}^{n}\sum_{j=1}^{n} s_j(\omega_r) x_j^{kl} - d_l \leq o_l(\omega_r), \qquad l = 1,...,m; r = 1,...,q \tag{3.17}$$

$$-\frac{b_j}{a_j}\left[\sum_{i=1}^{l-1}\left(d_i + d_i^{'}\right)x_j^{kl} + \sum_{g=1}^{k-1}\sum_{h=1}^{n} s_h(\omega_r) x_h^{gl} - M\left(1 - x_j^{kl}\right)\right] + b_j \geq y(\omega_r),$$

$$j = 1,...,n; k = 1,...,n; l = 1,...,m; r = 1,...,q \tag{3.18}$$

$$x_j^{kl} \in \{0, 1\}, \qquad j = 1,...,n; k = 1,...,n; l = 1,...,m \tag{3.19}$$

$$y(\omega_r) \ \text{free} \qquad r = 1,...,q \tag{3.20}$$

$$o_l(\omega_r) \geq 0, \qquad l = 1,...,m; r = 1,...,q \tag{3.21}$$

The SAA model needs to be solved several times with different sets of scenarios to obtain statistical results. The statistical results ensure the quality of solutions. Let u be the number of replications of the SAA model and ω_r^{ρ} be the scenario set $\{\omega_r^{\rho} | r = 1, \ldots, q\}$

used for the ρth SAA replication . The following definitions are used for a feasible schedule $X = \{x_j^{kl} | j \in J, k \in K, l \in L\}$:

$$\phi(X) := E\left[Y(X,\vec{S}) - \delta \sum_{l=1}^{m} O_l(X,\vec{S})\right],$$

$$v_q^\rho(X,\omega_r^\rho) := \frac{1}{q}\sum_{r=1}^{q}\left[y(X,\omega_r^\rho) - \delta \sum_{l=1}^{m} o_l(X,\omega_r^\rho)\right].$$

In the above definitions, $Y(X,\vec{S})$ and $O_l(X,\vec{S})$ represent Y and O_l in the stochastic MIP respectively but indicate that Y and O_l are dependent on X and $\vec{S}$. Likewise, $y(X,\omega_r^\rho)$ and $o_l(X,\omega_r^\rho)$ are another representations of $y(\omega_r^\rho)$ and $o_l(\omega_r^\rho)$ respectively.

Let X^* be the optimal solution of the stochastic MIP and $X^{\rho *}$ be the optimal solution of the ρth SAA replication. The optimality gap is defined as $\phi(X^*) - \phi(\hat{X})$ for a given solution $\hat{X}$. $\phi(X^*)$ and $\phi(\hat{X})$ are estimated respectively by

$$\tilde{v_q}^u := \frac{1}{u}\sum_{\rho=1}^{u} v_q^\rho\left(X^{\rho *},\omega_r^\rho\right), \tag{3.22}$$

$$\bar{v_q}^u\left(\hat{X}\right) := \frac{1}{u}\sum_{\kappa=1}^{u} v_q^\kappa\left(\hat{X},\omega_r^\kappa\right). \tag{3.23}$$

Note that $\tilde{v_q}^u$ is an optimistically biased estimator of $\phi(X^*)$ because

$$E\left[\tilde{v_q}^u\right] = \frac{1}{u}\sum_{\rho=1}^{u} E\left[v_q^\rho\left(X^{\rho *},\omega_r^\rho\right)\right] \geq \frac{1}{u}\sum_{\rho=1}^{u} E[v_q^\rho\left(X^*,\omega_r^\rho\right)] = \frac{1}{u}\sum_{\rho=1}^{u}\phi(X^*) = \phi(X^*).$$

Note that $\bar{v_q}^u(\hat{X})$ is an unbiased estimator of $\phi(\hat{X})$ because

$$\begin{aligned} E\left[\bar{v_q}^u\left(\hat{X}\right)\right] &= \frac{1}{u}\sum_{\kappa=1}^{u} E\left[\frac{1}{q}\sum_{r=1}^{q}\left\{y\left(\hat{X},\omega_r^\kappa\right) - \delta\sum_{l=1}^{m} o_l\left(\hat{X},\omega_r^\kappa\right)\right\}\right] \\ &= \frac{1}{u}\sum_{\kappa=1}^{u} E\left[Y\left(\hat{X},\vec{S}\right) - \delta\sum_{l=1}^{m} O_l\left(\hat{X},\vec{S}\right)\right] = \phi(\hat{X}). \end{aligned}$$

Since $E[\tilde{v_q}^u] \geq \phi(X^*)$ and $E[\bar{v_q}^u(\hat{X})] = \phi(\hat{X})$, $\tilde{v_q}^u - \bar{v_q}^u(\hat{X})$ overestimates the optimality gap $\phi(X^*) - \phi(\hat{X})$.

The variance of $\tilde{v_q}^u - \bar{v_q}^u(\hat{X})$ is $\frac{1}{u(u-1)}\sum_{\rho=1}^{u}\left[\left\{v_q^\rho\left(X^{\rho *},\omega_r^\rho\right) - v_q^\rho\left(\hat{X},\omega_r^\rho\right)\right\} - \left\{\tilde{v_q}^u - \bar{v_q}^u\left(\hat{X}\right)\right\}\right]^2$ because

$$Var\left[\tilde{v_q}^u - \bar{v_q}^u(\hat{X})\right] = Var\left[\frac{1}{u}\sum_{\rho=1}^{u} v_q^\rho\left(X^{\rho *},\omega_r^\rho\right) - \frac{1}{u}\sum_{\rho=1}^{u} v_q^\rho\left(\hat{X},\omega_r^\rho\right)\right]$$
$$= \frac{1}{u^2}\left\{Var\left[\sum_{\rho=1}^{u} v_q^\rho\left(X^{\rho *},\omega_r^\rho\right) - \sum_{\rho=1}^{u} v_q^\rho\left(\hat{X},\omega_r^\rho\right)\right]\right\} = \frac{1}{u^2}\sum_{\rho=1}^{u} Var\left[v_q^\rho\left(X^{\rho *},\omega_r^\rho\right) - v_q^\rho\left(\hat{X},\omega_r^\rho\right)\right]$$
$$= \frac{1}{u^2}\sum_{\rho=1}^{u}\frac{1}{u-1}\sum_{\rho=1}^{u}\left[\left\{v_q^\rho\left(x^{\rho *},\omega_r^\rho\right) - v_q^\rho\left(\hat{X},\omega_r^\rho\right)\right\} - \left\{\frac{1}{u}\sum_{\rho=1}^{u} v_q^\rho\left(X^{\rho *},\omega_r^\rho\right) - \frac{1}{u}\sum_{\rho=1}^{u} v_q^\rho\left(\hat{X},\omega_r^\rho\right)\right\}\right]^2$$
$$= \frac{1}{u(u-1)}\sum_{\rho=1}^{u}\left[\left\{v_q^\rho\left(X^{\rho *},\omega_r^\rho\right) - v_q^\rho\left(\hat{X},\omega_r^\rho\right)\right\} - \left\{\tilde{v_q}^u - \bar{v_q}^u\left(\hat{X}\right)\right\}\right]^2.$$

Note that $\tilde{v_q}^u - \bar{v_q}^u(\hat{X})$ is a sample mean of independent and identically distributed random variables. Thus, $\tilde{v_q}^u - \bar{v_q}^u(\hat{X})$ is approximately normally distributed for sufficiently large u by the central limit theorem. The $100(1-\alpha)\%$ confidence interval on the optimality gap (CIOOG) for a given solution $\hat{X}$ can be constructed as follows:

$$\tilde{v_q}^u - \bar{v_q}^u\left(\hat{X}\right) + z_\alpha\sqrt{\frac{1}{u(u-1)}\sum_{\rho=1}^{u}\left[\left\{v_q^\rho\left(X^{\rho *},\omega_r^\rho\right) - v_q^\rho\left(\hat{X},\omega_r^\rho\right)\right\} - \left\{\tilde{v_q}^u - \bar{v_q}^u\left(\hat{X}\right)\right\}\right]^2} \tag{3.24}$$

where z_α denotes the value such that $P(Z > z_\alpha) = \alpha$; Z is a standard normal random variable.

A proposed algorithm (SAA algorithm) for the problem of this study is as follows:

Step 1. Choose u and q.

Step 2. For each $\rho = 1,2,\ldots,u$, generate q scenarios and solve the ρth SAA replication. Obtain the optimal solution $X^{\rho *}$ and the corresponding objective value $v_q^\rho(X^{\rho *},\omega_r^\rho)$ of the ρth SAA problem.

Step 3. Calculate $\tilde{v_q}^u$ by (3.22).

Step 4. For each ρ, calculate $\bar{v_q}^u(X^{\rho *})$ by (3.23).

Step 5. Select one solution, denoted by $\hat{X}_{saa}$, that provides the maximum value of $\bar{v_q}^u(X^{\rho *})$.

Step 6. Construct the $100(1-\alpha)\%$ CIOOG for $\hat{X}_{saa}$ by (3.24).

3.4 Computational Study

The numerical experiments to test the SAA algorithm described in Sections 3.3 are presented in this section. The algorithm is implemented in the General Algebraic Modeling System (GAMS) 24.1.3 with CPLEX 12.5.1 for solving MIPs on a PC with a 2.4GHz Core i7 processor and 8GB RAM.

To generate the distribution of surgery duration S_j, several statistics of the six abdominal surgery durations from Strum et al. [82] are used. They are liver transplantation (class 1), abdomen exploration (class 2), inguinal hernia repair (class 3), kidney transplantation (class 4), laparoscopy and tubal cautery (class 5), and laparoscopic cholecystectomy (class 6). The percentages in Table 3.1 are calculated by the number of class's surgeries divided by the total number of abdominal surgeries. Surgeries are generated for the experiments based on each class's percentage. The study mean and study standard deviation ranges are shown in Table 3.1. These values are computed by allowing $\pm 20\%$ deviation from the mean and standard deviation of each class's surgery durations [82] to reflect considerable

Table 3.1
Generating surgery durations.

Surgery class	Percentage[a]	Mean[a,b]	Standard deviation[a,b]	Study mean[b]	Study standard deviation[b]
Class 1	23.83%	691	126	[552.8,829.2]	[100.8,151.2]
Class 2	24.11%	194	89.4	[155.2,232.8]	[71.52,107.28]
Class 3	21.16%	143	38.5	[114.4,171.6]	[30.8,46.2]
Class 4	16.96%	328	50	[262.4,393.6]	[40,60]
Class 5	7.52%	105	27.4	[84,126]	[21.92,32.88]
Class 6	6.42%	219	47.2	[175.2,262.8]	[37.73,56.64]

[a] statistics taken from Strum et al. [82].
[b] in minutes.

variations in surgery durations [83, 84]. Then, the mean and standard deviation of S_j are randomly selected within the study mean and study standard deviation ranges. It is assumed that S_j follows the lognormal distribution because it is widely used due to its ability to incorporate the variability inherent in surgery durations [85].

Patient health condition ranges from 0 (high risk of death or irrecoverable health condition) to 100 (normal health condition) in this computational study. The scale for patient health condition may be flexible depending on surgeons or physicians. However, it can be normalized into the 100-scale.

Regarding m (number of available days for surgeries), since it is a possible schedule in which only one surgery is performed for each day, m is set to be n (number of patients).

a_j (critical time point) and b_j (initial patient condition) are generated from the uniform distribution over the intervals [1440 minutes, $2880 \times m$ minutes] and [30, 100] respectively. Since this study does not focus on emergency surgery but on elective surgery, it is supposed that patients can wait, without risk of death or irrecoverable health condition, for at least 1 day (1440 minutes) and initial health condition of each patient is not urgent (not below 30).

Other parameters used for the experiments are set to be as follows: d_l (available time duration on day l) $= 480$ minutes for all l, d'_l (time duration between d_l and d_{l+1}) $= 960$ minutes for all l, and c_l (allowable maximum overtime on day l) $= 480$ minutes for all l.

Table 3.2 demonstrates the trade-off between the computation time of the SAA algorithm and the quality of SAA solutions. The SAA algorithm is tested for 36 cases of 10 instances each. As q and n increase, the average computation time of the SAA algorithm increases exponentially so that about 5 hours are needed to solve a problem with $q = 20$, u (number of SAA replications)$= 10$, and $n = 10$. It is also shown in Table 3.2 that statistically good solutions are obtained by increasing q.

The SAA algorithm with $q = 20$ provides near-optimal solutions. The average 95% CIOOG of SAA solutions with $q = 20$ is 0.38. Since the 95% CIOOG overestimates the optimality gap $\phi(X^*) - \phi(\hat{X})$, and the 100-scale is used to represent patient health condition in the experiments, 0.38 95% CIOOG is regarded as sufficiently small.

Table 3.2
Performance of SAA algorithm.

q	u	n	δ	Average computation time (seconds)	Average 95% CIOOG
5	5	8	0.05	83.649	0.470
			0.1	75.476	0.602
			0.2	60.929	1.736
		10	0.05	1716.883	0.837
			0.1	1217.869	2.015
			0.2	1050.219	3.505
	10	8	0.05	178.120	0.359
			0.1	178.184	0.567
			0.2	147.758	1.462
		10	0.05	2927.354	0.630
			0.1	2061.490	0.870
			0.2	1721.164	2.034
Average	$(q = 5)$			951.591	1.257
10	5	8	0.05	214.532	0.270
			0.1	243.954	0.374
			0.2	206.062	0.761
		10	0.05	3663.980	0.239
			0.1	2373.740	0.413
			0.2	1888.483	0.925
	10	8	0.05	530.693	0.271
			0.1	545.988	0.486
			0.2	498.849	0.864
		10	0.05	6929.815	0.253
			0.1	5480.185	0.459
			0.2	4778.149	1.403
Average	$(q = 10)$			2279.536	0.560
20	5	8	0.05	631.560	0.158
			0.1	593.821	0.277
			0.2	587.390	0.391
		10	0.05	9360.266	0.208
			0.1	8406.003	0.358
			0.2	5935.091	0.946
	10	8	0.05	1436.366	0.142
			0.1	1087.477	0.237
			0.2	1244.482	0.482
		10	0.05	17938.709	0.205
			0.1	15451.887	0.384
			0.2	12308.934	0.773
Average	$(q = 20)$			6248.499	0.380

However, the SAA algorithm can only be used to schedule surgeries that have long durations like transplantation (e.g., mean surgery durations for liver and kidney transplantation are 691 and 328 minutes respectively [82]). Surgery schedules are typically made on a weekly or monthly basis [11, 46, 53]. If more than two surgeries are assignable in a day and, accordingly, more than ten surgeries should be scheduled for a week, the SAA algorithm may not be appropriately used.

3.5 Conclusions

This chapter describes an OR planning problem under uncertain surgery durations and models it as a stochastic MIP. The minimum patient health condition is considered to improve patient safety, and total overtime is considered to reduce a hospital's expenses. The idea of maximizing the minimum patient health condition, which improves critical-patient safety, has not been discussed in previous work.

The SAA algorithm is presented in this chapter. The SAA algorithm is constructed solving the deterministic equivalent model (i.e., SAA model) of the proposed stochastic MIP several times with different sets of scenarios. Even though the SAA algorithm can yield good solutions which are statistically guaranteed, it is costly in terms of the computation time. To make up for the time inefficiency of the SAA algorithm and solve large-size problems, two heuristics are developed in the following chapter.

4. HEURISTIC APPROACHES TO ELECTIVE SURGERY SCHEDULING WITH LINEARLY DETERIORATING PATIENT HEALTH CONDITION

4.1 Introduction

This chapter deals with an elective surgery scheduling problem introduced in Chapter 3. The problem aims to obtain a surgery schedule that provides the optimal trade-off between maximizing the minimum patient health condition and minimizing total overtime of an OR. The minimum patient health condition is the health condition of the most critical patient.

The main purpose of this chapter is to develop solution approaches that are able to solve the problem within reasonable computation times while providing high-quality solutions. This chapter presents two solution approaches: a fastest ascent local search and a tabu search. Those are evaluated with the SAA algorithm.

The reminder of this chapter is organized as follows. The next section recalls the stochastic MIP and SAA model of the problem. Section 4.3 illustrates a fastest ascent local search and a tabu search. The two heuristics are compared with the SAA algorithm in Section 4.4. Section 4.5 concludes this chapter.

4.2 Problem Description

This section recalls the problem introduced in Chapter 3. Notations used in the stochastic MIP and SAA model are summarized in Table 4.1.

The stochastic MIP to formulate the problem is as follows:

$$\max \ E[Y - \delta \sum_{l=1}^{m} O_l] \tag{4.1}$$

$$\text{s.t.} \ \sum_{k=1}^{n} \sum_{l=1}^{m} x_j^{kl} = 1, \qquad j = 1, ..., n \tag{4.2}$$

$$\sum_{j=1}^{n} x_j^{kl} \leq 1, \qquad k = 1, ..., n; l = 1, ...m \tag{4.3}$$

$$\sum_{j=1}^{n} x_j^{kl} \geq \sum_{j=1}^{n} x_j^{k+1,l}, \qquad k = 1, ..., n-1; l = 1, ..., m \tag{4.4}$$

$$\sum_{k=1}^{n} \sum_{j=1}^{n} s_j^{\mu} x_j^{kl} - d_l \leq c_l, \qquad l = 1, ..., m \tag{4.5}$$

$$\sum_{k=1}^{n} \sum_{j=1}^{n} S_j x_j^{kl} - d_l \leq O_l, \qquad l = 1, ..., m \tag{4.6}$$

$$-\frac{b_j}{a_j} \left[\sum_{i=1}^{l-1} \left(d_i + d_i^{'} \right) x_j^{kl} + \sum_{g=1}^{k-1} \sum_{h=1}^{n} S_h x_h^{gl} - M \left(1 - x_j^{kl} \right) \right] + b_j \geq Y,$$

$$j = 1, ..., n; k = 1, ..., n; l = 1, ..., m \tag{4.7}$$

$$x_j^{kl} \in \{0, 1\}, \qquad j = 1, ..., n; k = 1, ..., n; l = 1, ..., m \tag{4.8}$$

$$Y \ \text{free} \tag{4.9}$$

$$O_l \geq 0, \qquad l = 1, ..., m \tag{4.10}$$

The SAA model of the stochastic MIP is as follows:

$$\max \ \frac{1}{q} \sum_{r=1}^{q} \left[y(\omega_r) - \delta \sum_{l=1}^{m} o_l(\omega_r) \right] \tag{4.11}$$

$$\text{s.t.} \ \sum_{k=1}^{n} \sum_{l=1}^{m} x_j^{kl} = 1, \qquad j = 1, ..., n \tag{4.12}$$

$$\sum_{j=1}^{n} x_j^{kl} \leq 1, \qquad k=1,\ldots,n; l=1,\ldots,m \tag{4.13}$$

$$\sum_{j=1}^{n} x_j^{kl} \geq \sum_{j=1}^{n} x_j^{k+1,l}, \qquad k=1,\ldots,n-1; l=1,\ldots,m \tag{4.14}$$

$$\sum_{k=1}^{n}\sum_{j=1}^{n} s_j^{\mu} x_j^{kl} - d_l \leq c_l, \qquad l=1,\ldots,m \tag{4.15}$$

$$\sum_{k=1}^{n}\sum_{j=1}^{n} s_j(\omega_r) x_j^{kl} - d_l \leq o_l(\omega_r), \qquad l=1,\ldots,m; r=1,\ldots,q \tag{4.16}$$

$$-\frac{b_j}{a_j}\left[\sum_{i=1}^{l-1}\left(d_i + d_i^{'}\right)x_j^{kl} + \sum_{g=1}^{k-1}\sum_{h=1}^{n} s_h(\omega_r)x_h^{gl} - M\left(1 - x_j^{kl}\right)\right] + b_j \geq y(\omega_r),$$
$$j=1,\ldots,n; k=1,\ldots,n; l=1,\ldots,m; r=1,\ldots,q \tag{4.17}$$

$$x_j^{kl} \in \{0,1\}, \qquad j=1,\ldots,n; k=1,\ldots,n; l=1,\ldots,m \tag{4.18}$$

$$y(\omega_r) \ \text{free} \qquad r=1,\ldots,q \tag{4.19}$$

$$o_l(\omega_r) \geq 0, \qquad l=1,\ldots,m; r=1,\ldots,q \tag{4.20}$$

4.3 Heuristic Approaches

A fastest ascent local search (FALS) and a tabu search (TS) are developed in this section considering the trade-off between the efficiency of computational time and the effectiveness of solutions. Sections 4.3.1, 4.3.2, and 4.3.3 illustrate key structural elements used in both the FALS and TS. The overall procedures of the two heuristics are described in Sections 4.3.4 and 4.3.5 respectively.

4.3.1 Feasible Solution Acquisition

The heuristics use the optimal solution of the expected value model (EVM) for their initial solutions. The EVM is obtained by replacing random parameters Y, O_l, and S_j with

Table 4.1
Notations used in stochastic MIP and SAA model

n	number of patients to be scheduled for surgeries
m	number of available days for surgeries
j	patient index, $j \in J$, $J = \{1, ..., n\}$
k	surgery sequence index on a day, $k \in K$, $K = \{1, ..., n\}$
l	day index, $l \in L$, $L = \{1, ..., m\}$
S_j	random surgery duration for patient j
s_j^{μ}	mean surgery duration of patient j
a_j	critical time point by which the surgery for patient j should be performed
b_j	initial health condition of patient j at the beginning of day 1
d_l	available time duration of an OR on day l in which surgeries can be performed without overtime
d_l'	time duration (unavailable for surgeries) between d_l and d_{l+1}
c_l	allowable maximum overtime on day l
O_l	overtime on day l which exceeds the available time duration d_l
Y	minimum patient health condition
M	sufficiently large number
δ	coefficient between Y and $\sum_{l=1}^{m} O_l$
q	number of scenarios
ω_r	rth scenario that defines the rth realization of the random surgery duration vector $\vec{S} = (S_1, S_2, ..., S_n)$, $r = 1, ..., q$
$s_j(\omega_r)$	element of scenario ω_r that defines the surgery duration of patient j
$y(\omega_r)$	realization of Y
$o_l(\omega_r)$	realization of O_l
x_j^{kl}	1 if patient j is assigned to the kth surgery on day l. 0 otherwise.

deterministic parameters y, o_l, and s_j^μ respectively in the stochastic MIP. Note that s_j^μ is the expected value of S_j.

4.3.2 Solution Evaluation

The heuristics generate q scenarios with which they estimate the objective value of the stochastic MIP for a given solution $\hat{X}$. Let $T_j(\hat{X},\vec{S})$ be the surgery start time of patient j in $\hat{X}$ under $\vec{S}$, $t_j(\hat{X},\omega_r)$ be the surgery start time of patient j in $\hat{X}$ under ω_r, and $\vec{s^\mu}$ be the mean surgery duration vector $(s_1^\mu, s_2^\mu, \ldots, s_n^\mu)$. The objective value of $\hat{X}$ is the weighted sum of two components: $E[Y(\hat{X},\vec{S})]$ and $E[\sum_{l=1}^m O_l(\hat{X},\vec{S})]$.

$E[Y(\hat{X},\vec{S})]$ can be written as

$$E\left[\min_j\left\{-\frac{b_j}{a_j}T_j\left(\hat{X},\vec{S}\right)+b_j\right\}\right] \tag{4.21}$$

where

$$T_j(\hat{X},\vec{S}) := \max_{k,l}\left\{\sum_{i=1}^{l-1}\left(d_i+d_i^{'}\right)\hat{x}_j^{kl}+\sum_{g=1}^{k-1}\sum_{h=1}^{n}S_h\hat{x}_h^{gl}-M\left(1-\hat{x}_j^{kl}\right)\right\}.$$

An estimator of (4.21), which is used in Section 4.2, is

$$\frac{1}{q}\sum_{r=1}^{q}\min_j\{-\frac{b_j}{a_j}t_j(\hat{X},\omega_r)+b_j\} \tag{4.22}$$

where

$$t_j(\hat{X},\omega_r) := \max_{k,l}\left\{\sum_{i=1}^{l-1}\left(d_i+d_i^{'}\right)\hat{x}_j^{kl}+\sum_{g=1}^{k-1}\sum_{h=1}^{n}s_h(\omega_r)\hat{x}_h^{gl}-M\left(1-\hat{x}_j^{kl}\right)\right\}.$$

(4.22) is used in the SAA model since it is an unbiased estimator of (4.21). However, the computational load for (4.22) is heavy so that another estimator is used in the heuristics. The heuristics estimate (4.21) by

$$\min_{j}\left\{-\frac{b_j}{a_j}t_j(\hat{X},\vec{s^{\mu}})+b_j\right\} \tag{4.23}$$

where

$$t_j(\hat{X},\vec{s^{\mu}}) := \max_{k,l}\left\{\sum_{i=1}^{l-1}\left(d_i+d_i^{'}\right)\hat{x}_j^{kl}+\sum_{g=1}^{k-1}\sum_{h=1}^{n}s_h^{\mu}\hat{x}_h^{gl}-M\left(1-\hat{x}_j^{kl}\right)\right\}.$$

The heuristics reduce their computation time by a factor q as they calculate (4.23), instead of (4.22), for a given solution. In addition, in most cases of SAA solutions, (4.23) is equal or close to (4.22). 360 SAA solutions, which are obtained respectively for 360 instances in Section 4.4, are analyzed to test the value difference between (4.23) and (4.22). The lowest value for patient health condition is set to be 0, which indicates death or irrecoverable health condition. On the other hand, the highest one is set to be 100 to represent normal health condition. Refer to Section 4.4 for the setting of the other parameters. (4.23) is equal to (4.22) in 344 out of 360 SAA solutions. If a patient assigned to the first surgery of any available day is the most critical patient in an SAA solution for any given q scenarios and $\vec{s^{\mu}}$, it is easy to show that (4.23) is equal to (4.22) in the SAA solution. Those 344 SAA solutions belong to this category. Even though (4.23) is not equal to (4.22) in 16 out of 360 SAA solutions, the average value difference between (4.23) and (4.22) is just 0.097 which is less than 0.1% of the range for patient health condition.

Let $o_l(\hat{X},\omega_r)$ be overtime on day l in $\hat{X}$ under ω_r. The heuristics estimate $E[\sum_{l=1}^{m}O_l(\hat{X},\vec{S})]$, with q scenarios, by

$$\frac{1}{q}\sum_{r=1}^{q}\sum_{l=1}^{m}o_l\left(\hat{X},\omega_r\right) \tag{4.24}$$

where

$$o_l(\hat{X},\omega_r) := \max\left\{0,\sum_{k=1}^{n}\sum_{j=1}^{n}s_j(\omega_r)\hat{x}_j^{kl}-d_l\right\}.$$

Note that (4.24) is an unbiased estimator of $E[\sum_{l=1}^{m} O_l(\hat{X},\vec{S})]$ and the same estimator used in the SAA model.

To sum up, the heuristics evaluate a given solution $\hat{X} = \{\hat{x}_j^{kl} | j \in J, k \in K, l \in L\}$, with q scenarios, by

$$\min_j \left\{ -\frac{b_j}{a_j} t_j(\hat{X},\vec{s^{\mu}}) + b_j \right\} - \delta \frac{1}{q} \sum_{r=1}^{q} \sum_{l=1}^{m} o_l(\hat{X},\omega_r) \tag{4.25}$$

where

$$t_j(\hat{X},\vec{s^{\mu}}) := \max_{k,l} \left\{ \sum_{i=1}^{l-1} \left(d_i + d_i^{'} \right) \hat{x}_j^{kl} + \sum_{g=1}^{k-1} \sum_{h=1}^{n} s_h^{\mu} \hat{x}_h^{gl} - M \left(1 - \hat{x}_j^{kl} \right) \right\},$$
$$o_l(\hat{X},\omega_r) := \max \left\{ 0, \sum_{k=1}^{n} \sum_{j=1}^{n} s_j(\omega_r) \hat{x}_j^{kl} - d_l \right\}.$$

4.3.3 Neighborhood Structure

Once an initial solution, denoted by ${}_{current}\hat{x}$, is obtained, the heuristics start to search for neighbor solutions based on the pairwise interchange mechanism [86]. Let ${}_{temp}\hat{x}$ be a temporary solution. The heuristics put ${}_{current}\hat{x}$ into ${}_{temp}\hat{x}$. The surgery assignments for a pair of patients are interchanged in ${}_{temp}\hat{x}$. Note that all the other surgery assignments except for both interchanged surgery assignments in the current iteration remain the same in ${}_{temp}\hat{x}$. If total mean duration of surgeries for each day fits within the time available for each day in ${}_{temp}\hat{x}$, ${}_{temp}\hat{x}$ is a feasible neighbor solution so that it is considered for a move from ${}_{current}\hat{x}$. However, this mechanism does not change the number of surgeries scheduled for each day. Therefore, the heuristics add a dummy patient's surgery to the last surgery scheduled for each day, and make the assignments of a patient's surgery and a dummy patient's surgery interchangeable.

To sum up, the neighborhood of the heuristics is all pairwise interchanges of surgery assignments including dummy patient surgery assignments.

4.3.4 Fastest Ascent Local Search

The FALS (also called a first improvement local search) explores neighbor solutions at a current solution. Once the FALS finds a neighbor solution whose objective value (4.25) is better than the current solution's objective value (4.25), a move form the current solution to the neighbor solution is made. This process continues until any neighbor solutions cannot improve the objective value (4.25).

This study does not illustrate a steepest ascent local search (also called a best improvement local search) which is an alternate strategy for exploring neighbor solutions [87] since it can be seen as a part of the TS illustrated in Section 4.3.5. A steepest ascent local search selects the best improvement in the entire neighborhood at each move while the TS selects the best neighbor solution regardless of whether the best neighbor solution improves the objective value (4.25) or not. It is obvious that TS solutions are at least as good as steepest ascent local search solutions.

Procedure of the Fastest Ascent Local Search

Step 1. Choose q. Solve the EVM and set ${}_{current}\hat{x} :=$ EVM's optimal solution, $f({}_{current}\hat{x}) :=$ objective value (4.25) of ${}_{current}\hat{x}$. Generate q scenarios.

Step 2. Add a dummy patient's surgery (surgery duration:=0) to the last surgery scheduled for each day in ${}_{current}\hat{x}$.

Step 3. Set $k' := 1, l' := 1$.

Step 4. If $l' \leq m$, go to step 5. Otherwise, terminate the algorithm.

Step 5. If $k' \leq$ number of surgeries scheduled on day l', go to step 6. Otherwise, set $k' := 1, l' := l' + 1$, and go to step 4.

Step 6. Find patient index j' such that ${x_{j'}}^{k'l'} = 1$.

Step 7. Set $k'' := k' + 1, l'' := l'$.

Step 8. If $l'' \leq m$, go to step 9. Otherwise, set $k' := k' + 1$ and go to step 5.

Step 9. If $k'' \leq$ number of surgeries scheduled on day l'', go to step 10. Otherwise, set $k'' := 1, l'' := l'' + 1$, and go to step 8.

Step 10. Find patient index j'' such that $x_{j''}{}^{k''l''} = 1$.

Step 11. If patient j' and patient j'' are all dummy patients, set $k'' := k'' + 1$ and go to step 9. Otherwise, go to step 12.

Step 12. Set ${}_{temp}\hat{x} := {}_{current}\hat{x}$. Interchange the assignments of patient j''s surgery and patient j'''s surgery in ${}_{temp}\hat{x}$. If the sums of mean surgery durations for l' and l'' fit within the available time durations $d_{l'}$ and $d_{l''}$ respectively in ${}_{temp}\hat{x}$, go to step 13. Otherwise, set $k'' := k'' + 1$ and go to step 9.

Step 13. Set $f({}_{temp}\hat{x}) :=$ objective value (4.25) of ${}_{temp}\hat{x}$. If $f({}_{temp}\hat{x}) > f({}_{current}\hat{x})$, set ${}_{current}\hat{x} := {}_{temp}\hat{x}, f({}_{current}\hat{x}) := f({}_{temp}\hat{x})$, erase all dummy patients' surgeries in ${}_{current}\hat{x}$, and go to step 2. Otherwise, set $k'' := k'' + 1$ and go to step 9.

4.3.5 Tabu Search

TS is a metaheuristic that escapes from a local optimum by allowing non-improving moves and forbidding cycling moves [88–90]. Even though the FALS provides its solutions fast, the FALS terminates its algorithm when it reaches a local optimum (see Figure 4.1. A TS is proposed in this section to continue the search beyond local optimality [91] (see 4.2).

A move is made from a current solution to its best neighbor solution in the entire neighborhood at each iteration in the TS. The best neighbor solution does not have to be better than the current solution for the move.

Tabu list is used to prevent the algorithm from cycling back to previously visited solutions. The tabu list is a first-in-first-out (FIFO) queue with length γ (this length is called tabu tenure) and each element of the tabu list is a pair of patient indices. At each move, the TS puts both indices of patients, whose surgery assignments are interchanged in the move,

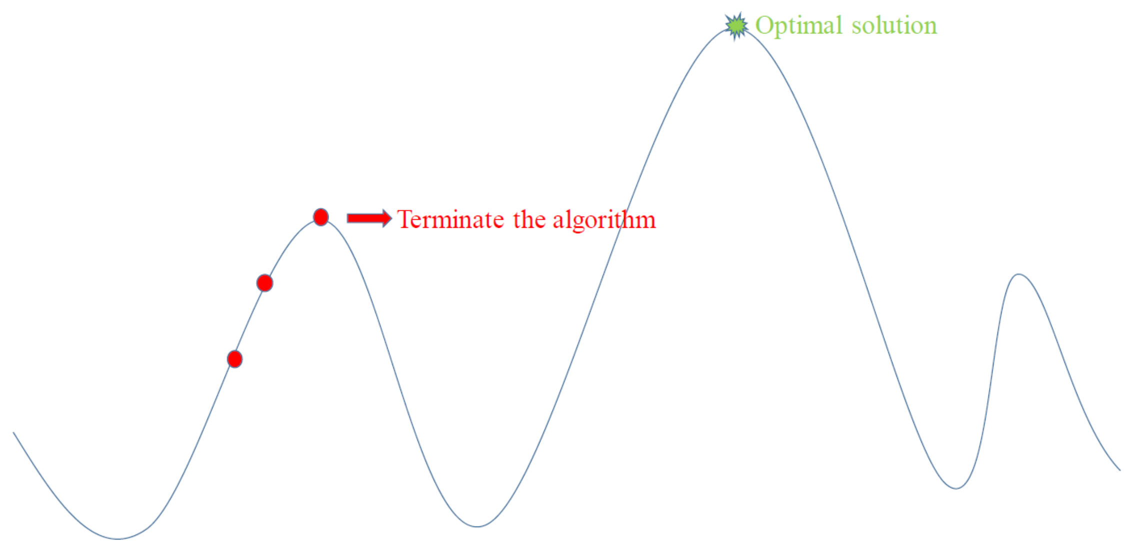


Figure 4.1. How FALS obtains its solution.

into the tabu list as a pair. Reversing the pair's surgery assignments is not considered during γ moves.

To efficiently search for solutions within reasonable computation times, this paper introduces search intensity. Search intensity is defined as the probability a neighbor solution is evaluated for a move. The TS controls the search intensity as follows:

- When the algorithm starts with an initial solution, the search intensity is 1.
- If the objective value (4.25) of the best neighbor solution is better than that of the current solution, the search intensity for the next move becomes 1.
- If the objective value (4.25) of the best neighbor solution is worse than or equal to that of the current solution, the search intensity for the next move decreases by Δ.
- When the search intensity becomes 0, the algorithm terminates.

The idea behind the concept of the search intensity is that the algorithm should examine "promising" [92] areas thoroughly not to miss the best solutions in those areas while the algorithm skims non-promising areas to reduce the computational burden.

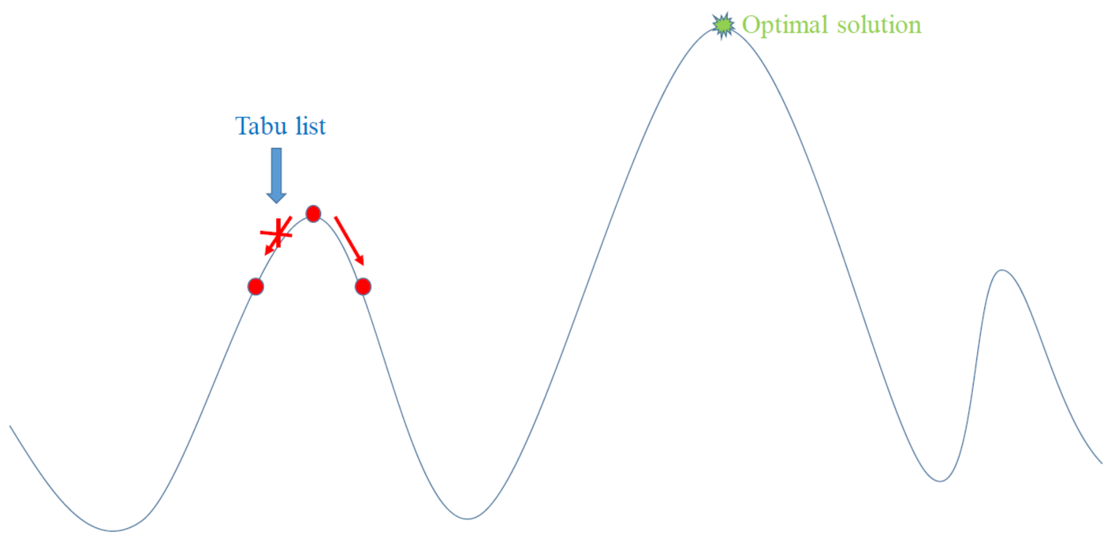


Figure 4.2. How TS escapes from a local optimum.

Procedure of the Tabu Search

Step 1. Choose γ, Δ, q. Solve the EVM and set ${}_{current}\hat{x} :=$ EVM's optimal solution, $f({}_{current}\hat{x}) :=$ objective value (4.25) of ${}_{current}\hat{x}$, tabu list := $\varnothing$, search intensity := 1. Generate q scenarios.

Step 2. Set ${}_{best}\hat{x} :={}_{current} \hat{x}$, $f({}_{best}\hat{x}) := f({}_{current}\hat{x})$.

Step 3. Add a dummy patient's surgery (surgery duration:=0) to the last surgery scheduled for each day in ${}_{current}\hat{x}$. Set $f({}_{bestneighbor}\hat{x}) := -M$ (note that M is a sufficiently large number).

Step 4. Set $k' := 1, l' := 1$.

Step 5. If $l' \leq m$, go to step 6. Otherwise, go to step 15.

Step 6. If $k' \leq$ number of surgeries scheduled on day l', go to step 7. Otherwise, set $k' := 1, l' := l' + 1$, and go to step 5.

Step 7. Find patient index j' such that ${x_{j'}}^{k'l'} = 1$.

Step 8. Set $k'' := k' + 1, l'' := l'$.

Step 9. If $l'' \leq m$, go to step 10. Otherwise, set $k' := k' + 1$ and go to step 6.

Step 10. If $k'' \leq$ number of surgeries scheduled on day l'', go to step 11. Otherwise, set $k'' := 1, l'' := l'' + 1$, and go to step 9.

Step 11. Find patient index j'' such that $x_{j''}{}^{k''l''} = 1$.

Step 12. Generate a real number β from the uniform distribution over the interval $[0,1]$. If patient j' and patient j'' are all dummy patients, $\beta >$ search intensity, or (j', j'') is an element of tabu list, set $k'' := k'' + 1$ and go to step 10. Otherwise, go to step 13.

Step 13. Set ${}_{temp}\hat{x} := {}_{current}\hat{x}$. Interchange the assignments of patient j''s surgery and patient j'''s surgery in ${}_{temp}\hat{x}$. If the sums of mean surgery durations for days l' and l'' fit within the available time durations $d_{l'}$ and $d_{l''}$ respectively in ${}_{temp}\hat{x}$, go to step 14. Otherwise, set $k'' := k'' + 1$ and go to step 10.

Step 14. Set $f({}_{temp}\hat{x}) :=$ objective value (4.25) of ${}_{temp}\hat{x}$. If $f({}_{temp}\hat{x}) > f({}_{bestneighbor}\hat{x})$, set ${}_{bestneighbor}\hat{x} := {}_{temp}\hat{x}, f({}_{bestneighbor}\hat{x}) := f({}_{temp}\hat{x}), k'' := k'' + 1$, tabu candidate := (j'', j'), erase all dummy patients' surgeries in ${}_{bestneighbor}\hat{x}$, and go to step 10. Otherwise, set $k'' := k'' + 1$ and go to step 10.

Step 15. If $f({}_{bestneighbor}\hat{x}) > f({}_{current}\hat{x})$, set search intensity := 1. Otherwise, search intensity := search intensity $- \Delta$.

Step 16. If search intensity $= 0$, terminate the algorithm. Otherwise, set ${}_{current}\hat{x} := {}_{bestneighbor}\hat{x}, f({}_{current}\hat{x}) := f({}_{bestneighbor}\hat{x})$, put tabu candidate into tabu list, and go to step 17.

Step 17. If $f({}_{current}\hat{x}) > f({}_{best}\hat{x})$, go to step 2. Otherwise, go to step 3.

4.4 Computational Study

The numerical experiments to test the heuristics described in Section 4.3 are presented in this section. Parameters are set to be the same as those in Section 3.4. Please refer to 3.4 to see the setting of parameters. The heuristics and the SAA algorithm are implemented in the General Algebraic Modeling System (GAMS) 24.1.3 with CPLEX 12.5.1 for solving MIPs on a PC with a 2.4GHz Core i7 processor and 8GB RAM.

A set of scenarios is used to obtain an SAA solution or a heuristic solution, and a different set of 200 scenarios is used to evaluate the obtained solution $\hat{X} = \{\hat{x}_j^{kl} | j \in J, k \in K, l \in L\}$. $\hat{X}$ is evaluated by

$$\frac{1}{200}\sum_{r=1}^{200} \min_j \left\{ -\frac{b_j}{a_j} \max_{k,l} \left\{ \sum_{i=1}^{l-1} \left(d_i + d_i^{'}\right) \hat{x}_j^{kl} + \sum_{g=1}^{k-1}\sum_{h=1}^{n} s_h(\omega_r)\hat{x}_h^{gl} - M\left(1 - \hat{x}_j^{kl}\right) \right\} + b_j \right\}$$
$$- \delta \frac{1}{200}\sum_{r=1}^{200}\sum_{l=1}^{m} \max\left\{0, \sum_{k=1}^{n}\sum_{j=1}^{n} s_j(\omega_r)\hat{x}_j^{kl} - d_l\right\}. \quad (4.26)$$

Note that (4.26) is the SAA objective value (4.11) of $\hat{X}$ with 200 scenarios. It is supposed in this computational study that when solutions are compared, the solutions are obtained for the same instance (i.e., the same number of patients , the same critical time point and initial patient condition for each patient, the same coefficient δ, and the same distribution for each patient's surgery duration), and the objective values (4.26) of the solutions are obtained using the same test set of 200 scenarios.

The performance of the SAA algorithm with $q = 20$ and $u = 10$ is compared to that of each heuristic since the SAA algorithm with $q = 20$ and $u = 10$ provides the best 95% CIOOG on average. Since the SAA algorithm uses 200 scenarios in total (i.e., $20(q)\times 10(u)$) to solve an instance for comparison, the FALS and TS use the same set of 200 scenarios to solve the instance for the sake of fairness.

Table 4.2 demonstrates the effects of γ (tabu tenure) and Δ (stepsize to decrease the search intensity) on the quality of TS solutions. 10 instances with $n = 20$ and $\delta = 0.1$ are solved by the TS varying γ and Δ. It is easy to find that, in general, the selection of γ and Δ

Table 4.2
Objective values (4.26) of TS solutions for 10 instances with $n = 20$ and $\delta = 0.1$.

		Objective value (4.26) of TS solution for each instance										
γ	Δ	1	2	3	4	5	6	7	8	9	10	Average
	0.05	-17.319	-97.368	-68.568	-17.986	-176.208	-130.224	-159.328	-42.619	-129.218	-164.882	-100.372
3	0.1	-18.181	-97.368	-68.568	-22.082	-177.725	-130.224	-159.328	-42.619	-129.294	-164.882	-101.027
	0.2	-18.181	-97.368	-68.568	-22.082	-179.817	-130.224	-159.328	-42.619	-129.294	-164.882	-101.236
	0.05	-17.421	-97.399	-68.427	-17.986	-176.297	-130.224	-159.328	-42.619	-129.218	-164.882	-100.380
5	0.1	-17.691	-97.368	-68.568	-22.082	-179.817	-130.224	-159.328	-42.619	-129.294	-164.882	-101.187
	0.2	-18.181	-97.368	-68.568	-22.082	-179.817	-130.224	-159.328	-42.619	-129.294	-164.882	-101.236
	0.05	-17.420	-97.368	-68.568	-20.935	-179.817	-130.224	-156.604	-42.619	-129.218	-164.882	-100.766
7	0.1	-17.319	-97.368	-68.568	-22.082	-179.817	-130.224	-156.604	-42.619	-129.294	-164.882	-100.878
	0.2	-18.181	-97.368	-68.427	-22.082	-179.817	-130.224	-159.328	-41.848	-129.294	-164.882	-101.145

Table 4.3
Computation times of EVM, FALS, and TS.

n	δ	Average computation time (seconds)		
		EVM	FALS	TS
	0.05	1.045	0.878	1130.531
8	0.1	1.359	0.830	902.425
	0.2	1.742	1.026	1263.671
Average	($n = 8$)	1.382	0.911	1098.876
	0.05	6.213	2.079	2131.505
10	0.1	5.473	2.224	1585.871
	0.2	4.959	2.356	3157.691
Average	($n = 10$)	5.548	2.219	2291.689
	0.05	166.805	21.960	1177.355
15	0.1	279.340	23.290	636.701
	0.2	303.600	20.902	600.893
Average	($n = 15$)	249.915	22.051	804.983
	0.05	5673.867	163.340	1683.541
20	0.1	3403.288	178.201	1749.867
	0.2	3723.531	171.479	1320.860
Average	($n = 20$)	4266.895	171.007	1584.756

does not significantly improve or worsen the quality of TS solutions. For comparison with the other algorithms, the combination of $\gamma = 3$ and $\Delta = 0.05$ is selected for the TS since it gives a slightly better average objective value (4.26) than the other combinations. Note that γ and Δ are not used in the FALS and, therefore, a preliminary experiment to select the parameters is not needed for the FALS.

This research presents the performances of the FALS and TS up to $n = 20$. Surgery schedules are typically made on a weekly or monthly basis [11, 46, 53]. Therefore, it is assumed that solutions (i.e., surgery schedules) should be, to be practical, at least for a week. In the numerical experiments for instances with $n = 20$, FALS and TS solutions always exceed the surgery capacity for 5 days. Therefore, $n = 20$ is considered large enough in practice.

Table 4.3 illustrates the FALS and TS provide their solutions for large-size problems within reasonable computation times. They were tested for 12 cases of 10 instances each. The average computation times to solve the EVM and to implement each heuristic after

Table 4.4
Average objective value (4.26) and average difference of objective values (4.26).

		Average objective value (4.26)				Average difference of objective values (4.26)					
n	δ	SAA	TS	FALS	EVM	SAA-TS	SAA-FALS	SAA-EVM	TS-FALS	TS-EVM	FALS-EVM
8	0.05	15.240	15.278	15.202	14.515	-0.037	0.038	0.725	0.075	0.762	0.687
	0.1	-2.990	-2.997	-3.527	-4.732	0.007	0.537	1.742	0.530	1.735	1.205
	0.2	-39.455	-39.455	-39.495	-43.665	0.000	0.041	4.210	0.040	4.210	4.170
10	0.05	9.604	9.611	9.486	8.982	-0.007	0.119	0.622	0.126	0.629	0.503
	0.1	-13.605	-13.607	-13.928	-15.104	0.002	0.323	1.499	0.322	1.497	1.176
	0.2	-59.370	-59.390	-60.493	-64.015	0.020	1.123	4.645	1.103	4.625	3.522
15	0.05	N/A	-18.558	-18.873	-20.163	N/A	N/A	N/A	0.315	1.605	1.289
	0.1		-69.920	-70.769	-73.359				0.849	3.439	2.590
	0.2		-171.848	-174.597	-179.371				2.749	7.523	4.774
20	0.05	N/A	-43.120	-43.451	-44.919	N/A	N/A	N/A	0.330	1.799	1.468
	0.1		-100.372	-101.419	-104.735				1.047	4.363	3.316
	0.2		-214.162	-215.196	-222.531				1.033	8.369	7.335
					Average	-0.003	0.363	2.241	0.710	3.380	2.670

Table 4.5
P-value of paired *t*-test (two-tailed) for the mean difference of objective values (4.26)

	SAA	TS	FALS	EVM
SAA	-	0.726 ($df^{\dagger} = 59$)	0.003*($df^{\dagger} = 59$)	<0.001*($df^{\dagger} = 59$)
TS	-	-	<0.001*($df^{\dagger} = 119$)	<0.001*($df^{\dagger} = 119$)
FALS	-	-	-	<0.001*($df^{\dagger} = 119$)
EVM	-	-	-	-

* significant mean difference of objective values.
† degree of freedom.

obtaining an initial solution are summarized. As n increases, the computation time to obtain an initial solution by solving the EVM increases exponentially so that it limits the FALS and TS to solve large-size problems. However, an EVM solution serves as a lower-bound solution of the FALS and TS. One thing that should be noticed in this table is that the computation time of the TS is not very sensitive to the problem size due to the search intensity mechanism. To solve an instance with $n = 20$, the FALS and TS spend less than 2 hours on average including the computation time to obtain an initial solution, while the SAA algorithm cannot complete its procedure within a computation time limit of 12 hours for any instances with $n = 15$ or 20.

Tables 4.4 and 4.5 illustrate the qualities of SAA, TS, FALS, and EVM solutions. EVM solutions are used as benchmarks for the other algorithms' solutions [93]. Table 4.4 shows the average objective value (4.26) of each algorithm for 12 cases of 10 instances each and the average difference of the objective values (4.26). In addition, the result of the two-tailed paired t-test [94], to check the statistical significance for the mean difference of the objective values (4.26), is described in Table 4.5.

The qualitative difference between TS and SAA solutions is not statistically significant. Note that those SAA solutions are near-optimal but time-consuming to be obtained. The TS remarkably improves its initial solutions with much shorter computation time than the SAA algorithm's computation time. The mean objective values (4.26) of SAA and TS solutions are significantly better than that of FALS solutions. The mean objective value (4.26) of FALS solutions is significantly better than that of EVM solutions. The solution improvement procedure of the FALS, from its initial solutions, is successful but not as effective as that of the TS.

4.5 Conclusions

This chapter presents two heuristics (i.e., FALS and TS) for an elective surgery scheduling problem with linearly deteriorating patient health condition.

The FALS provides its solutions faster than the TS. However, the quality of FALS solutions is not as good as that of TS solutions. The computational study demonstrates, through the comparison with the SAA algorithm, that the TS provides effective solutions within reasonable computation times.

5. ELECTIVE SURGERY SCHEDULING WITH STEP-DETERIORATING PATIENT HEALTH CONDITION

5.1 Introduction

Hospitals and healthcare organizations have paid increased attention to improving patient safety in recent decades [1]. However, most of approaches to improve patient safety focus on the reporting and prevention of medical errors, [2, 3, 95], rather than on the managerial aspect of a healthcare system.

This research is motivated by the fact that surgery scheduling considering patient health condition can contribute to improving patient safety. Disease exacerbates patient health condition with respect to waiting time for surgery. Surgeons and patients may want to schedule their surgeries as early as possible in order to escape from the patients' risks of deaths or turning the current diseases into more severe diseases. However, the resource limitation on surgeons, anesthesiologists, nurses, operating rooms (ORs), post-anesthesia care units, etc., forces surgical schedulers to prioritize surgeries. Patient health condition may be a key factor to consider in prioritizing and scheduling surgeries.

This study considers time-dependent patient health condition. Whenever patients are diagnosed by practitioners, patient health condition may be able to be recorded using severity level measures such as Kanofsky grade [77], dyspnea index [76], and model for end-stage liver disease score [78]). Karnofsky grade is given in Table 5.1 as an example of severity level measures for diseases. Those records can be grouped by diseases, ages, genders, and/or disease histories at practitioners' discretion. This study assumes to use those records to represent patient health condition.

This study aims to provide the optimal solution to maximize the minimum patient health condition. the minimum patient health condition is defined as the health condition of the most critical patient. The objective of maximizing the minimum patient health condition allows the surgery schedule to minimize the possibility of sentinel events.

The remainder of this chapter is organized as follows. Section 5.2 describes the problem and formulates it as a stochastic MIP. In Section 5.3, an SAA algorithm is presented. Section 5.4 presents the performance of the SAA algorithm and the analyses on the SAA solutions. Finally, concluding remarks are made in Section 5.5.

5.2 Problem Description

A set of patients $J = \{1, \ldots, n\}$ of a surgeon or a surgical group is scheduled for elective surgeries in a given block schedule under a full-day block system, a half-day block system,

Table 5.1
An example of severity level measures for diseases.

Karnofsky Status	**Karnofsky Grade**
Normal, no complaints	100
Able to carry on normal activities. Minor signs or symptoms of disease	90
Normal activity with effort	80
Care for self. Unable to carry on normal activity or to do active work	70
Requires occasional assistance, but able to care for most of his needs	60
Requires considerable assistance and frequent medical care	50
Disabled. Requires special care and assistance	40
Severly disabled. Hospitalisation indicated though death nonimminent	30
Very sick. Hospitalisation necessary. Active supportive treatment necessary	20
Moribund	10
Dead	0

or a hybrid system of full-day and half-day blocks. See Figure 5.1 for an example of each block system. There are a set of available days $L = \{1,\ldots,m\}$ for surgeries and a set of the days $L_b \subset L$ in which surgical blocks of the surgeon or the surgical group exist. Each day $l \in L_b$ has one or two surgical blocks (Being I the set of surgical blocks on a day) in which surgeries by the surgeon or the surgical group can be performed.

The surgery for patient $j \in J$ is assigned to block $i \in I$ on day $l \in L_b$. Therefore, the surgery assignment decision variable is as follows:

$$x_j^{il} = \begin{cases} 1, & \text{if the surgery for patient } j \text{ is assigned to block } i \text{ on day } l \\ 0, & \text{otherwise} \quad (i \in I,\ j \in J, l \in L_b) \end{cases}$$

Each patient j is characterized by its mean surgery duration s_j^{μ} and its random health condition, denoted by H_{jl}, on day l.

Let d_{il} be available time duration of block i on day l in which surgeries of the surgeon or the surgical group can be performed without overtime. d_{il} can be manipulated depending on the block system used. If a full-day block system is used, d_{1l} should be non-zero and d_{2l} should be zero. If a half-day block system is used, both d_{1l} and d_{2l} should be non-zero. Obviously, depending on day l, the two different block systems can be implemented alternately (i.e., hybrid block system).

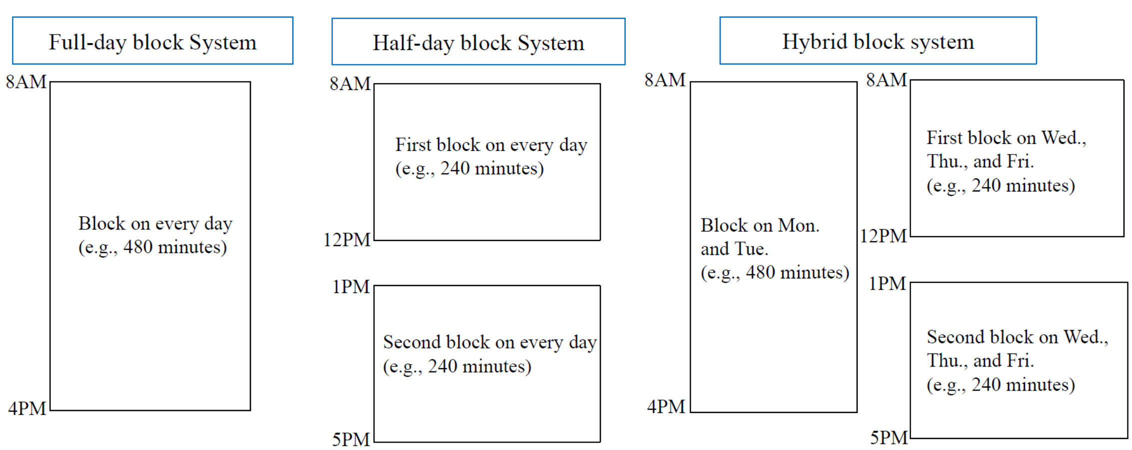


Figure 5.1. Examples of surgical block systems.

Since a surgical block may not be continuously run over the specified overtime range due to maintenance, staffing, or its next block's schedule, allowable maximum overtime of block i on day l, denoted by c_{il}, is introduced.

The objective function of the problem is to maximize the minimum patient health condition. Given a feasible schedule $X = \{x_j^{il} | i \in I, j \in J, l \in L_b\}$, the health condition of patient j is calculated by $\sum_{l \in L_b} H_{jl} \sum_{i \in I} x_j^{il}$. Then, the minimum patient health condition, denoted by Y, is $\min_j \{\sum_{l \in L_b} H_{jl} \sum_{i \in I} x_j^{il}\}$. Since the uncertainty in patient health condition is considered in the problem, the expected value of minimum patient health condition, $E[Y]$ is maximized in the objective function.

Note that the surgery sequence on a day does not affect the objective function. Surgeons can determine the surgery sequence on a day based on their preferences.

A stochastic MIP to formulate the problem of this study is proposed as follows:

$$\max \quad E[Y] \tag{5.1}$$

$$\text{s.t.} \sum_{i \in I} \sum_{l \in L_b} x_j^{il} = 1, \qquad \forall j \in J \tag{5.2}$$

$$\sum_{j \in J} s_j^{\mu} x_j^{il} - d_{il} \leq c_{il}, \qquad \forall i \in I, \forall l \in L_b \tag{5.3}$$

$$\sum_{l \in L_b} H_{jl} \sum_{i \in I} x_j^{il} \geq Y, \qquad \forall j \in J \tag{5.4}$$

$$x_j^{il} \in \{0, 1\}, \qquad \forall i \in I, \forall j \in J, \forall l \in L_b \tag{5.5}$$

$$Y \quad \text{free} \tag{5.6}$$

Since the left-hand side of constraint (5.4) forces Y to be $\min_j \{\sum_{l \in L_b} H_{jl} \sum_{i \in I} x_j^{il}\}$, Y in the objective function (5.1) is a variable for the minimum patient health condition. Constraint (5.2) ensures that every patient should be scheduled. Constraints (5.3) states that surgeries are able to be assigned to a block only if total mean durations of surgeries fit within the available time of the block for surgeries. This constraint is used to limit the overtime of the block [11, 13].

5.3 Sample Average Approximation

In this section, an SAA algorithm is developed referring to Kleywegt et al. [16] and Shapiro [96]. An SAA algorithm is constructed solving an SAA model several times changing the sets of scenarios for the SAA model.

Let q be the number of scenarios for an SAA model. Let ω_r be the rth scenario that defines the rth realization of patient health condition vector $\vec{H} = (H_{11}, \dots, H_{1m}, H_{21}, \dots, H_{2m}, \dots, H_{nm})$, $r \in R$, $R = \{1, \dots, q\}$ and $h_{il}(\omega_r)$ be the element of ω_r that defines the health condition of patient j on day l. Let $y(\omega_r)$ be the realization of Y under scenario ω_r. Note that $\frac{1}{q}\sum_{r\in R} y(\omega_r)$ in the following SAA model approximates $E[Y]$ in the stochastic MIP.

The SAA model corresponding to the stochastic MIP introduced in Section 5.2 is as follows:

$$\max \ \frac{1}{q}\sum_{r\in R} y(\omega_r) \tag{5.7}$$

$$\text{s.t.} \sum_{i\in I}\sum_{l\in L_b} x_j^{il} = 1, \qquad \forall j \in J \tag{5.8}$$

$$\sum_{j\in J} s_j^{\mu} x_j^{il} - d_{il} \le c_{il}, \qquad \forall i \in I, \forall l \in L_b \tag{5.9}$$

$$\sum_{l\in L_b} h_{jl}(\omega_r) \sum_{i\in I} x_j^{il} \ge y(\omega_r), \qquad \forall j \in J, \forall r \in R \tag{5.10}$$

$$x_j^{il} \in \{0,1\}, \qquad \forall i \in I, \forall j \in J, \forall l \in L_b \tag{5.11}$$

$$y(\omega_r) \ \text{ free} \qquad \forall r \in R \tag{5.12}$$

Let u be the number of SAA replication and ω_r^{ρ} be the scenario set used for the ρth SAA replications, $\rho = \{1, \dots, u\}$.

For a feasible schedule $X = \{x_j^{il} | i \in I, j \in J, l \in L_b\}$, the following definitions are used.

$$\phi(X) := E\left[Y(X, \vec{H})\right],$$
$$v_q^{\rho}(X, \omega_r^{\rho}) := \frac{1}{q}\sum_{r\in R} y(X, \omega_r^{\rho}).$$

$Y(X,\vec{H})$ represents Y in the stochastic MIP but indicates that Y is dependent on X and $\vec{H}$. in the same way, $y(x,\omega_r^\rho)$ is another representation of $y(\omega_r^\rho)$.

Let X^* be the optimal solution of the stochastic MIP. The optimality gap for a given solution $\hat{X} = \{\hat{x}_j^{il} | i \in I, j \in J, l \in L_b\}$ is defined as

$$\phi(X^*) - \phi(\hat{X}).$$

Let $X^{\rho *}$ be the optimal solution of the ρth SAA solution. $\phi(X^*)$ is estimated by

$$\tilde{v_q}^u := \frac{1}{u} \sum_{\rho=1}^{u} v_q^\rho \left(X^{\rho *}, \omega_r^\rho\right). \tag{5.13}$$

$\tilde{v_q}^u$ is a biased estimator of $\phi(X^*)$ because

$$E\left[\tilde{v_q}^u\right] = \frac{1}{u} \sum_{\rho=1}^{u} E\left[v_q^\rho\left(X^{\rho *}, \omega_r^\rho\right)\right] \geq \frac{1}{u} \sum_{\rho=1}^{u} E[v_q^\rho\left(X^*, \omega_r^\rho\right)] = \frac{1}{u} \sum_{\rho=1}^{u} \phi(X^*) = \phi(X^*).$$

$\phi(\hat{X})$ is estimated by

$$\bar{v_q}^u\left(\hat{X}\right) := \frac{1}{u} \sum_{\kappa=1}^{u} v_q^\kappa\left(\hat{X}, \omega_r^\kappa\right). \tag{5.14}$$

$\bar{v_q}^u(\hat{X})$ is an unbiased estimator of $\phi(\hat{X})$ because

$$\begin{aligned} E\left[\bar{v_q}^u\left(\hat{X}\right)\right] &= \frac{1}{u} \sum_{\kappa=1}^{u} E[v_q^\kappa(\hat{X}, \omega_r^\kappa)] = \frac{1}{u} \sum_{\kappa=1}^{u} E\left[\frac{1}{q} \sum_{r \in R} y(\hat{X}, \omega_r^\kappa)\right] \\ &= \frac{1}{u} \sum_{\kappa=1}^{u} E\left[E[Y(\hat{X},\vec{H})]\right] = \frac{1}{u} \sum_{\kappa=1}^{u} E[Y(\hat{X},\vec{H})] = \frac{1}{u} \sum_{\kappa=1}^{u} \phi(\hat{X}) = \phi(\hat{X}). \end{aligned}$$

The variance of $\tilde{v_q}^u - \bar{v_q}^u(\hat{X})$ is calculated as follows:

$$Var\left[\tilde{v_q}^u - \bar{v_q}^u(\hat{X})\right] = Var\left[\frac{1}{u}\sum_{\rho=1}^{u} v_q^\rho\left(X^{\rho *}, \omega_r^\rho\right) - \frac{1}{u}\sum_{\rho=1}^{u} v_q^\rho\left(\hat{X}, \omega_r^\rho\right)\right]$$

$$= \frac{1}{u^2}\left\{Var\left[\sum_{\rho=1}^{u} v_q^\rho\left(X^{\rho *}, \omega_r^\rho\right) - \sum_{\rho=1}^{u} v_q^\rho\left(\hat{X}, \omega_r^\rho\right)\right]\right\} = \frac{1}{u^2}\sum_{\rho=1}^{u} Var\left[v_q^\rho\left(X^{\rho *}, \omega_r^\rho\right) - v_q^\rho\left(\hat{X}, \omega_r^\rho\right)\right]$$

$$= \frac{1}{u^2}\sum_{\rho=1}^{u}\frac{1}{u-1}\sum_{\rho=1}^{u}\left[\left\{v_q^\rho\left(x^{\rho *}, \omega_r^\rho\right) - v_q^\rho\left(\hat{X}, \omega_r^\rho\right)\right\} - \left\{\frac{1}{u}\sum_{\rho=1}^{u} v_q^\rho\left(X^{\rho *}, \omega_r^\rho\right) - \frac{1}{u}\sum_{\rho=1}^{u} v_q^\rho\left(\hat{X}, \omega_r^\rho\right)\right\}\right]^2$$

$$= \frac{1}{u(u-1)}\sum_{\rho=1}^{u}\left[\left\{v_q^\rho\left(X^{\rho *}, \omega_r^\rho\right) - v_q^\rho\left(\hat{X}, \omega_r^\rho\right)\right\} - \left\{\tilde{v_q}^u - \bar{v_q}^u\left(\hat{X}\right)\right\}\right]^2.$$

Note that $\tilde{v_q}^u - \bar{v_q}^u(\hat{X})$ is a sample mean of independant and identically distributed random variables and, thus, approximately normally distributed for sufficiently large u by the central limit theorem. Therefore, $100(1-\alpha)\%$ confidence interval on the optimality gap (CIOOG) for a given solution $\hat{X}$ is

$$\tilde{v_q}^u - \bar{v_q}^u\left(\hat{X}\right) + z_\alpha\sqrt{\frac{1}{u(u-1)}\sum_{\rho=1}^{u}\left[\left\{v_q^\rho\left(X^{\rho *}, \omega_r^\rho\right) - v_q^\rho\left(\hat{X}, \omega_r^\rho\right)\right\} - \left\{\tilde{v_q}^u - \bar{v_q}^u\left(\hat{X}\right)\right\}\right]^2} \tag{5.15}$$

where z_α denotes the value such that $P(Z > z_\alpha) = \alpha$; Z is a standard normal random variable.

An SAA algorithm proposed in this study is as follows:

Step 1. Choose u and q.

Step 2. For each $\rho = 1, 2, \ldots, u$, generate q scenarios and solve the ρth SAA replication. Obtain the optimal solution $X^{\rho *}$ and the corresponding objective value $v_q^\rho(X^{\rho *}, \omega_r^\rho)$ of the ρth SAA problem.

Step 3. Calculate $\tilde{v_q}^u$ by (5.13).

Step 4. For each ρ, calculate $\bar{v_q}^u(X^{\rho *})$ by (5.14).

Step 5. Select one solution, denoted by $\hat{X}_{saa}$, that provides the maximum value of $\bar{v_q}^u(X^{\rho *})$.

Step 6. Construct the $100(1-\alpha)\%$ CIOOG for $\hat{X}_{saa}$ by (5.15).

5.4 Numerical Results

This section presents the performance of the SAA algorithm described in Section 5.3 and the analyses on the solutions. The SAA algorithm is implemented in the General Algebraic Modeling System (GAMS) 24.1.3 on a PC with a 2.4 GHz Core i7 processor and 8GB RAM. The GAMS uses CPLEX 12.5.1 for solving MIPs.

5.4.1 Performance of the SAA Algorithm

To generate the distribution of mean surgery duration s_j^{μ}, several statistics of abdominal surgery durations are taken from [82]. They are abdomen exploration (class 1), inguinal hernia repair (class 2), laparoscopy and tubal cautery (class 3), and laparoscopic cholecystectomy (class 4). Each class's percentage of occurrence is calculated by the number of each class's surgeries divided by the total number of the four classes' surgeries. Based on the percentages, surgeries are generating. The study mean ranges are shown in Table 5.2. These values are $\pm 10\%$ deviated from the mean of each class's surgery duration to incorporate more variations in mean surgery durations so that the SAA algoritm can be tested on a wide range of surgeries. s_j^{μ} is randomly selected within the study mean range.

Table 5.2
Generating mean surgery durations.

Surgery class	Percentage of occurrence	Mean (minutes)	Study mean (minutes)
class 1	40.71%	194	[174.6, 213.4]
class 2	35.76%	143	[128.7, 157.3]
class 3	12.69%	105	[94.5, 115.5]
class 4	10.84%	219	[197.1, 240.9]

The scale for Karnofsky grade is used to represent patient health condition. It ranges from 0 (death) to 100 (normal health condition). It is assumed that H_{j1} (initial health condition of patient j) follows the normal distribution with mean μ^{INIT} and standard deviation σ^{INIT}, and the deteriorating rate from day l to day $l+1$ follows the folded normal distribution [97] with mean $\Delta\mu_{jl}$ and standard deviation $\Delta\sigma_{jl}$. Figure 5.2 illustrates how patient health condition is generated. The parameters and the distributions are somewhat arbitrary due to the lack of real data. However, those are made to represent the deteriorating trend of patient health condition under the following suppositions. First, since this study focuses on elective surgery, the initial health condition of each patient is not urgent (not below 30 on average) and the possibility of patient health condition becoming 0 (death) within a month is very low. Second, each patient needs to have a surgery, his/her initial health condition is not in the normal range (not above 90 on average).

In this section, block day is defined as the day where the surgical block(s) of the surgeon or the surgical group exist(s). Other parameters used for the experiments are set to be as follows: L_b (set of block days) $= \{2,5,9,11,16,18,23,25\}$ (i.e., the surgeon or the surgical group has surgical blocks every Tuesday and Thursday for a month), d_{1l} (available time of the first block on day l) $= 480$ minutes $\forall l \in L_b$, d_{2l} (available time of the second block on day l) $= 0\ \forall l \in L_b$, $c_{il} = 0\ \forall i \in I, \forall l \in L_b$, and n (number of patients) $= 16$.

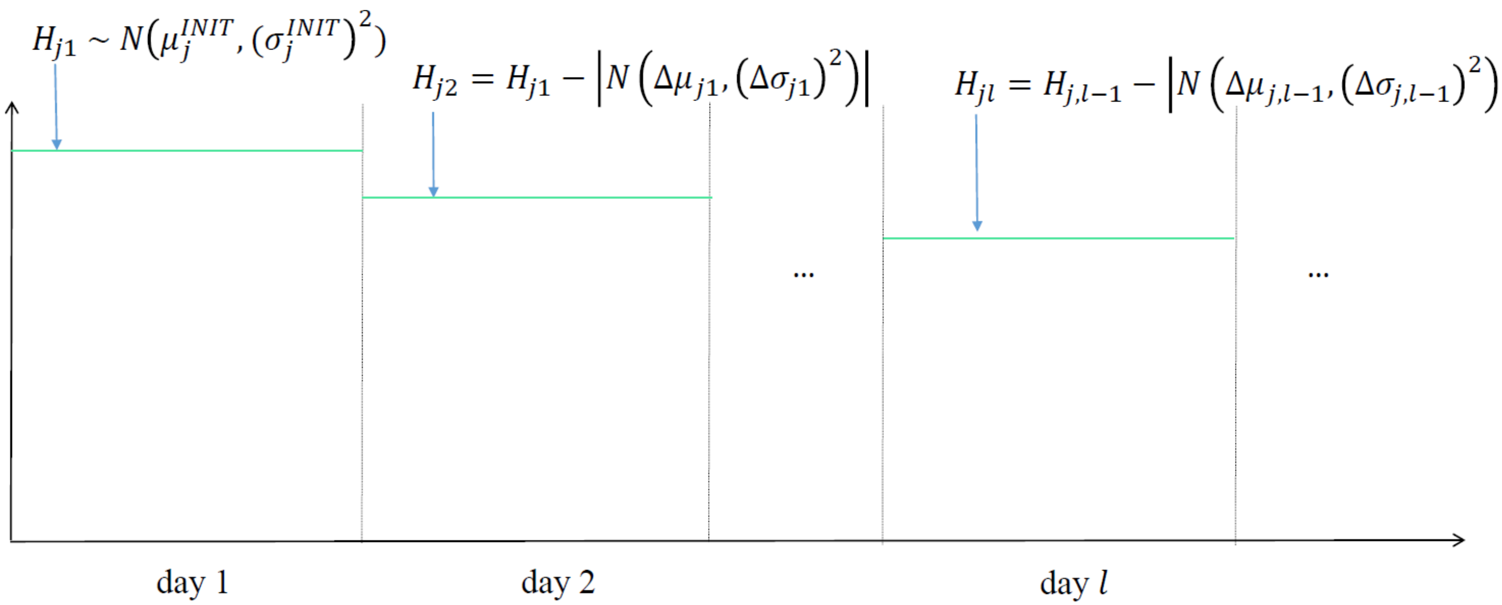
A set of scenarios is used to obtain an SAA solution, and a different set of 400 scenarios is used to evaluate the obtained solution $\hat{X} = \{\hat{x}_j^{il} | i \in I, j \in J, l \in L_b\}$. $\hat{X}$ is evaluated by

$$\frac{1}{400}\sum_{r=1}^{400}\min_j\left\{\sum_{l\in L_b} h_{jl}(\omega_r)\sum_{i\in I}\hat{x}_j^{il}\right\}. \tag{5.16}$$

When the objective values of solutions are compared in this section, it is supposed that the solutions are obtained for the same instance (i.e, the same distribution for each patient's mean surgery duration and the same distribution for each patient's health condition), and the objective values of solutions are evaluated using the same set of 400 scenarios.

The SAA algorithm is tested for 40 cases of 10 instances each.

Table 5.3 demonstrates the trade-off between the computation time of the SAA algorithm and the CIOOG. In general, as q (number of scenarios for an SAA replication) and u (number of SAA replications) increase, the computation time of the SAA algoritm in-



where $\mu_j^{INIT} \sim U(30,90)$, $\sigma_j^{INIT} \sim U(0,5)$, $\Delta\mu_{jl} \sim U(0,2)$, $\Delta\sigma_{jl} \sim U(0,2)$, $\forall j \in J, \forall l \in L \setminus \{m\}$

Figure 5.2. Generating patient health condition.

creases and the CIOOG of the solution decreases. In other words, to statistically prove that the solution is very close to the optimal solution (i.e., near-optimal solution), large q and u need to be used. However, it increases the computation time.

q and u can be selected based on the convergence of objective values (5.16). Figure 5.3 (a) shows the average objective value of the solutions with respect to q and u. The solution with $u = 10$ and that with $u = 20$ converge asymptotically to the same objective value (5.16) from $q = 12$. The algorithm may not need to increase u to find a better solution if $q \geq 12$. When $u = 10$, the convergence occurs at $q = 12$. When $u = 20$, the convergence occurs at $q = 10$. Even though, in both cases, the solutions converge asymptotically to the same objective value (5.16), the algorithm with $q = 12$ and $u = 10$ has less computation time that that with $q = 10$ and $u = 20$. This study suggests to use $q = 12$ and $u = 10$ in practice and the analyses presented in Section 5.4.2 use $q = 12$ and $u = 10$ to run the algorithm.

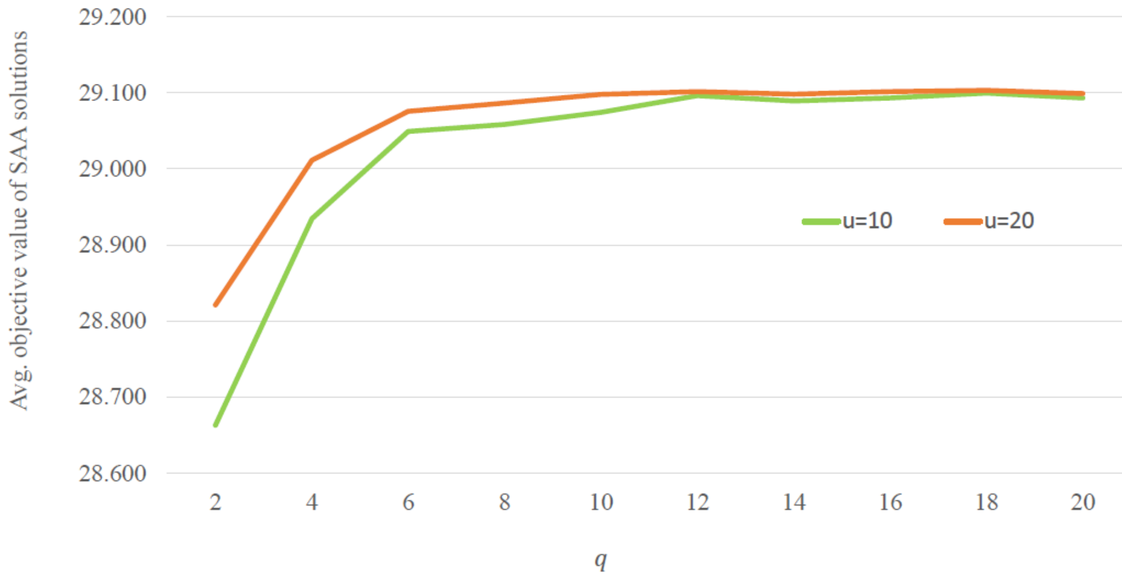
Figures 5.3 (a) and (b) also show that while the convergence of objective values (5.16) occurs at $q = 12$ and $u = 10$ (or $q = 10$ and $u = 20$), the CIOOG keeps decreasing as q and u increase. It tells that, even though the solutions with $q = 12$ and $u = 10$ is statistically worse than those with $q = 20$ and $u = 10$, the actual quality (i.e., objective values (5.16)) of the solutions with $q = 12$ and $u = 10$ is as good as that of the solutions with $q = 20$ and $u = 10$.

5.4.2 Analyses

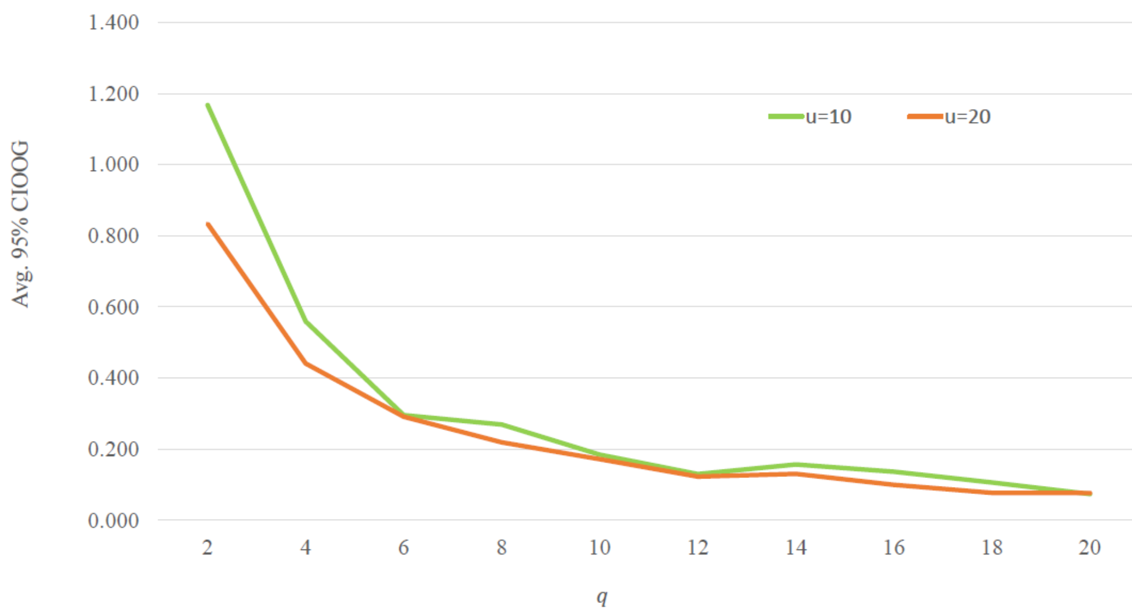
This subsection analyzes the SAA solutions based on *Ceteris Paribus* [98]. The technique allows to manipulate one or two parameter(s) of patient health condition with other parameters remaining the same and draw managerial insights from the solutions for elective surgery scheduling to maximize patient health condition. In the experiments for the full-day block system, d_{1l} is set to be 480 minutes for all $l \in L_b$ and d_{2l} is set to be 0 for all $l \in L_b$, while, in the experiments for the half-day block system, both d_{1l} and d_{2l} are set to be 240 minutes for all $l \in L_b$. It is supposed that L_b, c_{il}, n and s_j^{μ} are set to be the same or generated on the same way in Subsection 5.4.1 unless stated otherwise. Recall

Table 5.3
Performance of SAA algorithm.

q	u	Average computation time (seconds)	Average 95% CIOOG
2	10	2.496	1.167
4	10	2.741	0.559
6	10	2.862	0.296
8	10	3.078	0.269
10	10	3.067	0.184
12	10	3.308	0.130
14	10	3.465	0.157
16	10	3.641	0.136
18	10	3.831	0.106
20	10	4.124	0.073
2	20	5.220	0.834
4	20	5.861	0.441
6	20	6.225	0.291
8	20	6.916	0.219
10	20	7.564	0.172
12	20	7.902	0.123
14	20	8.336	0.130
16	20	8.351	0.099
18	20	8.748	0.077
20	20	9.289	0.076



(a) Average objective value (5.16) of SAA solutions



(b) Average 95% CIOOG

Figure 5.3. Selecting q and u, (a) average objective value (5.16) of SAA solutions, (b) average 95% CIOOG.

that L_b (set of block days)= $\{2,5,9,11,16,18,23,25\}$. For convenience, let "block day" $1,2,3,4,5,6,7$, and 8 denote day $2,5,9,11,16,18,23$ and 25, respectively.

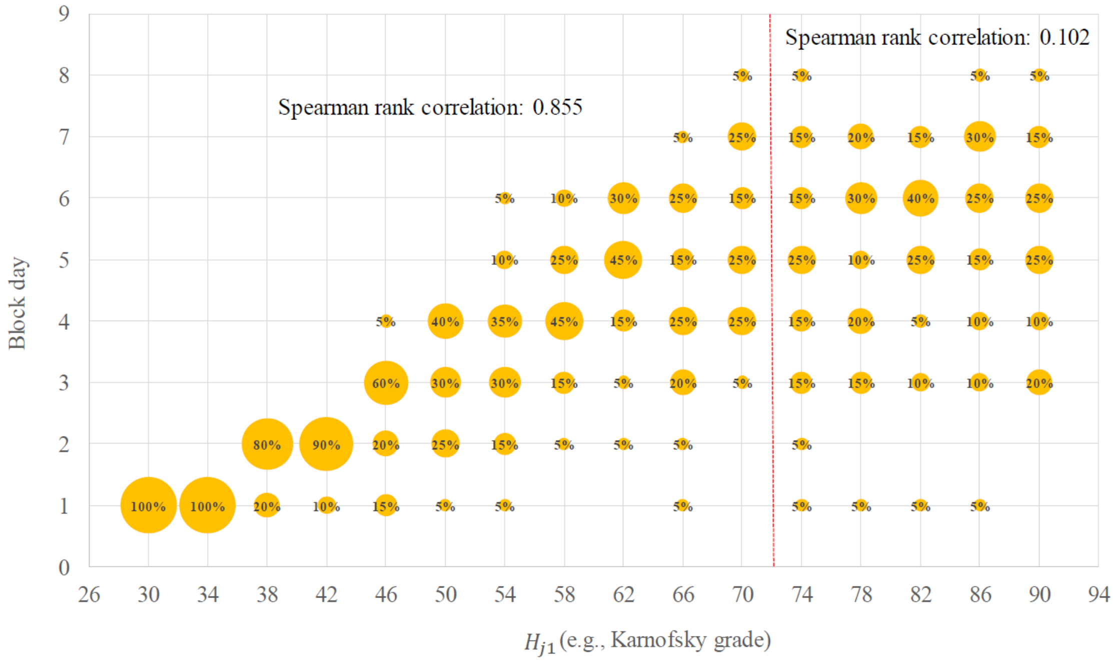
Should sicker patients be scheduled earlier than others?

This analysis deals with the case that patients waiting for surgeries have different severity levels but patients' disease progressions are assumed to be the same due to the lack of data, based on a physician's opinion, or because the historical data say the progressions of such diseases are the same. The increment of H_{j1} is set to be 4: $H_{1,1} = 30, H_{2,1} = 34, \ldots, H_{16,1} = 90$. Disease progression parameters (i.e., $\Delta\mu_{jl}$ and $\Delta\sigma_{jl}$) are set to be the same: $\Delta\mu_{jl} = 1, \Delta\sigma_{jl} = 1 \;\; \forall j \in J, \forall l \in L \backslash \{m\}$.

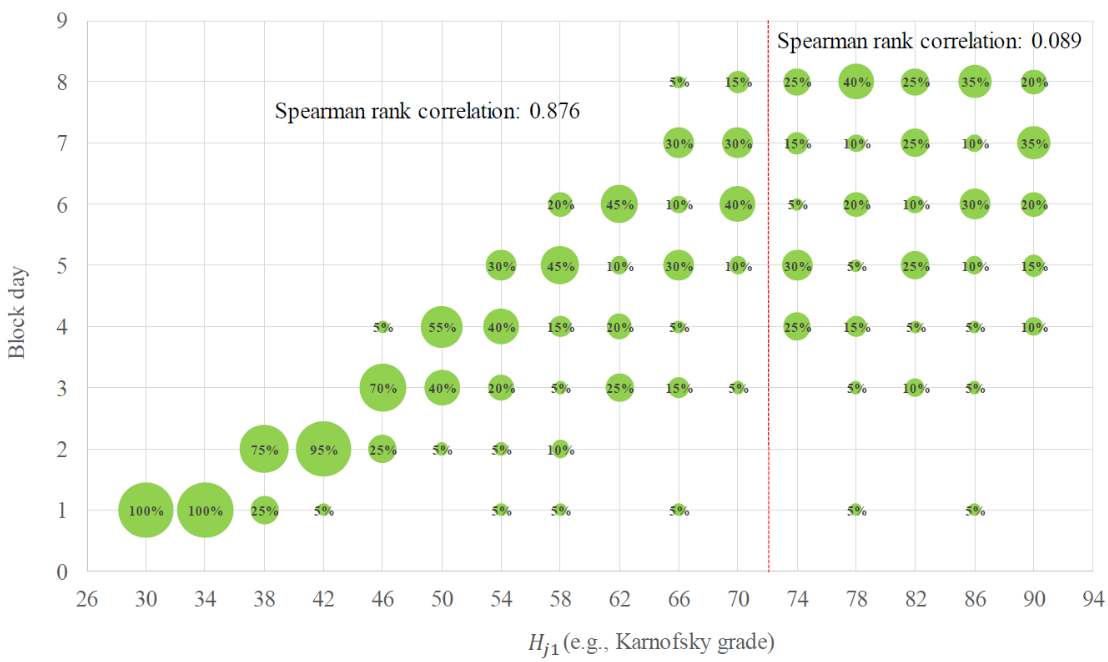
Figure 5.4 shows the block days on which each patient is scheduled for surgery based on his/her H_{j1}. A bubble represents the percentage of instances each patient is assigned to each block day out of 20 instances. Sicker patients tend to be scheduled earlier than others until

Table 5.4
Spearman rank correlation when the increment of H_{j1} is 4.

		Spearman rank correlation	
From	To	Full-day block system	Half-day block system
30	38	0.739	0.707
30	42	0.792	0.814
30	46	0.789	0.886
30	50	0.811	0.919
30	54	0.815	0.899
30	58	0.850	0.878
30	62	0.873	0.879
30	66	0.841	0.863
30	70	0.855	0.884
30	74	0.821	0.876
30	78	0.800	0.856
30	82	0.789	0.844
30	86	0.778	0.826
30	90	0.762	0.821



(a) Full-day block system



(b) Half-day block system

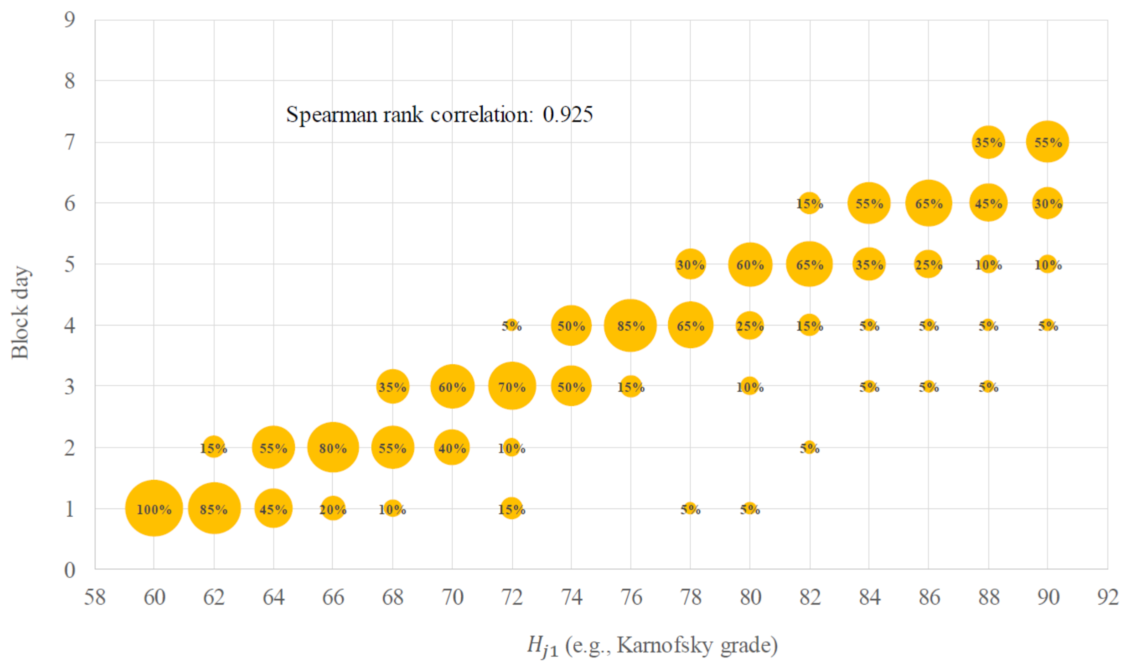
Figure 5.4. Block days on which each patient is scheduled for surgery based on his/her H_{j1} when the increment of H_{j1} is 4, (a) full-day block system, (b) half-day block system.

the dotted lines. The dotted lines are decided observing Spearman rank correlation [94] between H_{j1} and block day. Spearman rank correlation is calculated increasing H_{j1} (see Table 5.4). The dotted lines are drawn when Spearman rank correlation starts to drop and never goes up. According to Mukaka [99], correlation 0.7 (-0.7) to 1 (-1) is interpreted as high correlation while correlation 0 to 0.3 (-0.3) is interpreted as negligible correlation. The patients after the dotted lines tend to be assigned to the remaining slots arbitrarily rather than being kept in the sequence. It is because the patient after the dotted line have little likelihood to be the most critical patient even if they are assigned to the last block day.

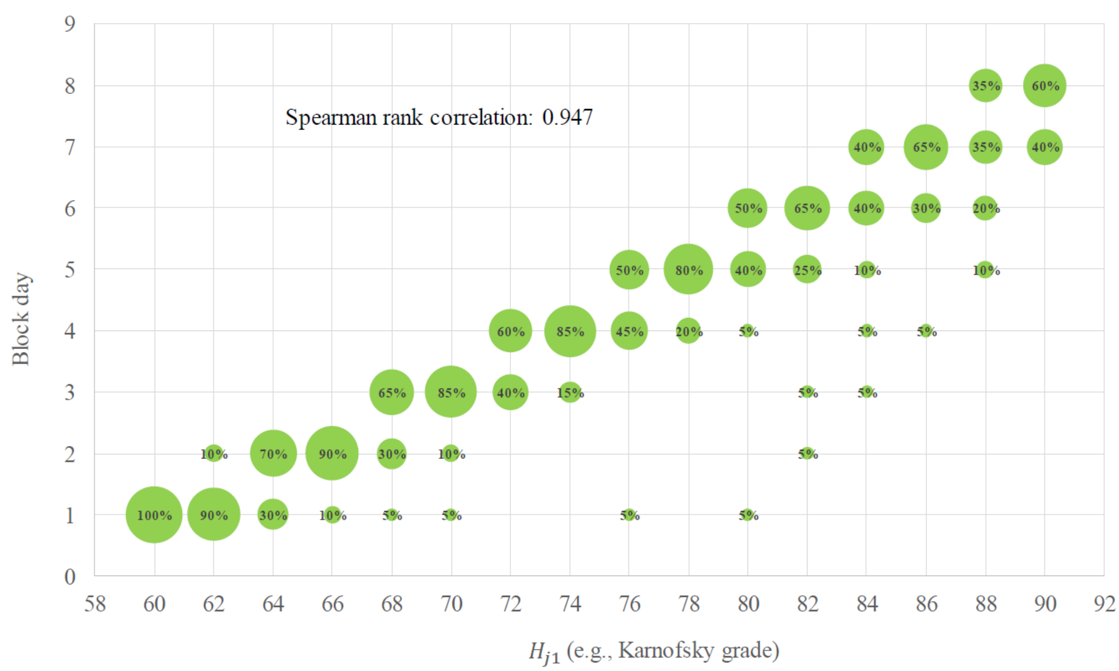
Figure 5.5 shows that the surgery assignment when the increment of H_{j1} is 2. In this case, the statement that sicker patient needs to be scheduled earlier applies to the entire group of patients. The difference of the likelihood to be the most critical patient when assigned to a block day is relatively small between patients. If patient A, who is sicker than patient B, is schedule after patient B, the minimum patient health condition (i.e., objective value) may become worse. Therefore, the surgery sequence tends to be kept in the increasing order of H_{j1}.

The main findings in this analysis are summarized. To schedule a similar group of patients in terms of initial patient health condition, the surgery sequence needs to be kept in the increasing order of initial patient health condition. If the difference of initial patient health condition between patients is relatively large, patients can be divided into two groups depending on whether the patient has the likelihood to be the most critical patient or not when he/she is assigned to the last block day. The group having the likelihood to be the most critical patient needs to be scheduled in the increasing order of initial patient health condition. Another group can be assigned to the remaining slots in order to pursue a secondary performance measure like reducing the number of block days used for surgeries.

Note that the number of blocks used for surgeries in the full-day block system is smaller than that in the half-day block system since one large time duration can accommodate more surgeries than two small time durations. It is found in all analyses of this subsection.



(a) Full-day block system



(b) Half-day block system

Figure 5.5. Block days on which each patient is scheduled for surgery based on his/her H_{j1} when the increment of H_{j1} is 2, (a) full-day block system, (b) half-day block system.

How to schedule if the variations of patients' disease progressions are different

This analysis deals with the case that patients waiting for surgeries have the same initial health condition and are predicted to deteriorate at the same rate on average but the variations of patients' disease progressions are different. The increment of $\Delta\sigma_{ij}$ is set to be 0.2: $\Delta\sigma_{1l} = 0.2, \Delta\sigma_{2l} = 0.4, \ldots, \Delta\sigma_{16l} = 3.4 \ \ \forall l \in L \backslash \{m\}$. H_{j1} is set to be 70 $\forall j \in J$ and $\Delta\mu_{jl}$ is set to be 1 $\forall j \in J, \forall l \in L \backslash \{m\}$.

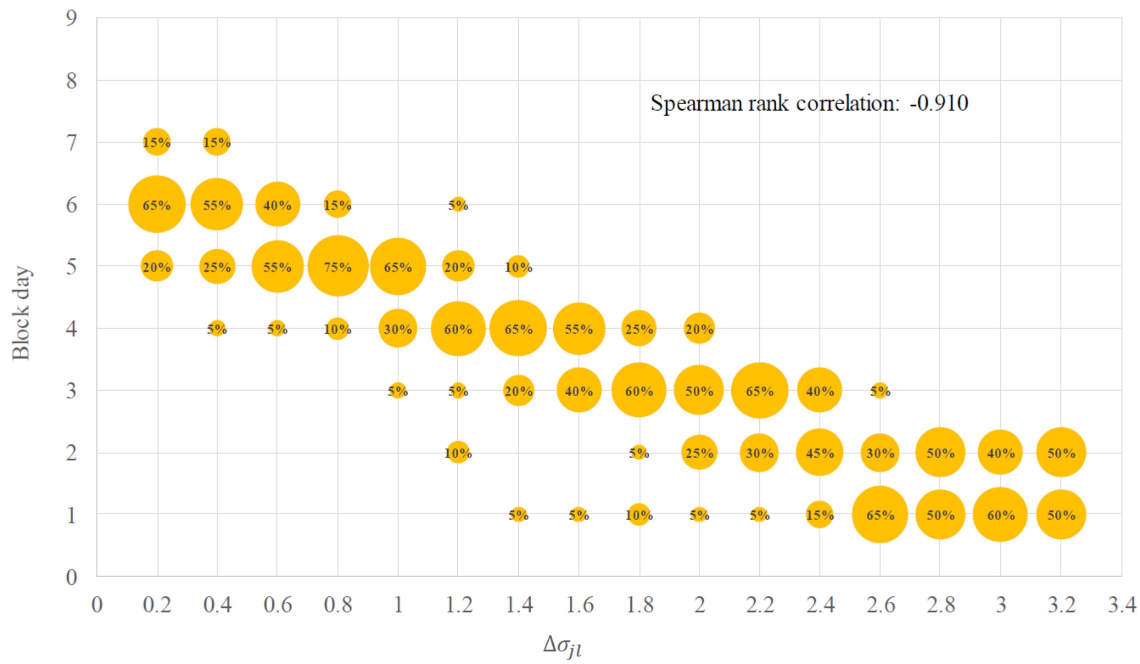
Figure 5.6 shows, obtaining the solutions for 20 instances, the block days on which each patient is scheduled for surgery based on his/her $\Delta\sigma_{jl}$. Patients tend to be scheduled in the decreasing order of $\Delta\sigma_{jl}$. A patient having higher $\Delta\sigma_{jl}$ has a higher likelihood to be the most critical patient than others in any block days. Therefore, the patient is scheduled first before the patient gets worse not to decrease the minimum patient health condition. Even though this study presents a case that the increment of $\Delta\sigma_{jl}$ is 0.2, this trend is kept regardless of how large the increment is.

How to schedule if a patient is healthier (or sicker) than others but his/her health condition deteriorates rapidly (or slowly)

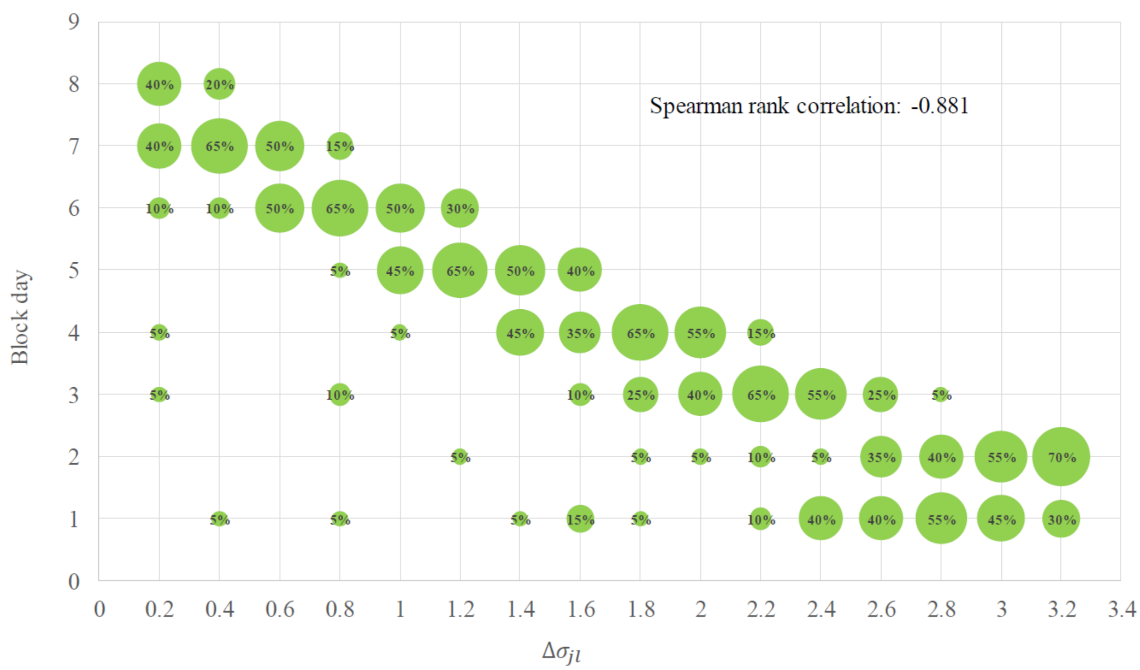
This analysis deals with the case that a patient's initial health condition is better (or worse) than others but deteriorates rapidly (or slowly) due to the innate characteristic of the disease.

First, Patient 1 is characterized by the following parameters: $H_{11} = 80, \Delta\mu_{1l} = 2, \Delta\sigma_{1l} = 1 \ \ \forall l \in L \backslash \{m\}$. Other patients have the following parameters: $H_{j1} = 70, \Delta\mu_{jl} = 1, \Delta\sigma_{jl} = 1 \ \ \forall j \in J \backslash \{1\}, \forall l \in L \backslash \{m\}$.

Figure 5.7 shows that the number of instances (out of 20 instances) Patient 1 is assigned to each block. Patient 1 is scheduled before or on block day 4. Figure 5.8 explains the reason why Patient 1 is scheduled before or on block day 4. Figure 5.8 is obtained using 400 scenarios for two different sets of patient health condition parameters described above. Mean patient health condition of Patient 1 is higher than those of others before and on

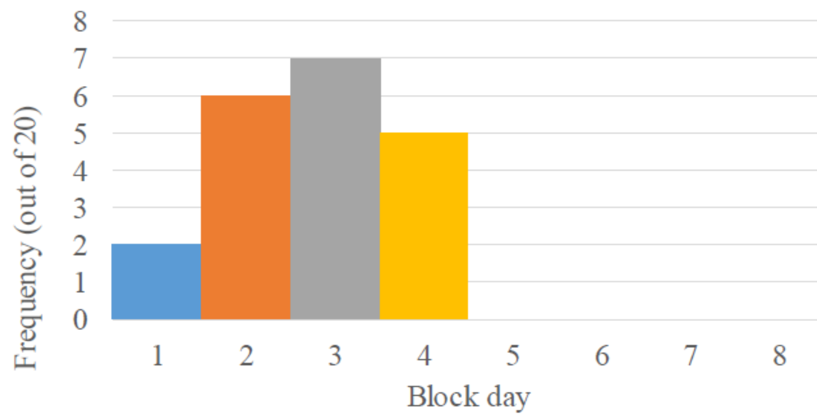


(a) Full-day block system

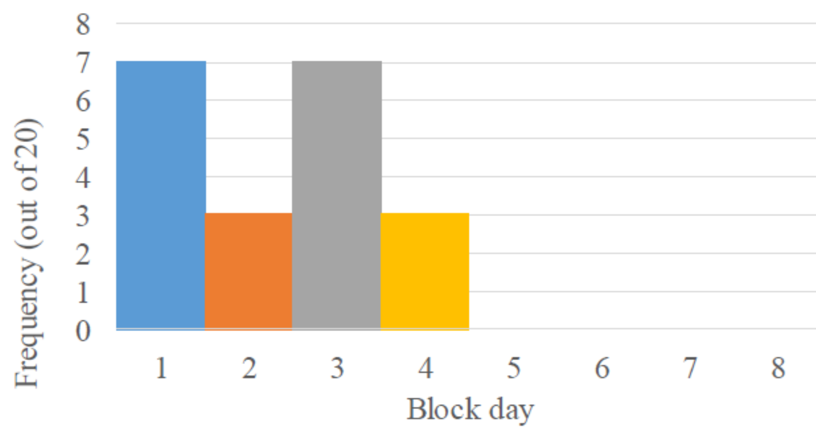


(b) Half-day block system

Figure 5.6. Block days on which each patient is scheduled for surgery based on his/her $\Delta\sigma_{jl}$ when the increment of $\Delta\sigma_{jl}$ is 0.2, (a) full-day block system, (b) half-day block system.



(a) Full-day block system



(b) Half-day block system

Figure 5.7. Block days on which Patient 1 is scheduled for surgery when Patient 1's initial health condition is better than others but deteriorates rapidly, (a) full-day block system, (b) half-day block system.

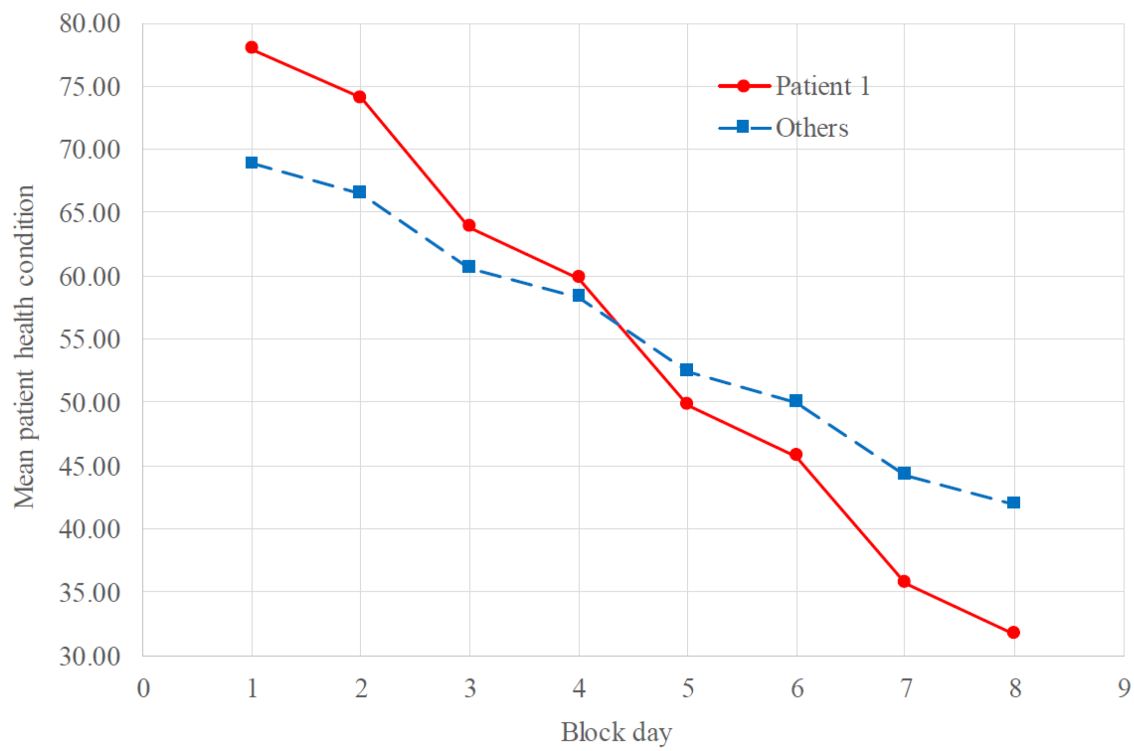
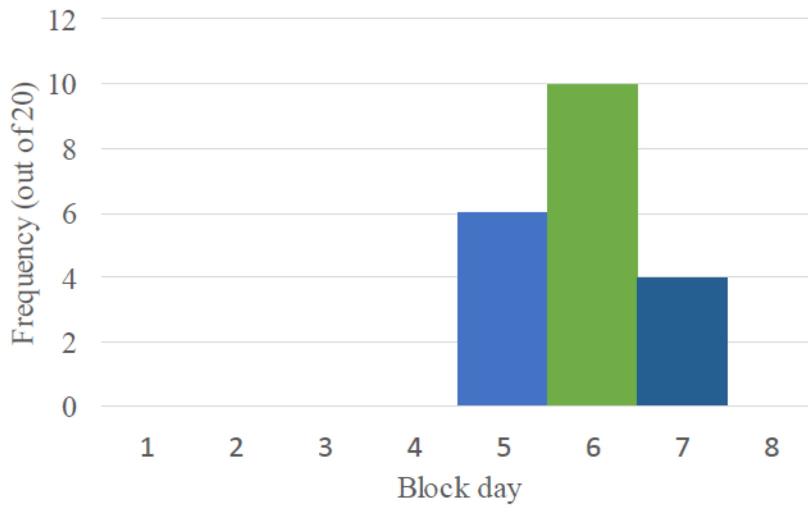


Figure 5.8. Mean patient health condition depending on block days.

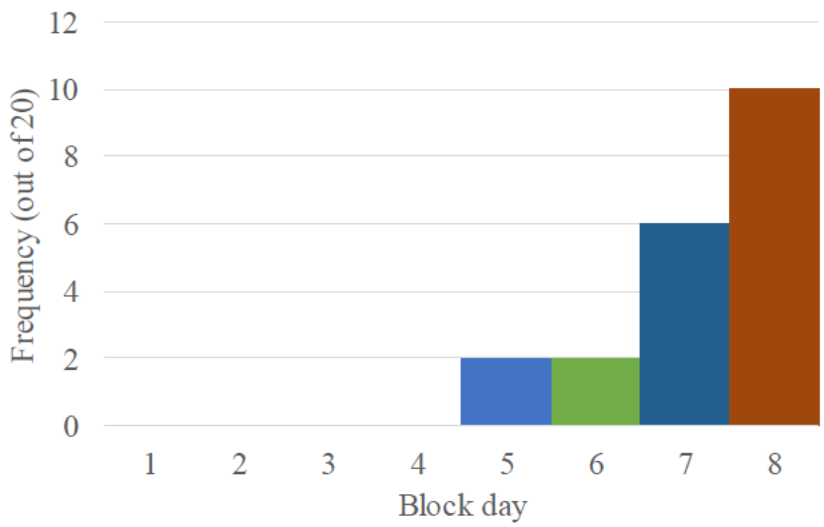
block day 4. Therefore, to escape from the risk to be the most critical patient, Patient 1 is scheduled before his/her mean health condition is lower than others' health conditions.

Patient condition parameter are manipulated as follows to implement the opposite case (i.e., a patient's initial health condition is worse than others but deteriorates slowly): $H_{11} = 70, \Delta\mu_{1l} = 1, \Delta\sigma_{1l} = 1 \ \ \forall l \in L \backslash \{m\}, H_{j1} = 80, \Delta\mu_{jl} = 2, \Delta\sigma_{jl} = 1 \ \ \forall j \in J \backslash \{1\}, \forall l \in L \backslash \{m\}$. In this case, Patient 1 is scheduled after or on block day 5 (see Figure 5.9) not to delay the surgeries for other patients who are highly likely to be the most critical patient. However, it does not mean that Patient 1 should be scheduled late.

If n (the number of patients) is 8, the number of block days used for surgeries is 3 or 4. Then, during the block days used for surgeries, Patient 1's mean health condition is always lower than others' health condition. Therefore, not to decrease the minimum patient health condition, Patient 1 needs to be scheduled early (see Figure 5.10).

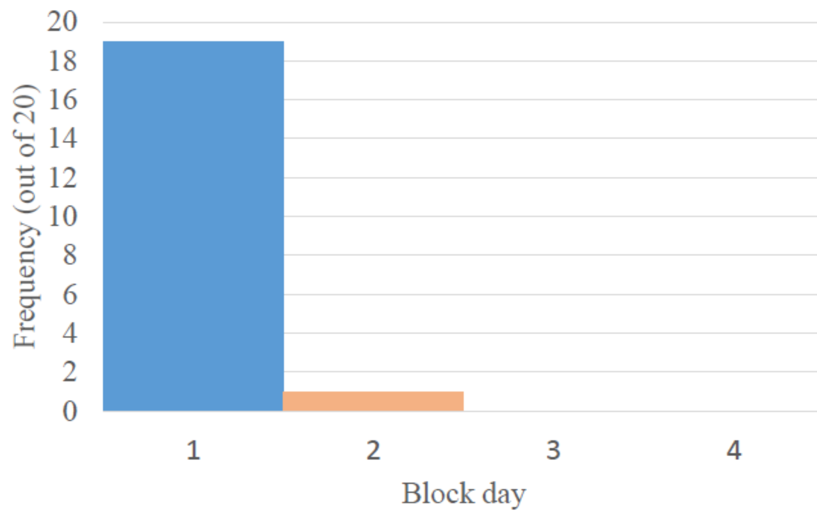


(a) Full-day block system

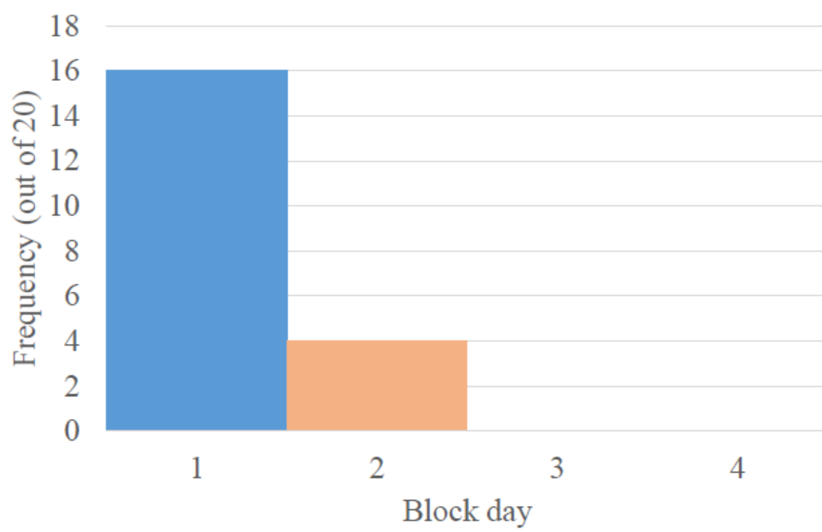


(b) Half-day block system

Figure 5.9. Block days on which Patient 1 is scheduled for surgery when Patient 1's initial health condition is worse than others but deteriorates slowly ($n = 16$), (a) full-day block system, (b) half-day block system.



(a) Full-day block system



(b) Half-day block system

Figure 5.10. Block days on which Patient 1 is scheduled for surgery when Patient 1's initial health condition is worse than others but deteriorates slowly ($n = 8$), (a) full-day block system, (b) half-day block system.

5.5 Conclusions

In this chapter, a elective surgery scheduling problem with step-deteriorating patient health condition is described and formulated as a stochastic MIP. The problem incorporates the concept of maximizing the minimum patient health condition, which improves critical-patient safety.

An SAA algorithm is presented to solve the stochastic MIP in this chapter. The computational study shows that the SAA algorithm applied to the elective surgery scheduling problem with step-deteriorating patient health condition provides near-optimal solutions within short computation times.

In addition, the SAA solutions are analyzed based on *Ceteris Paribus* [98]. It allows to manipulate one or two parameter(s) of patient health condition with other parameters remaining the same and draw managerial insights from the solutions. The analyses discuss how to schedule (1) if the initial conditions of patients are different, (2) if the variations of patients' disease progressions are different, and (3) if a patient is healthier (or sicker) than others but his/her health condition deteriorates rapidly (or slowly).

6. CONCLUSIONS AND FUTURE RESEARCH

In this dissertation, elective surgery scheduling problems considering patient health condition are discussed. Since the shape of a function to represent patient health condition has rarely been studied in the field of medical science, a couple of shapes are suggested in this dissertation. This dissertation also provides a taxonomy of the literature that considers overtime according to performance measures, solution approaches, and whether to incorporate the uncertainty in surgery durations. Based on the taxonomy, it is easy to know that the idea of maximizing patient health condition, which improves the most critical patient safety, has not been discussed in the literature.

6.1 Sample Average Approximation Approach to Elective Surgery Scheduling with Linearly Deteriorating Patient Health Condition

To schedule a surgery, surgical schedulers must know the time frame within which the surgery should be performed [73]. The time frame is decided appropriately by clinicians considering patient health condition [74, 75]. In other words, clinicians assess the health condition of a patient when he/she is diagnosed and, according to his/her health condition, set the critical time point by which the surgery should be performed. This study uses the information about the current patient health condition and the critical time point. The line connecting the two points (i.e., current patient health condition and critical time point) is employed as an approximation for deteriorating patient health condition.

The objective of the problem is to find a surgery scheduling that provides the optimal trade-off between maximizing minimum patient health condition and minimizing total overtime of an OR. Since, in the problem, the uncertainty in surgery durations is incorporated, The sample average approximation method is employed to manage the uncertainty. The advantage for using the sample average approximation method is that the quality of the obtained solution can be statistically obtained. However, the sample average approximation used for this problem is very time consuming and another solution approaches are needed to solve large-size problems.

6.2 Heuristic Approach to Elective Surgery Scheduling with Linearly Deteriorating Patient Health Condition

Heuristics are developed for the problem described in the previous chapter. Key structural elements to implement the heuristics are summarized as follows: how to obtain an initial feasible solution, how to evaluate the current solution, and how to move from a current solution to the next solution. Those are designed considering solution effectiveness and algorithm efficiency.

Based on the key structural elements, a fastest ascent local search and a tabu search are developed. The concept of search intensity to implement the tabu search is introduced. The idea behind the concept of the search intensity is that the algorithm examine promising areas thoroughly not to miss the best solutions in those areas while the algorithm skims non-promising areas to reduce the computational burden. The fastest ascent local search and the tabu search are compared with the sample average approximation algorithm in the computational study. The computational study show that the tabu search provides near-optimal solutions within reasonable computation times.

6.3 Elective Surgery Scheduling with Step-Deteriorating Patient Health Condition

In this study, it is assumed that patient severity levels are recorded whenever patients are diagnosed by practitioners using severity level measures like Kanorfsky grade [77]. Those data are used to represent time-dependent patient health condition. The uncertainty in patient health condition is considered in this study. The sample average approximation method works well for this problem in terms of solution effectiveness and computation time efficiency.

The sample average approximation solutions are analyzed to discuss how to schedule (1) if the initial severity levels of patients are different, (2) if the variations of patients' disease progressions are different, and (3) if a patient is healthier (or sicker) than others but his/her health condition deteriorates rapidly (or slowly).

6.4 Future Research

The following two broad topics will be pursued: 1) understanding dynamic patient health condition and 2) its applications in healthcare delivery systems.

6.4.1 Understanding Dynamic Patient Health Condition

There are many disease-specific measures that indicate patient health condition. Examples of the disease-specific measures are dyspnea index (DI) for lung disease, Karnofsky grade for cancer, and model for end-stage liver disease (MELD) score. Using the trajectories of those measures and patient information (e.g., age, gender, smoking, and disease history), The health condition patterns of individual patients will be characterized.

6.4.2 Applications of Dynamic Patient Health Condition in Healthcare Delivery Systems

First, the health condition patterns of individual patients will be used to reduce medical costs in the diagnostic process. Making a diagnosis is a complex and difficult task that requires sequential diagnostic tests. Patients sometimes undergo unnecessary tests and, thereby, their medical costs increase. By incorporating the health condition patterns into the diagnostic process, more cost-effective and time-efficient transition between diagnostic tests will be implemented.

Second, the health condition patterns of individual patients will be used to extend my doctoral research for surgery scheduling. The research focuses on incorporating patient health condition to improve patient safety. However, due to the lack of relevant research and real-world data, the research uses a couple of simple but plausible functions to represent patient health condition. The research will be extended to the one that incorporates the health condition patterns supported by real-world data.

LIST OF REFERENCES

LIST OF REFERENCES

[1] J. R. Clarke, J. C. Lerner, and W. Marella. The role for leaders of health care organizations in patient safety. *American Journal of Medical Quality*, 22(5):311–318, 2007.

[2] L. C. Kaldjian, E. W. Jones, B. J. Wu, V. L. Forman-Hoffman, B. H. Levi, and G. E. Rosenthal. Reporting medical errors to improve patient safety: a survey of physicians in teaching hospitals. *Archives of internal medicine*, 168(1):40–46, 2008.

[3] T. Grundgeiger and P. Sanderson. Interruptions in healthcare: theoretical views. *International journal of medical informatics*, 78(5):293–307, 2009.

[4] S. Garfield, N. Barber, P. Walley, A. Willson, and L. Eliasson. Quality of medication use in primary care-mapping the problem, working to a solution: a systematic review of the literature. *BMC medicine*, 7(1):50, 2009.

[5] D. Brock, E. Abu-Rish, C.-R. Chiu, D. Hammer, S. Wilson, L. Vorvick, K. Blondon, D. Schaad, D. Liner, and B. Zierler. Interprofessional education in team communication: working together to improve patient safety. *BMJ Quality & Safety*, 22(5):414–423, 2013.

[6] World Health Organization. World alliance for patient safety progress report 2006-2007. 2008.

[7] B. Cardoen, E. Demeulemeester, and J. Beliën. Operating room planning and scheduling: A literature review. *European Journal of Operational Research*, 201(3):921–932, 2010.

[8] M. Milinski and G. A. Parker. Competition for resources. *Behavioural ecology: an evolutionary approach*, 3:137–68, 1991.

[9] The Office of the Actuary. *National Health Expenditure Projections 2014-2024*. Centers for Medicare and Medicaid Services, 2015.

[10] The Office of the Actuary. *National Health Expenditure Historical and Projections 1960-2024*. Centers for Medicare and Medicaid Services, 2015.

[11] B. Denton, J. Viapiano, and A. Vogl. Optimization of surgery sequencing and scheduling decisions under uncertainty. *Health Care Management Science*, 10(1):13–24, 2007.

[12] D.-N. Pham and A. Klinkert. Surgical case scheduling as a generalized job shop scheduling problem. *European Journal of Operational Research*, 185(3):1011–1025, 2008.

[13] D. Min and Y. Yih. Scheduling elective surgery under uncertainty and downstream capacity constraints. *European Journal of Operational Research*, 206(3):642–652, 2010.

[14] S. Lee and Y. Yih. Reducing patient-flow delays in surgical suites through determining start-times of surgical cases. *European Journal of Operational Research*, 238(2):620–629, 2014.

[15] A. Macario. What does one minute of operating room time cost? *Journal of clinical anesthesia*, 22(4):233–236, 2010.

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

[17] J. M. Magerlein and J. B. Martin. Surgical demand scheduling: a review. *Health Services Research*, 13(4):418, 1978.

[18] Z. H. Przasnyski. Operating room scheduling: A literature review. *AORN Journal*, 44(1):67–82, 1986.

[19] J. T. Blake and M. W. Carter. Surgical process scheduling: a structured review. *Journal of the Society for Health Systems*, 5(3):17–30, 1996.

[20] J. H. May, W. E. Spangler, D. P. Strum, and L. G. Vargas. The surgical scheduling problem: Current research and future opportunities. *Production and Operations Management*, 20(3):392–405, 2011.

[21] P. J. H. Hulshof, N. Kortbeek, R. J. Boucherie, E. W. Hans, and P. J. M. Bakker. Taxonomic classification of planning decisions in health care: a structured review of the state of the art in OR/MS. *Health Systems*, 1(2):129–175, 2012.

[22] I. J. B. F. Adan and J. M. H. Vissers. Patient mix optimisation in hospital admission planning: a case study. *International journal of operations & production management*, 22(4):445–461, 2002.

[23] J. Bowers and G. Mould. Managing uncertainty in orthopaedic trauma theatres. *European Journal of Operational Research*, 154(3):599–608, 2004.

[24] M. E. Bruni, P. Beraldi, and D. Conforti. A stochastic programming approach for operating theatre scheduling under uncertainty. *IMA Journal of Management Mathematics*, 26(1):99–119, 2015.

[25] B. Cardoen and E. Demeulemeester. Capacity of clinical pathwaysâĂŤa strategic multi-level evaluation tool. *Journal of medical systems*, 32(6):443–452, 2008.

[26] S. Chaabane, N. Meskens, A. Guinet, and M. Laurent. Comparison of two methods of operating theatre planning: application in belgian hospital. *Journal of Systems Science and Systems Engineering*, 17(2):171–186, 2008.

[27] B. Denton and D. Gupta. A sequential bounding approach for optimal appointment scheduling. *IIE Transactions*, 35(11):1003–1016, 2003.

[28] R. M. Van Slyke and R. Wets. L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics*, 17(4):638–663, 1969.

[29] B. Denton, A. J. Miller, H. J. Balasubramanian, and T. R. Huschka. Optimal allocation of surgery blocks to operating rooms under uncertainty. *Operations research*, 58(4-part-1):802–816, 2010.

[30] F. Dexter. A strategy to decide whether to move the last case of the day in an operating room to another empty operating room to decrease overtime labor costs. *Anesthesia & Analgesia*, 91(4):925–928, 2000.

[31] F. Dexter and A. Macario. When to release allocated operating room time to increase operating room efficiency. *Anesthesia & Analgesia*, 98(3):758–762, 2004.

[32] F. Dexter and R. D. Traub. How to schedule elective surgical cases into specific operating rooms to maximize the efficiency of use of operating room time. *Anesthesia & Analgesia*, 94(4):933–942, 2002.

[33] F. Dexter, R. D. Traub, and A. Macario. How to release allocated operating room time to increase efficiency: predicting which surgical service will have the most underutilized operating room time. *Anesthesia & Analgesia*, 96(2):507–512, 2003.

[34] F. Dexter, A. Willemsen-Dunlap, and J. D. Lee. Operating room managerial decision-making on the day of surgery with and without computer recommendations and status displays. *Anesthesia & Analgesia*, 105(2):419–429, 2007.

[35] R. H. Epstein and F. Dexter. Uncertainty in knowing the operating rooms in which cases were performed has little effect on operating room allocations or efficiency. *Anesthesia & Analgesia*, 95(6):1726–1730, 2002.

[36] H. Fei, C. Chu, N. Meskens, and A. Artiba. Solving surgical cases assignment problem by a branch-and-price approach. *International Journal of Production Economics*, 112(1):96–108, 2008.

[37] G. B. Dantzig and P. Wolfe. The decomposition algorithm for linear programs. *Econometrica: Journal of the Econometric Society*, pages 767–778, 1961.

[38] A. Fügener, E. W. Hans, R. Kolisch, N. Kortbeek, and P. T. Vanberkel. Master surgery scheduling with consideration of multiple downstream units. *European Journal of Operational Research*, 2014.

[39] B. Gladish, M. A. Parra, A. B. Terol, and M. V. R. UrıÌĄa. Management of surgical waiting lists through a possibilistic linear multiobjective programming problem. *Applied Mathematics and Computation*, 167(1):477–495, 2005.

[40] Didier Dubois and Henri Prade. *Possibility theory*. Springer, 1988.

[41] A. Guinet and S. Chaabane. Operating theatre planning. *International Journal of Production Economics*, 85(1):69–81, 2003.

[42] D. Gupta. Surgical suites' operations management. *Production and Operations Management*, 16(6):689–700, 2007.

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

[44] W. L. Herring and J. W. Herrmann. The single-day surgery scheduling problem: sequential decision-making and threshold-based heuristics. *OR spectrum*, 34(2):429–459, 2012.

[45] A. Jebali, Atidel B. Hadj A., and P. Ladet. Operating rooms scheduling. *International Journal of Production Economics*, 99(1):52–62, 2006.

[46] M. Lamiri, X. Xie, A. Dolgui, and F. Grimaud. A stochastic model for operating room planning with elective and emergency demand for surgery. *European Journal of Operational Research*, 185(3):1026–1037, 2008.

[47] M. Lamiri, X. Xie, and S. Zhang. Column generation approach to operating theater planning with elective and emergency patients. *IIE Transactions*, 40(9):838–852, 2008.

[48] P. Lebowitz. Schedule the short procedure first to improve or efficiency. *AORN journal*, 78(4):651–659, 2003.

[49] C. Mannino, E. J. Nilssen, and T. Eric. Nordlander. A pattern based, robust approach to cyclic master surgery scheduling. *Journal of Scheduling*, 15(5):553–563, 2012.

[50] E. Marcon and F. Dexter. Impact of surgical sequencing on post anesthesia care unit staffing. *Health Care Management Science*, 9(1):87–98, 2006.

[51] S. M. Johnson. Optimal two-and three-stage production schedules with setup times included. *Naval research logistics quarterly*, 1(1):61–68, 1954.

[52] E. Marcon, S. Kharraja, and G. Simonnet. The operating theatre planning by the follow-up of the risk of no realization. *International Journal of Production Economics*, 85(1):83–90, 2003.

[53] I. Marques, M. E. Captivo, and M. V. Pato. An integer programming approach to elective surgery scheduling. *OR spectrum*, 34(2):407–427, 2012.

[54] D. Min and Y. Yih. An elective surgery scheduling problem considering patient priority. *Computers and Operations Research*, 37(6):1091–1099, 2010.

[55] S. N. Ogulata and R. Erol. A hierarchical multiple criteria mathematical programming approach for scheduling general surgery operations in large hospitals. *Journal of Medical Systems*, 27(3):259–270, 2003.

[56] I. Ozkarahan. Allocation of surgeries to operating rooms by goal programing. *Journal of Medical Systems*, 24(6):339–378, 2000.

[57] H. Saadouli, B. Jerbi, A. Dammak, L. Masmoudi, and A. Bouaziz. A stochastic optimization and simulation approach for scheduling operating rooms and recovery beds in an orthopedic surgery department. *Computers & Industrial Engineering*, 80:72–79, 2015.

[58] A. Sciomachen, E. Tanfani, and A. Testi. Simulation models for optimal schedules of operating theatres. *International Journal of Simulation*, 6(12-13):26–34, 2005.

[59] A. Testi, E. Tanfani, and G. Torre. A three-phase approach for operating theatre schedules. *Health Care Management Science*, 10(2):163–172, 2007.

[60] D. C. Tyler, C. A. Pasquariello, and C.-H. Chen. Determining optimum operating room utilization. *Anesthesia & Analgesia*, 96(4):1114–1121, 2003.

[61] C. Van Huele and M. Vanhoucke. Analysis of the integration of the physician rostering problem and the surgery scheduling problem. *Journal of medical systems*, 38(6):1–16, 2014.

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

[63] F. Dexter. Operating room utilization: information management systems. *Current Opinion in Anesthesiology*, 16(6):619–622, 2003.

[64] F. Dexter and R. H. Epstein. Operating room efficiency and scheduling. *Current Opinion in Anesthesiology*, 18(2):195–198, 2005.

[65] Y. Gerchak, D. Gupta, and M. Henig. Reservation planning for elective surgery under uncertain demand for emergency surgery. *Management Science*, 42(3):321–334, 1996.

[66] J. R. Birge. Decomposition and partitioning methods for multistage stochastic linear programs. *Operations Research*, 33(5):989–1007, 1985.

[67] G. Laporte and F. V. Louveaux. The integer L-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters*, 13(3):133–142, 1993.

[68] V. I. Norkin, Y. M. Ermoliev, and A. Ruszczyński. On optimal allocation of indivisibles under uncertainty. *Operations Research*, 46(3):381–395, 1998.

[69] C. Mancilla and R. Storer. A sample average approximation approach to stochastic appointment sequencing and scheduling. *IIE Transactions*, 44(8):655–670, 2012.

[70] J. F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1):238–252, 1962.

[71] M. A. Makary, J. B. Sexton, J. A. Freischlag, E. A. Millman, D. Pryor, C. Holzmueller, and P. J. Pronovost. Patient safety in surgery. *Annals of Surgery*, 243(5):628, 2006.

[72] N. Zantuck, S. Mackay, and M. L. Wong. Surgical causes of upper abdominal pain. *Australian Family Physician*, 37(8):614, 2008.

[73] K. Bonewit-West, S. Hunt, and E. Applegate. *Today's Medical Assistant: Clinical & Administrative Procedures*. Saunders, St. Louis, MO, 3rd edition, 2014.

[74] Ministry of Health, New South Wales, Australia. Waiting time and elective surgery policy. `http://www0.health.nsw.gov.au/policies/pd/2012/pdf/PD2012_011.pdf`, 2012. Accessed: 20 Apr, 2016.

[75] National Confidential Enquiry into Patient Outcome and Death (NCEPOD), UK. The NCEPOD classification of intervention. `http://www.ncepod.org.uk/classification.html`, 2004. Accessed: 20 Apr, 2016.

[76] D. A. Mahler and C. K. Wells. Evaluation of clinical methods for rating dyspnea. *CHEST Journal*, 93(3):580–586, 1988.

[77] M. M. Oken, R. H. Creech, D. C. Tormey, J. Horton, T. E. Davis, E. T. Mcfadden, and P. P. Carbone. Toxicity and response criteria of the eastern cooperative oncology group. *American journal of clinical oncology*, 5(6):649–656, 1982.

[78] P. S. Kamath and W. Kim. The model for end-stage liver disease (MELD). *Hepatology*, 45(3):797–805, 2007.

[79] D. L. Patrick and R. A. Deyo. Generic and disease-specific measures in assessing health status and quality of life. *Medical Care*, 27(3):S217–S232, 1989.

[80] S. Ahmed and A. Shapiro. *The sample average approximation method for stochastic programs with integer recourse*. Technical Report, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atalanta, GA, 2002.

[81] G. Bayraksan and D. P. Morton. Assessing solution quality in stochastic programs. *Mathematical Programming*, 108(2-3):495–514, 2006.

[82] D. P. Strum, A. R. Sampson, J. H. May, and L. G. Vargas. Surgeon and type of anesthesia predict variability in surgical procedure times. *Anesthesiology*, 92(5):1454–1466, 2000.

[83] B. Berg, B. Denton, H. Nelson, H. Balasubramanian, A. Rahman, A. Bailey, and K. Lindor. A discrete event simulation model to evaluate operational performance of a colonoscopy suite. *Medical Decision Making*, 30(3):380–387, 2009.

[84] S. Gul, B. T. Denton, J. W. Fowler, and T. Huschka. Bi-criteria scheduling of surgical services for an outpatient procedure center. *Production and Operations Management*, 20(3):406–417, 2011.

[85] W. E. Spangler, D. P. Strum, L. G. Vargas, and J. H. May. Estimating procedure times for surgeries by determining location parameters for the lognormal model. *Health care management science*, 7(2):97–104, 2004.

[86] K. R. Baker. *Introduction to sequencing and scheduling*. John Wiley, New York, 1974.

[87] P. Hansen and N. Mladenović. First vs. best improvement: An empirical study. *Discrete Applied Mathematics*, 154(5):802–817, 2006.

[88] F. Glover. Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 13(5):533–549, 1986.

[89] M. Gendreau and J.-Y. Potvin. Metaheuristics in combinatorial optimization. *Annals of Operations Research*, 140(1):189–213, 2005.

[90] F. Glover. Tabu search-Uncharted domains. *Annals of Operations Research*, 149(1):89–98, 2007.

[91] C. C. Ribeiro and P. Hansen, editors. *Essays and surveys in metaheuristics*. Kluwer Academic Publishers, Boston, MA, 2002.

[92] M. Gendreau. An introduction to tabu search. In F. Glover and G. A. Kochenberger, editors, *Handbook of Metaheuristics*. Kluwer Academic Publishers, Boston, MA, 2003.

[93] J. R. Birge and F. Louveaux. *Introduction to stochastic programming*. Springer, New York, 1997.

[94] J. H. McDonald. *Handbook of biological statistics*. Sparky House Publishing, Baltimore, MD, 3rd edition, 2014.

[95] R. J. Blendon, C. M. DesRoches, M. Brodie, J. M. Benson, A. B. Rosen, E. Schneider, D. E. Altman, K. Zapert, M. J. Herrmann, and A. E. Steffenson. Views of practicing physicians and the public on medical errors. *New England Journal of Medicine*, 347(24):1933–1940, 2002.

[96] A. Shapiro. Sample average approximation. In *Encyclopedia of Operations Research and Management Science*, pages 1350–1355. Springer, 2013.

[97] F. C. Leone, L. S. Nelson, and R. B. Nottingham. The folded normal distribution. *Technometrics*, 3(4):543–550, 1961.

[98] S. Schiffer. Ceteris paribus laws. *Mind*, 100(1):1–17, 1991.

[99] M. M. Mukaka. A guide to appropriate use of correlation coefficient in medical research. *Malawi Medical Journal*, 24(3):69–71, 2012.

APPENDICES

APPENDIX A
PERFORMANCE OF SAA ALGORITHM (ELECTIVE SURGERY SCHEDULING WITH LINEARLY DETERIORATING PATIENT HEALTH CONDITION)

Table A.1
Computation times of SAA algorithm (all instances).

				Computation time (seconds) for each instance										
q	u	n	δ	1	2	3	4	5	6	7	8	9	10	Average
5	5	8	0.05	181.341	24.753	227.643	15.816	22.159	20.144	88.060	39.210	31.270	186.093	83.649
			0.1	120.940	40.741	229.741	20.987	22.507	25.472	71.993	41.821	40.739	139.819	75.476
			0.2	55.635	27.769	130.477	18.597	23.754	32.538	65.105	57.585	33.380	164.451	60.929
		10	0.05	321.710	450.133	2531.368	333.321	355.481	2323.166	1880.758	2650.369	5891.006	431.516	1716.883
			0.1	306.030	438.815	1412.133	532.668	417.995	1623.054	1667.437	2116.416	3085.823	578.321	1217.869
			0.2	326.030	230.835	1509.747	339.582	481.641	1310.193	1390.272	1322.649	3007.024	584.215	1050.219
	10	8	0.05	357.956	53.569	456.328	42.978	51.428	61.038	168.624	95.967	71.620	421.696	178.120
			0.1	244.533	70.181	450.358	48.225	54.007	74.196	200.308	97.903	86.167	455.964	178.184
			0.2	120.126	60.320	315.208	45.352	53.569	92.983	162.147	130.593	83.408	413.870	147.758
		10	0.05	544.682	1219.509	4089.155	805.499	506.738	4230.735	3085.782	5695.406	8244.698	851.331	2927.354
			0.1	498.608	892.015	2068.049	922.526	516.967	3476.921	3035.539	3627.460	4472.538	1104.274	2061.490
			0.2	414.510	838.364	2060.194	881.272	1020.663	2377.277	1935.939	2365.165	4351.246	967.011	1721.164
10	5	8	0.05	293.172	110.407	557.203	49.616	48.819	41.927	185.902	105.079	47.990	705.204	214.532
			0.1	199.808	83.272	441.391	63.148	51.366	47.647	174.275	129.469	73.205	1175.961	243.954
			0.2	143.597	160.357	339.220	80.830	47.695	84.418	175.062	113.642	74.621	841.182	206.062
		10	0.05	591.206	1132.564	7141.613	964.514	915.802	5226.302	5157.630	3532.894	10952.100	1025.178	3663.980
			0.1	520.300	1232.693	2401.573	1086.193	1057.737	3243.060	2657.279	3532.241	7000.968	1005.359	2373.740
			0.2	560.448	879.072	1393.657	878.018	689.015	4250.163	2222.539	2447.276	4354.498	1210.144	1888.483
	10	8	0.05	806.224	156.978	1501.199	134.704	136.001	108.562	502.096	321.383	145.242	1494.539	530.693
			0.1	692.837	134.845	1214.732	187.040	146.319	169.850	473.897	336.833	195.594	1907.935	545.988
			0.2	423.084	190.542	1133.538	165.582	161.640	189.207	470.280	254.451	223.369	1776.793	498.849
		10	0.05	1270.995	2805.359	12932.600	1790.026	1732.888	10940.600	8827.412	7704.428	18905.500	2388.338	6929.815
			0.1	1114.485	2140.684	5829.451	2265.343	3009.461	8283.681	5869.558	7602.265	16350.900	2336.024	5480.185
			0.2	1052.305	2138.179	3400.560	2083.648	2280.505	8867.416	5872.861	6012.163	12852.500	3221.357	4778.149
20	5	8	0.05	675.195	113.890	1348.568	416.126	438.819	104.751	619.984	537.459	505.363	1555.443	631.560
			0.1	638.629	183.970	1061.654	513.586	378.275	137.849	593.603	520.805	486.564	1423.274	593.821
			0.2	433.236	176.183	1211.787	508.949	425.947	150.597	563.893	388.234	402.966	1612.105	587.390
		10	0.05	1630.604	2542.025	6488.479	5541.384	2078.355	12297.300	13820.100	7727.314	28806.000	12671.100	9360.266
			0.1	1296.204	2016.355	5425.773	15446.900	1658.682	12752.400	7293.387	5013.725	17226.400	15930.200	8406.003
			0.2	1140.417	2126.810	4207.568	3637.753	3452.940	6833.046	6839.512	5777.662	13763.100	11572.100	5935.091
	10	8	0.05	1369.083	259.288	2868.026	1067.133	959.646	223.586	1272.731	986.806	955.694	4401.667	1436.366
			0.1	1257.574	281.373	2244.563	1092.202	658.085	286.731	1184.260	1062.581	724.110	2083.294	1087.477
			0.2	935.602	296.736	2299.642	1119.493	918.173	444.524	1121.823	957.623	986.360	3364.842	1244.482
		10	0.05	3159.211	4771.603	14904.300	12494.400	6417.575	29488.800	19910.000	15611.600	41622.400	31007.200	17938.709
			0.1	2704.642	4058.410	8473.702	22857.800	6010.814	18462.600	15396.300	10582.900	34859.500	31112.200	15451.887
			0.2	2370.693	4205.990	9445.770	10042.700	6643.884	16003.600	12532.500	10930.700	24555.900	26357.600	12308.934

Table A.2
95% CIOOGs of SAA solutions (all instances).

				95% CIOOG of SAA solution for each instance										
q	u	n	δ	1	2	3	4	5	6	7	8	9	10	Average
5	5	8	0.05	1.562	0.300	0.177	0.393	0.491	0.000	0.000	1.318	0.457	0.000	0.470
			0.1	1.404	0.874	0.000	0.785	0.981	0.000	0.000	0.456	1.520	0.000	0.602
			0.2	2.933	2.086	0.000	2.872	1.962	1.357	0.000	0.913	3.977	1.258	1.736
		10	0.05	1.920	0.000	0.821	0.018	2.311	2.232	0.000	0.672	0.000	0.395	0.837
			0.1	10.319	0.000	1.514	0.000	2.147	4.250	0.000	0.405	0.000	1.511	2.015
			0.2	10.319	0.535	2.480	0.071	8.745	4.715	0.070	3.434	0.000	4.684	3.505
	10	8	0.05	1.693	0.161	0.289	0.433	0.213	0.000	0.000	0.000	0.804	0.000	0.359
			0.1	1.094	0.359	0.000	0.865	0.570	0.523	0.000	0.000	1.774	0.482	0.567
			0.2	2.443	1.114	0.000	2.595	0.852	0.839	0.000	0.959	3.958	1.860	1.462
		10	0.05	1.644	0.000	1.293	0.000	0.844	1.626	0.000	0.681	0.000	0.210	0.630
			0.1	0.431	0.000	1.070	0.002	1.514	3.149	0.000	1.878	0.000	0.652	0.870
			0.2	2.219	1.894	1.150	0.102	4.255	3.819	1.794	2.057	0.000	3.045	2.034
10	5	8	0.05	1.213	0.076	0.119	0.296	0.194	0.000	0.000	0.000	0.605	0.200	0.270
			0.1	0.661	0.152	0.000	0.592	0.388	0.312	0.000	0.000	1.236	0.401	0.374
			0.2	1.498	1.012	0.000	1.185	0.776	0.283	0.000	0.000	2.861	0.000	0.761
		10	0.05	0.373	0.000	0.726	0.000	0.098	0.743	0.000	0.367	0.000	0.080	0.239
			0.1	0.000	0.000	0.000	0.000	0.382	1.951	0.000	1.638	0.000	0.160	0.413
			0.2	0.000	1.683	0.000	0.000	1.985	2.395	0.802	2.067	0.000	0.320	0.925
	10	8	0.05	1.137	0.306	0.280	0.229	0.247	0.000	0.000	0.000	0.409	0.100	0.271
			0.1	0.937	0.977	0.000	0.459	0.494	0.156	0.000	0.000	0.849	0.984	0.486
			0.2	2.325	0.906	0.000	0.917	0.988	1.108	0.000	0.000	1.915	0.478	0.864
		10	0.05	0.812	0.000	0.414	0.000	0.140	0.655	0.000	0.218	0.101	0.194	0.253
			0.1	0.000	0.000	1.445	0.000	0.347	1.296	0.000	0.892	0.220	0.387	0.459
			0.2	2.321	1.694	4.382	0.000	1.483	1.303	0.401	1.209	0.460	0.774	1.403
20	5	8	0.05	0.745	0.201	0.009	0.136	0.048	0.000	0.000	0.000	0.342	0.100	0.158
			0.1	0.439	1.075	0.000	0.271	0.096	0.000	0.000	0.000	0.685	0.200	0.277
			0.2	0.958	0.383	0.000	0.542	0.193	0.000	0.000	0.000	1.433	0.401	0.391
		10	0.05	0.692	0.000	0.411	0.000	0.096	0.591	0.000	0.093	0.088	0.112	0.208
			0.1	0.000	0.000	1.629	0.000	0.193	0.876	0.000	0.449	0.213	0.223	0.358
			0.2	1.722	1.217	3.597	0.000	1.012	0.556	0.000	0.391	0.463	0.504	0.946
	10	8	0.05	0.702	0.181	0.002	0.075	0.213	0.000	0.000	0.000	0.171	0.077	0.142
			0.1	0.626	0.672	0.000	0.150	0.426	0.000	0.000	0.000	0.342	0.154	0.237
			0.2	1.340	0.504	0.000	0.384	0.852	0.703	0.000	0.000	0.726	0.308	0.482
		10	0.05	0.541	0.000	0.406	0.000	0.124	0.564	0.000	0.213	0.044	0.158	0.205
			0.1	0.000	0.000	1.110	0.000	0.464	1.339	0.000	0.503	0.106	0.317	0.384
			0.2	0.861	1.867	1.872	0.000	1.056	0.733	0.009	0.461	0.231	0.640	0.773

Table A.3
Objective values (4.26) of SAA solutions (all instances).

				Objective value (4.26) for each instance										
q	u	n	δ	1	2	3	4	5	6	7	8	9	10	Average
5	5	8	0.05	1.448	13.455	-3.645	38.432	11.814	16.931	11.764	20.542	22.847	18.294	15.188
			0.1	-23.465	-5.146	-39.720	37.864	-6.372	2.862	-14.484	4.580	10.813	2.616	-3.045
			0.2	-72.666	-42.293	-111.871	36.727	-42.744	-25.804	-66.980	-27.840	-13.256	-29.518	-39.624
		10	0.05	8.812	-18.801	1.390	24.116	28.835	8.750	-0.293	6.247	16.232	20.892	9.618
			0.1	-3.790	-64.602	-26.919	11.236	18.973	-9.389	-43.553	-13.733	-15.198	9.739	-13.724
			0.2	-27.843	-156.206	-82.040	-14.537	-3.015	-42.579	-130.071	-53.693	-78.057	-13.523	-60.156
	10	8	0.05	1.136	13.455	-3.645	38.432	12.134	16.931	11.764	20.988	22.215	18.357	15.177
			0.1	-23.465	-5.090	-39.720	37.864	-5.732	2.862	-14.484	4.976	9.550	2.616	-3.062
			0.2	-72.666	-42.127	-111.871	36.727	-41.464	-25.804	-66.980	-27.840	-15.781	-29.518	-39.733
		10	0.05	7.910	-18.801	1.390	24.173	28.953	8.750	-0.293	6.247	16.232	20.892	9.545
			0.1	-3.790	-64.602	-26.401	11.236	18.973	-8.952	-43.553	-13.733	-15.198	9.955	-13.606
			0.2	-27.843	-156.206	-81.228	-14.537	-0.653	-41.753	-130.071	-52.154	-78.057	-12.563	-59.506
10	5	8	0.05	1.136	13.468	-3.645	38.432	12.134	16.931	11.764	20.988	22.215	18.294	15.172
			0.1	-23.465	-5.064	-39.720	37.864	-5.732	2.862	-14.484	4.976	9.550	2.587	-3.063
			0.2	-72.666	-42.180	-111.871	36.727	-41.464	-25.776	-66.980	-27.048	-15.781	-28.570	-39.561
		10	0.05	8.236	-18.801	1.398	24.173	28.953	8.753	-0.293	6.247	16.232	21.120	9.602
			0.1	-3.790	-64.602	-26.401	11.347	18.973	-8.881	-43.553	-13.733	-15.198	10.239	-13.560
			0.2	-27.843	-156.206	-81.228	-14.306	-0.248	-42.105	-130.071	-52.154	-78.057	-11.522	-59.374
	10	8	0.05	1.527	13.468	-3.645	38.432	11.814	16.931	11.764	20.988	22.847	18.294	15.242
			0.1	-23.465	-5.064	-39.720	37.864	-6.372	2.862	-14.484	4.976	10.813	2.587	-3.000
			0.2	-73.973	-42.365	-111.871	36.727	-42.744	-25.776	-66.980	-27.048	-13.256	-28.570	-39.586
		10	0.05	8.236	-18.801	1.398	24.173	28.953	8.750	-0.293	6.247	16.232	21.120	9.601
			0.1	-3.790	-64.602	-26.927	11.347	18.973	-8.881	-43.553	-13.733	-15.198	10.239	-13.613
			0.2	-27.843	-156.206	-81.228	-14.306	-0.248	-42.105	-130.071	-52.154	-78.057	-11.522	-59.374
20	5	8	0.05	1.448	13.468	-3.645	38.432	11.814	16.931	11.764	20.988	22.847	18.294	15.234
			0.1	-23.465	-5.064	-39.720	37.864	-6.372	2.862	-14.484	4.976	10.813	2.587	-3.000
			0.2	-73.973	-42.419	-111.871	36.727	-42.744	-25.776	-66.980	-27.048	-13.256	-28.825	-39.616
		10	0.05	8.236	-18.801	1.390	24.173	28.953	8.750	-0.293	6.247	16.232	21.120	9.601
			0.1	-3.790	-64.602	-26.927	11.347	18.973	-8.881	-43.553	-13.733	-15.198	10.239	-13.613
			0.2	-27.843	-156.206	-81.228	-14.306	-0.248	-42.105	-130.071	-52.154	-78.057	-11.522	-59.374
	10	8	0.05	1.448	13.468	-3.645	38.432	11.814	16.931	11.764	20.988	22.847	18.357	15.240
			0.1	-23.465	-5.090	-39.720	37.864	-6.372	2.862	-14.484	4.976	10.813	2.715	-2.990
			0.2	-72.666	-42.365	-111.871	36.727	-42.744	-25.776	-66.980	-27.048	-13.256	-28.570	-39.455
		10	0.05	8.236	-18.801	1.398	24.173	29.036	8.750	-0.293	6.247	16.232	21.065	9.604
			0.1	-3.790	-64.602	-26.919	11.347	19.140	-8.874	-43.553	-13.733	-15.198	10.130	-13.605
			0.2	-27.843	-156.206	-81.228	-14.306	-0.248	-41.851	-130.071	-52.154	-78.057	-11.740	-59.370

APPENDIX B

PERFORMANCE OF EVM (ELECTIVE SURGERY SCHEDULING WITH LINEARLY DETERIORATING PATIENT HEALTH CONDITION)

Table B.1
Computation times to solve EVMs (all instances).

		Computation time (seconds) for each instance										
n	δ	1	2	3	4	5	6	7	8	9	10	Average
8	0.05	2.164	0.854	1.230	0.638	0.923	0.888	1.272	0.790	1.142	0.547	1.045
	0.1	3.300	0.754	1.306	0.635	1.358	1.149	1.645	0.892	1.629	0.921	1.359
	0.2	3.595	0.885	2.855	0.531	1.224	3.833	1.091	1.015	1.641	0.752	1.742
10	0.05	5.614	16.275	5.533	1.696	3.203	9.970	7.227	5.830	5.476	1.306	6.213
	0.1	4.191	15.160	3.870	1.773	2.293	6.316	2.670	8.640	8.472	1.345	5.473
	0.2	5.538	16.196	3.929	1.788	3.204	3.681	3.097	3.293	7.180	1.685	4.959
15	0.05	111.718	111.835	616.174	106.546	151.152	64.130	157.712	112.932	64.877	170.977	166.805
	0.1	133.969	553.034	776.040	85.794	291.204	86.945	222.422	214.217	202.552	227.218	279.340
	0.2	373.550	296.374	651.320	146.609	601.338	122.169	175.061	154.185	343.847	171.542	303.600
20	0.05	1776.657	2137.340	2031.126	9213.350	2907.310	2123.424	1496.210	1860.233	25363.036	7829.986	5673.867
	0.1	2891.058	2166.567	1413.514	6498.261	2700.519	3852.332	5819.370	2576.870	1456.003	4658.383	3403.288
	0.2	1795.593	15064.438	1183.729	2345.358	3264.431	3651.541	3217.076	2473.027	2035.840	2204.273	3723.531

Table B.2
Objective values (4.26) of EVM solutions (all instances).

		Objective value (4.26) of EVM solution for each instance										
n	δ	1	2	3	4	5	6	7	8	9	10	Average
8	0.05	1.833	12.756	-3.645	37.029	11.840	16.856	11.764	18.469	21.374	16.877	14.515
	0.1	-24.279	-5.225	-39.720	35.059	-8.480	2.862	-17.890	1.950	8.794	-0.391	-4.732
	0.2	-74.197	-44.979	-111.871	31.117	-44.667	-25.574	-81.326	-34.684	-17.292	-33.181	-43.665
10	0.05	8.454	-18.801	0.527	23.219	28.351	7.811	-0.293	5.412	16.232	18.912	8.982
	0.1	-4.342	-64.602	-30.160	9.313	17.331	-9.514	-43.553	-14.892	-18.664	8.041	-15.104
	0.2	-29.792	-156.206	-84.814	-14.528	-5.244	-48.299	-130.071	-60.496	-82.902	-27.800	-64.015
15	0.05	-18.061	-25.286	-75.224	-4.319	-54.327	3.488	-14.761	19.574	2.305	-35.016	-20.163
	0.1	-80.570	-77.515	-180.784	-46.341	-137.306	-19.007	-60.690	-3.913	-21.102	-106.363	-73.359
	0.2	-196.597	-179.904	-391.553	-122.109	-305.212	-64.839	-146.113	-52.825	-80.634	-253.927	-179.371
20	0.05	-1.969	-41.939	-31.852	-0.085	-80.560	-57.130	-70.127	-11.083	-69.748	-84.693	-44.919
	0.1	-21.808	-100.011	-77.298	-24.177	-182.633	-133.314	-163.355	-44.628	-130.644	-169.480	-104.735
	0.2	-61.311	-210.518	-158.661	-67.052	-385.871	-287.167	-340.885	-126.067	-255.856	-331.925	-222.531

APPENDIX C

PERFORMANCE OF FALS (ELECTIVE SURGERY SCHEDULING WITH LINEARLY DETERIORATING PATIENT HEALTH CONDITION)

Table C.1
Computation times of FALS (all instances).

		Computation time†(seconds) for each instance										
n	δ	1	2	3	4	5	6	7	8	9	10	Average
8	0.05	0.339	0.663	0.529	0.779	0.721	1.102	0.554	1.889	1.569	0.634	0.878
	0.1	0.570	0.664	0.467	0.787	0.610	0.599	1.393	1.101	1.245	0.859	0.830
	0.2	0.572	0.641	0.467	0.787	1.051	0.530	1.504	1.670	1.250	1.789	1.026
10	0.05	3.017	0.932	1.756	2.618	2.609	1.915	1.176	1.058	1.342	4.363	2.079
	0.1	3.027	0.857	3.159	2.579	4.046	1.872	1.106	1.104	2.033	2.453	2.224
	0.2	3.712	0.938	1.015	1.274	1.208	3.604	1.139	2.429	1.980	6.259	2.356
15	0.05	9.230	51.437	4.422	17.071	16.420	9.358	17.834	14.347	71.057	8.422	21.960
	0.1	20.231	30.802	7.599	31.586	12.956	5.575	32.232	30.003	50.526	11.387	23.290
	0.2	10.225	7.937	4.424	22.049	21.170	28.867	21.478	29.493	51.797	11.583	20.902
20	0.05	198.951	68.029	581.419	37.545	64.134	60.766	76.604	297.907	122.200	125.847	163.340
	0.1	257.792	105.806	555.898	87.652	59.047	87.825	183.590	197.924	154.690	91.790	178.201
	0.2	105.036	143.760	376.075	248.086	82.637	88.143	197.287	254.240	94.145	125.382	171.479

† computation time after obtaining an initial solution by solving the EVM.

Table C.2
Objective values (4.26) of FALS solutions (all instances).

		Objective value (4.26) of FALS solution for each instance										
n	δ	1	2	3	4	5	6	7	8	9	10	Average
8	0.05	1.833	13.455	-3.645	38.432	11.814	16.931	11.764	20.790	22.847	17.804	15.202
	0.1	-23.465	-5.160	-39.720	37.864	-8.480	2.862	-14.484	2.901	10.813	1.600	-3.527
	0.2	-72.666	-42.180	-111.871	36.727	-42.744	-25.574	-66.980	-27.840	-13.256	-28.570	-39.495
10	0.05	8.129	-18.801	1.083	24.173	29.036	8.820	-0.293	5.412	16.232	21.065	9.486
	0.1	-3.790	-64.602	-27.549	10.311	19.140	-8.881	-43.553	-14.892	-15.198	9.730	-13.928
	0.2	-27.843	-156.206	-84.814	-14.537	-5.244	-42.358	-130.071	-53.693	-78.057	-12.110	-60.493
15	0.05	-18.058	-21.790	-75.224	-2.290	-53.777	3.612	-13.966	20.495	6.138	-33.871	-18.873
	0.1	-74.376	-74.354	-180.667	-41.959	-136.086	-19.007	-55.576	-1.487	-18.026	-106.153	-70.769
	0.2	-193.174	-178.196	-391.553	-117.865	-302.299	-57.095	-138.785	-46.955	-69.396	-250.656	-174.597
20	0.05	-0.826	-41.120	-27.834	-0.085	-80.010	-56.339	-68.520	-8.833	-68.449	-82.488	-43.451
	0.1	-19.391	-97.368	-68.840	-22.082	-180.138	-130.248	-159.328	-42.619	-129.294	-164.882	-101.419
	0.2	-53.562	-204.293	-150.267	-61.831	-381.236	-278.677	-331.426	-113.374	-252.251	-325.040	-215.196

APPENDIX D

PERFORMANCE OF TS (ELECTIVE SURGERY SCHEDULING WITH LINEARLY DETERIORATING PATIENT HEALTH CONDITION)

Table D.1
Computation times of TS (all instances).

		Computation time†(seconds) for each instance										
n	δ	1	2	3	4	5	6	7	8	9	10	Average
8	0.05	968.359	1011.224	1030.542	1306.679	956.824	1185.527	1099.796	1156.617	1344.643	1245.095	1130.531
	0.1	1204.919	1075.554	845.075	1332.762	48.558	236.777	69.832	1398.420	1497.662	1314.694	902.425
	0.2	1604.779	1067.688	1190.517	1296.584	1389.434	1187.688	1129.061	1152.708	1405.623	1212.626	1263.671
10	0.05	3054.865	2531.273	112.849	3137.687	3624.360	288.507	618.589	3448.480	815.738	3682.703	2131.505
	0.1	3646.515	107.869	3153.156	3474.665	482.610	168.416	245.364	2545.554	1682.401	352.156	1585.871
	0.2	3610.285	3154.074	2830.710	3357.050	3671.432	752.900	2833.969	3954.254	3909.677	3502.557	3157.691
15	0.05	296.315	2148.771	118.463	1054.695	649.531	302.183	2982.733	3244.757	813.007	163.090	1177.355
	0.1	192.804	125.903	217.767	1453.319	453.670	101.721	582.679	1395.324	1626.044	217.780	636.701
	0.2	213.577	826.155	937.832	203.323	950.428	718.913	519.172	587.400	968.989	83.144	600.893
20	0.05	2522.259	388.412	4723.522	954.646	448.470	276.777	4915.592	1248.780	655.723	701.227	1683.541
	0.1	414.985	931.810	512.500	2227.918	3944.395	3603.628	622.336	2477.794	2166.249	597.053	1749.867
	0.2	425.219	1777.297	1948.498	776.871	342.449	365.185	415.463	6451.159	168.720	537.740	1320.860

† computation time after obtaining an initial solution by solving the EVM.

Table D.2
Objective values (4.26) of TS solutions (all instances).

		Objective value (4.26) of TS solution for each instance										
n	δ	1	2	3	4	5	6	7	8	9	10	Average
8	0.05	1.833	13.455	-3.645	38.432	11.814	16.931	11.764	20.988	22.847	18.357	15.278
	0.1	-23.465	-5.160	-39.720	37.864	-6.372	2.862	-14.484	4.976	10.813	2.715	-2.997
	0.2	-72.666	-42.366	-111.871	36.727	-42.744	-25.776	-66.980	-27.048	-13.256	-28.570	-39.455
10	0.05	8.236	-18.801	1.398	24.173	29.036	8.820	-0.293	6.247	16.232	21.065	9.611
	0.1	-3.790	-64.602	-26.927	11.347	19.140	-8.881	-43.553	-13.733	-15.198	10.130	-13.607
	0.2	-27.843	-156.206	-81.426	-14.306	-0.248	-41.851	-130.071	-52.154	-78.057	-11.740	-59.390
15	0.05	-17.670	-21.790	-75.224	-1.716	-53.777	5.536	-13.891	20.607	6.219	-33.871	-18.558
	0.1	-72.254	-74.354	-180.667	-40.434	-136.086	-14.228	-55.576	-0.786	-18.662	-106.153	-69.920
	0.2	-184.241	-176.962	-391.553	-117.865	-301.072	-50.295	-134.118	-44.076	-67.601	-250.697	-171.848
20	0.05	-0.283	-40.982	-27.648	-0.085	-80.092	-56.339	-68.520	-6.808	-67.957	-82.488	-43.120
	0.1	-17.319	-97.368	-68.568	-17.986	-176.208	-130.224	-159.328	-42.619	-129.218	-164.882	-100.372
	0.2	-53.470	-203.557	-149.986	-58.430	-380.008	-278.665	-331.426	-108.992	-252.251	-324.838	-214.162

APPENDIX E

PERFORMANCE OF SAA ALGORITHM (ELECTIVE SURGERY SCHEDULING WITH STEP- DETERIORATING PATIENT HEALTH CONDITION)

Table E.1
Computation times of SAA algorithm (all instances).

		Computation time (seconds) for each instance										
q	*u*	1	2	3	4	5	6	7	8	9	10	Average
2	10	2.140	1.953	2.109	2.250	2.235	2.047	3.154	3.367	2.141	3.559	2.496
4	10	2.953	2.125	2.203	2.188	2.016	2.063	3.307	3.435	3.444	3.675	2.741
6	10	2.500	2.313	2.485	2.297	2.301	2.235	3.351	3.617	3.611	3.907	2.862
8	10	2.735	2.391	2.641	3.109	2.490	3.182	3.423	3.413	3.454	3.945	3.078
10	10	2.828	2.485	2.625	2.922	2.372	2.297	3.819	3.501	3.751	4.074	3.067
12	10	2.782	2.641	3.251	3.078	2.484	3.385	3.590	3.613	3.981	4.270	3.308
14	10	3.000	2.860	2.938	3.437	2.875	3.287	3.644	4.199	4.111	4.297	3.465
16	10	2.937	2.735	3.453	3.704	2.985	3.078	3.793	3.893	4.052	5.775	3.641
18	10	3.563	2.751	3.641	4.329	3.453	3.282	4.070	4.266	4.152	4.798	3.831
20	10	3.844	3.547	4.125	3.829	4.469	3.422	4.125	4.418	4.459	5.001	4.124
2	20	4.735	4.156	4.469	4.750	4.657	4.297	6.553	6.718	4.844	7.018	5.220
4	20	4.844	4.485	5.125	5.016	5.007	4.532	7.064	6.891	7.116	8.529	5.861
6	20	5.485	5.501	4.860	5.766	5.763	5.298	7.062	7.334	7.468	7.717	6.225
8	20	7.298	5.706	6.641	6.907	6.604	5.985	7.546	7.162	6.767	8.547	6.916
10	20	7.798	6.548	7.220	7.845	6.469	7.110	7.650	7.658	8.135	9.204	7.564
12	20	7.813	7.000	7.673	7.566	6.900	7.016	8.649	8.412	8.348	9.642	7.902
14	20	8.360	7.688	7.594	7.704	6.891	7.091	9.936	9.280	8.485	10.333	8.336
16	20	7.485	7.673	7.845	8.016	6.958	7.257	8.657	8.624	10.375	10.617	8.351
18	20	7.579	7.126	9.563	9.114	7.260	7.809	8.548	8.924	10.012	11.545	8.748
20	20	8.610	7.204	8.392	10.219	7.172	8.837	9.522	9.751	11.101	12.079	9.289

Table E.2
95% CIOOGs of SAA solutions (all instances)

		95% CIOOG of SAA solution for each instance										
q	*u*	1	2	3	4	5	6	7	8	9	10	Average
2	10	1.412	1.302	1.099	0.985	2.143	0.166	1.030	0.403	1.684	1.445	1.167
4	10	0.184	0.617	0.595	1.246	0.467	0.000	0.124	0.457	1.149	0.753	0.559
6	10	0.158	0.360	0.483	0.332	0.118	0.135	0.000	0.025	0.706	0.640	0.296
8	10	0.233	0.226	0.455	0.287	0.246	0.063	0.000	0.134	0.616	0.432	0.269
10	10	0.194	0.055	0.414	0.243	0.182	0.050	0.000	0.038	0.419	0.244	0.184
12	10	0.162	0.036	0.316	0.115	0.153	0.000	0.000	0.028	0.282	0.208	0.130
14	10	0.149	0.095	0.308	0.302	0.125	0.037	0.000	0.205	0.332	0.014	0.157
16	10	0.090	0.078	0.273	0.288	0.123	0.032	0.000	0.103	0.187	0.187	0.136
18	10	0.111	0.021	0.181	0.319	0.070	0.000	0.000	0.072	0.218	0.068	0.106
20	10	0.077	0.000	0.181	0.210	0.087	0.000	0.000	0.017	0.127	0.035	0.073
2	20	0.559	0.945	0.679	1.337	0.650	0.125	0.436	0.775	1.566	1.263	0.834
4	20	0.198	0.477	0.529	0.552	0.264	0.103	0.000	0.362	1.277	0.643	0.441
6	20	0.146	0.285	0.439	0.369	0.198	0.245	0.104	0.012	0.586	0.526	0.291
8	20	0.129	0.167	0.383	0.480	0.159	0.033	0.000	0.014	0.450	0.376	0.219
10	20	0.113	0.028	0.340	0.462	0.092	0.047	0.000	0.104	0.294	0.237	0.172
12	20	0.099	0.026	0.287	0.282	0.077	0.000	0.000	0.014	0.239	0.203	0.123
14	20	0.088	0.048	0.229	0.258	0.065	0.005	0.000	0.104	0.346	0.159	0.130
16	20	0.070	0.039	0.188	0.286	0.070	0.000	0.000	0.009	0.203	0.127	0.099
18	20	0.064	0.011	0.116	0.273	0.006	0.000	0.000	0.036	0.184	0.078	0.077
20	20	0.091	0.013	0.208	0.189	0.028	0.000	0.000	0.013	0.127	0.096	0.076

Table E.3
Objective values (5.16) of SAA solutions (all instances).

		Objective value (5.16) of SAA solution for each instance										
q	*u*	1	2	3	4	5	6	7	8	9	10	Average
2	10	29.028	25.251	31.981	29.635	30.553	29.393	28.207	30.121	27.017	25.443	28.663
4	10	30.049	25.432	32.478	29.611	30.984	29.415	28.396	30.418	27.125	25.435	28.934
6	10	29.990	25.432	32.537	30.146	31.023	29.425	28.598	30.581	27.320	25.438	29.049
8	10	30.112	25.432	32.608	30.120	31.020	29.491	28.590	30.415	27.352	25.443	29.058
10	10	30.111	25.432	32.669	30.120	31.054	29.476	28.602	30.514	27.320	25.443	29.074
12	10	30.111	25.426	32.692	30.146	31.069	29.513	28.600	30.578	27.382	25.443	29.096
14	10	30.111	25.426	32.707	30.120	31.062	29.476	28.598	30.573	27.374	25.443	29.089
16	10	30.139	25.432	32.710	30.120	31.059	29.492	28.608	30.552	27.374	25.443	29.093
18	10	30.139	25.432	32.721	30.146	31.044	29.513	28.598	30.584	27.374	25.443	29.099
20	10	30.049	25.432	32.719	30.146	31.060	29.513	28.608	30.578	27.382	25.443	29.093
2	20	29.617	25.245	32.454	29.635	31.022	29.343	28.312	30.121	27.017	25.443	28.821
4	20	30.049	25.432	32.548	30.120	30.998	29.201	28.590	30.573	27.168	25.435	29.011
6	20	30.091	25.432	32.679	30.120	31.042	29.425	28.590	30.581	27.352	25.443	29.075
8	20	30.111	25.432	32.719	30.120	31.049	29.492	28.600	30.544	27.352	25.443	29.086
10	20	30.139	25.432	32.721	30.146	31.054	29.492	28.608	30.567	27.374	25.443	29.098
12	20	30.139	25.432	32.710	30.146	31.069	29.513	28.600	30.578	27.382	25.443	29.101
14	20	30.139	25.432	32.721	30.146	31.062	29.484	28.598	30.573	27.382	25.443	29.098
16	20	30.139	25.432	32.719	30.146	31.059	29.513	28.600	30.581	27.382	25.443	29.101
18	20	30.139	25.432	32.721	30.146	31.069	29.513	28.600	30.584	27.382	25.443	29.103
20	20	30.127	25.426	32.719	30.146	31.045	29.513	28.608	30.578	27.382	25.443	29.099

APPENDIX F

BLOCK DAYS ON WHICH EACH PATIENT IS SCHEDULED FOR SURGERY (ELECTIVE SURGERY SCHEDULING WITH STEP-DETERIORATING PATIENT HEALTH CONDITION)

Table F.1
Block days on which each patient is scheduled for surgery based on his/her H_{j1} when the increment of H_{j1} is 4 (all instances under full-day block system).

		Block day on which each patient is scheduled for surgery for each instance																			
j	H_{j1}	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	30	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	34	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	38	1	2	1	2	2	2	2	2	2	1	2	2	2	2	2	2	2	2	2	1
4	42	2	2	2	2	2	2	2	2	1	2	2	1	2	2	2	2	2	2	2	2
5	46	2	1	2	3	3	1	3	1	2	3	3	2	3	4	3	3	3	3	3	3
6	50	3	3	3	4	3	4	4	2	4	2	4	2	3	1	4	4	4	2	2	3
7	54	2	5	2	4	3	3	3	4	3	5	1	3	4	3	6	4	4	4	4	2
8	58	3	4	6	4	5	3	2	3	4	4	4	5	4	5	4	6	5	4	5	4
9	62	4	5	5	6	6	4	6	3	6	4	5	5	5	5	6	2	6	5	5	5
10	66	4	3	5	5	4	6	6	6	5	4	3	7	2	4	3	3	4	6	1	6
11	70	5	3	4	7	4	8	7	4	6	7	6	4	7	7	5	5	6	5	4	5
12	74	5	4	7	1	2	6	5	7	4	3	4	6	7	8	5	5	3	3	6	5
13	78	6	6	4	6	7	4	5	3	3	7	7	4	6	5	7	1	6	6	3	4
14	82	6	7	3	5	1	5	4	6	7	5	5	6	6	6	6	6	3	6	5	7
15	86	7	6	7	3	5	7	7	5	6	7	6	8	5	6	4	1	7	3	6	4
16	90	4	7	6	3	6	5	4	5	5	6	7	3	5	3	7	3	5	8	6	6

Table F.2

Block days on which each patient is scheduled for surgery based on his/her H_{j1} when the increment of H_{j1} is 4 (all instances under half-day block system).

		Block day on which each patient is scheduled for surgery for each instance																			
j	H_{j1}	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	30	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	34	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	38	2	1	1	2	2	2	1	2	2	2	2	2	2	1	2	2	2	2	1	2
4	42	2	2	2	2	2	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2
5	46	3	3	2	3	3	2	2	2	3	3	3	2	3	3	3	3	3	4	3	3
6	50	3	3	4	3	4	4	4	3	4	4	4	3	3	3	4	4	3	2	4	4
7	54	5	5	5	4	4	4	3	4	5	5	2	3	4	4	5	4	4	3	3	1
8	58	4	5	4	5	1	5	6	3	5	6	4	6	5	2	5	6	5	5	2	5
9	62	6	4	6	6	3	3	4	4	6	3	5	4	6	5	3	3	6	6	6	6
10	66	5	3	3	5	5	6	5	6	3	4	7	7	7	7	7	7	1	5	5	8
11	70	7	7	8	7	6	8	6	5	6	8	6	6	7	7	6	5	6	3	7	6
12	74	4	6	7	5	7	8	4	8	7	5	5	5	8	5	8	8	4	4	5	4
13	78	8	5	3	8	6	6	8	8	4	1	8	4	4	6	6	8	7	8	8	7
14	82	6	8	8	8	5	3	5	7	7	7	3	8	5	5	4	5	7	8	6	7
15	86	8	8	6	6	7	7	3	6	8	8	6	5	8	6	8	1	5	6	4	8
16	90	7	5	5	4	7	5	7	7	8	6	8	7	6	8	7	6	8	7	4	6

Table F.3

Block days on which each patient is scheduled for surgery based on his/her H_{j1} when the increment of H_{j1} is 2 (all instances under full-day block system).

j	H_{j1}	Block day on which each patient is scheduled for surgery for each instance																			
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	60	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	62	1	1	1	2	1	1	1	1	1	2	1	1	1	2	1	1	1	1	1	1
3	64	1	1	2	2	2	2	1	2	1	2	2	1	2	1	2	1	2	2	1	1
4	66	2	2	2	1	1	1	2	2	2	1	2	2	2	2	2	2	2	2	2	2
5	68	2	2	3	1	2	2	2	3	2	3	1	2	3	3	3	2	3	2	2	2
6	70	2	2	3	3	3	2	3	3	3	2	3	2	3	2	3	2	3	3	2	3
7	72	3	3	1	4	2	3	3	1	3	3	3	3	1	3	3	3	2	3	3	3
8	74	3	3	4	3	3	3	3	4	4	4	3	3	4	4	4	3	4	4	3	4
9	76	4	3	4	4	4	4	4	4	3	4	4	3	4	4	4	4	4	4	4	4
10	78	4	4	5	4	4	4	4	5	4	5	4	4	5	5	5	1	4	4	4	4
11	80	5	4	5	5	4	4	5	5	5	5	5	4	5	4	5	3	5	1	3	5
12	82	5	5	4	5	5	5	5	6	5	6	5	4	2	5	6	4	5	5	5	5
13	84	6	4	6	5	5	3	6	6	6	6	6	6	6	6	5	5	6	5	5	5
14	86	6	6	3	6	5	6	6	5	5	4	6	5	6	6	6	6	6	6	5	6
15	88	3	6	6	6	6	6	7	7	6	7	5	5	7	7	7	4	7	6	6	6
16	90	4	5	7	6	6	5	6	7	7	7	7	6	7	7	7	6	7	7	6	7

Table F.4
Block days on which each patient is scheduled for surgery based on his/her H_{j1} when the increment of H_{j1} is 2 (all instances under half-day block system).

		Block day on which each patient is scheduled for surgery for each instance																			
j	H_{j1}	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	60	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	62	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1	1	2	1	1	1
3	64	2	2	2	2	1	2	1	2	1	2	2	2	2	2	2	2	1	2	1	1
4	66	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1	2	2	2	2
5	68	2	3	3	3	2	3	2	3	3	3	1	3	3	3	3	2	3	3	2	2
6	70	3	3	3	3	3	3	3	3	3	3	3	1	3	3	3	2	3	3	2	3
7	72	3	4	4	4	3	4	3	4	4	4	3	3	4	4	4	3	4	4	3	3
8	74	4	4	4	4	4	4	4	4	4	4	3	4	4	4	4	3	4	4	3	4
9	76	4	5	5	5	4	5	4	4	5	5	4	4	5	5	5	4	5	1	4	4
10	78	5	5	5	5	5	5	5	5	5	5	4	4	5	5	5	4	5	5	4	5
11	80	5	6	6	6	1	6	5	5	6	6	5	5	6	6	6	4	6	5	5	5
12	82	6	6	6	3	5	2	6	6	6	6	5	5	6	6	6	5	6	6	5	6
13	84	6	7	7	6	4	5	6	6	7	7	6	6	7	7	7	5	7	6	3	6
14	86	7	7	7	7	6	6	4	7	7	7	6	6	7	7	7	6	7	7	6	7
15	88	7	5	8	6	6	7	7	7	8	8	5	7	8	8	8	6	8	7	6	7
16	90	8	8	8	7	7	7	7	8	8	8	7	7	8	8	8	7	8	8	7	8

Table F.5
Block days on which each patient is scheduled for surgery based on his/her $\Delta\sigma_{jl}$ when the increment of $\Delta\sigma_{jl}$ is 0.2 (all instances under full-day block system).

		Block day on which each patient is scheduled for surgery for each instance																			
j	$\Delta\sigma_{jl}$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0.2	6	5	5	6	5	6	5	7	6	6	6	6	7	6	6	6	7	6	6	6
2	0.4	5	6	5	5	6	7	4	6	6	6	5	6	7	6	6	5	7	6	6	6
3	0.6	5	5	5	6	5	5	6	6	5	5	6	5	6	5	5	6	6	5	4	6
4	0.8	4	5	4	5	5	6	5	5	5	5	5	5	6	5	5	5	6	5	5	5
5	1	5	4	3	5	4	5	5	5	5	4	5	4	5	5	5	5	4	4	5	5
6	1.2	4	4	4	4	4	6	4	2	4	3	4	5	5	4	2	4	5	4	4	5
7	1.4	3	4	4	4	4	5	1	3	4	4	3	4	4	4	4	4	5	3	4	4
8	1.6	4	3	3	4	3	4	3	4	4	3	4	4	4	3	1	3	4	4	3	4
9	1.8	3	3	3	2	3	3	4	4	4	1	4	3	3	3	3	1	4	3	3	3
10	2	3	1	2	3	3	4	3	4	2	4	3	2	3	3	4	3	3	2	3	2
11	2.2	2	2	3	3	2	3	3	3	3	3	2	3	2	1	3	3	3	3	2	3
12	2.4	1	3	2	3	2	2	2	3	3	2	2	3	3	2	3	2	3	1	2	1
13	2.6	1	2	1	2	1	1	1	2	2	1	3	1	1	1	2	2	1	1	1	1
14	2.8	1	1	1	1	2	2	2	2	2	2	1	1	1	2	1	1	1	2	2	2
15	3	2	2	1	1	1	2	1	1	1	2	1	2	2	1	1	1	2	1	1	2
16	3.2	2	1	2	2	1	1	2	1	1	1	2	1	1	2	2	2	2	2	1	1

Table F.6

Block days on which each patient is scheduled for surgery based on his/her $\Delta\sigma_{jl}$ when the increment of $\Delta\sigma_{jl}$ is 0.2 (all instances under half-day block system).

		Block day on which each patient is scheduled for surgery for each instance																			
j	$\Delta\sigma_{jl}$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0.2	6	8	4	8	7	7	3	8	7	7	8	7	7	7	8	7	8	8	6	8
2	0.4	7	8	6	7	7	8	7	7	6	7	8	7	7	7	7	1	8	7	7	7
3	0.6	6	7	6	7	6	7	7	7	6	6	7	6	6	6	7	6	7	7	6	7
4	0.8	6	7	3	6	6	1	6	6	5	6	7	3	6	6	6	6	7	6	6	6
5	1	5	6	5	6	5	6	6	6	4	5	6	6	5	5	6	5	6	5	5	6
6	1.2	5	6	5	5	5	6	5	5	5	5	6	6	5	5	5	5	6	6	2	5
7	1.4	4	5	4	5	4	5	1	4	4	4	5	5	4	4	5	4	5	5	5	5
8	1.6	1	5	4	4	4	5	5	5	3	4	5	5	1	4	5	1	5	4	4	3
9	1.8	2	4	3	4	3	4	4	4	4	3	4	4	3	1	4	4	4	4	3	4
10	2	3	3	3	4	3	4	4	4	4	2	4	4	4	3	3	3	4	3	4	4
11	2.2	4	4	3	3	3	3	3	3	3	3	3	3	3	1	4	2	3	1	3	2
12	2.4	3	3	2	3	1	1	1	3	1	1	1	3	3	3	3	3	3	1	3	1
13	2.6	1	1	1	2	2	3	3	1	3	2	2	1	1	1	2	1	2	3	2	3
14	2.8	1	1	1	2	2	1	1	2	2	1	3	2	1	2	1	2	2	1	1	1
15	3	2	2	2	1	1	2	2	2	1	1	2	1	2	2	1	1	1	2	1	2
16	3.2	2	2	2	1	1	2	2	1	2	2	1	2	2	2	2	2	1	2	2	1

VITA

VITA

Joonyup Eun was born in Deagu, South Korea. He received his Bachelor of Engineering in Industrial Systems and Information Engineering from Korea University in 2007 and Master of Science in Industrial and Systems Engineering from Korea Advanced Institute of Science and Technology (KAIST) in 2009. Upon receiving his Doctor of Philosophy in Industrial Engineering from Purdue University in August 2016, he will join the Department of Anesthesiology, Vanderbilt University School of Medicine, as a postdoctoral research fellow.