2019

Resource constrained operating room scheduling

Musa Demirtas
Western New England University

Resource Constrained Operating Room Scheduling

by

Musa Demirtas

A dissertation submitted to the Faculty of Western
New England University
in partial fulfillment of the
requirements for the Degree of Doctor of Philosophy
in Engineering Management

Springfield, MA

August 15, 2019

Approved by

Dr. Christian Salmon, Interim Chair & Associate Professor, IEEM, Co-Chair	Date
Dr. Seyed Niknam, Assistant Professor, IEEM, Co-Chair	Date
Dr. Zhaojun Li, Assistant Professor, IEEM, Member	Date
Dr. Kourosh Rahnamai, Professor, ECE, External Member	Date
Dr. S. Hossein Cheraghi, Dean, College of Engineering	Date

Abstract

Due to an increasing number of patients, surgeries, competition, and limited government support, the healthcare industry needs to respond to these challenges as quickly as possible and use its limited resources efficiently. In hospitals, Operating Rooms (ORs) are considered among the most important and costly departments that generate a significant portion of revenues. Management of ORs is not easy due to the integration of many stakeholders with conflicting priorities, such as surgeons, patients, nurses, and management. At the same time, uncertainties, limited resources, and increased patient demand make the OR planning and scheduling one of the complex tasks in hospitals. Healthcare providers are facing pressure of how to decrease hospital costs and improve the waiting time of patients. For this, this study considers developing an efficient OR planning and scheduling system to minimize the costs and improve patients' waiting time. Three mathematical models are developed to schedule patients in ORs to have an efficient OR planning and scheduling system. The first model considers of only elective patient scheduling to minimize the cost of overtime, ORs, overtime in recovery units, and patients. The objective is minimizing the patients' total waiting time. The second model considers rescheduling and resequencing of elective patients and scheduling of emergency patients. Moreover, certain emergency patients may require an operation immediately or within a short time. Accordingly, the third model considers dedicated rooms in the reschedule phase. A standard optimization algorithm, LINGO 18, and a multi-heuristic algorithm, Genetic Algorithm, are used to find global optimum and decent feasible solutions, respectively, for a case study.

Acknowledgments

I would never have been able to finish my dissertation without the guidance and help of many people. It is my pleasure to offer my gratitude to them all.

Firstly, I would like to express my sincere gratitude to my previous advisor Dr. Keyser for the continuous support of my Ph.D. study and related research, for his patience, motivation, and immense knowledge.

Following my previous advisor, I want to specially recognize Dr. Niknam for being my advisor on the last year my Ph.D. study. His guidance helped me in this time of research and writing of this dissertation. I could not have imagined having a better advisor for my last year of Ph.D. study.

Besides my advisors, I would like to thank the rest of my dissertation committee: Dr. Salmon, Dr. Li, and Dr. Rahnamai, for their insightful comments and encouragement, but also for the hard questions which incented me to widen my research from various perspectives.

Last but not the least, I am deeply grateful to my wife, Dr. Esra Demirtas, and my son, Hasan Demirtas, for supporting me spiritually throughout writing this thesis and my life in general. They deserve special mention for their support and prayers.

Table of Contents

List of Tables

List of Figures

List of Acronyms

BIM	Break-In-Moment
CMP	Case Mix Plan
GDP	Gross Domestic Product
ILP	Integer Linear Program
LEPST	Longest Expected Processing with Setup Time
MILP	Mixed Integer Linear Programming
MSSP	Master Surgical Scheduling Plan
NHE	National Health Expenditures
OR	Operating Room
OT	Operating Theatre
PACU	Post Anesthesia Care Unit
RSM	Rescheduling Model
RSDM	Rescheduling Dedicated Model
SCAP	Surgical Case Assignment Problem
SICU	Scheduling Model
SM	Surgical Intensive Care Unit
SO	Surgical Operation
SSSP	Surgery Scheduling and Sequencing Plan

CHAPTER 1: INTRODUCTION

Healthcare is one of the most complex, largest and fastest-growing industries due to aging population and growing demand. The U.S. population is expected to increase by 18 percent between 2005 and 2025, and, at the same time, the population aged 65 and above is expected to increase by 73 percent [1]. Hence, considering higher demand, higher customer expectations, and limited resources, hospitals are in need of providing medical services in the most effective way to improve patient waiting times and to minimize hospital costs. National Health Expenditures (NHE) in the U.S. increased from around $1.3 trillion in 2000 to around $3.3 trillion in 2016, which accounts for 17.9 percent of the Gross Domestic Product (GDP) [2]. Figure 1.1 shows the NHE as a percentage of GDP between the years 2000 and 2016.

Hospital care expenditures were around $1.1 trillion in 2016, which is almost one third of the total health expenditures, as shown in Figure 1. 2 [2]. It is intriguing to note that only 11 percent of Gross Domestic Product (GDP) goes to healthcare expenses in western European countries, based on a report published by Statistics Explained in 2015 [3]. Furthermore, the cost of "overuse, underuse, misuse, duplication, system failures, unnecessary repetition, poor communication, and inefficiency" is more than half a trillion dollars per year for the healthcare system [4]. Moreover, it is crucial for hospitals to provide effective and efficient quality and safe service [5].

According to a study in 2012, nearly 200,000 Americans die annually from preventable medical errors [6]. The costs of these medical errors are between $17 billion and $29 billion [7]. Thus, the increasing demand and cost of healthcare has made the healthcare industry one of the largest and fastest-growing industries in the developed world and the largest domestic industry in the United States [8, 9, 10, 11,12]. Needless to say, healthcare providers are under pressure to use the limited resources in the most efficient way.

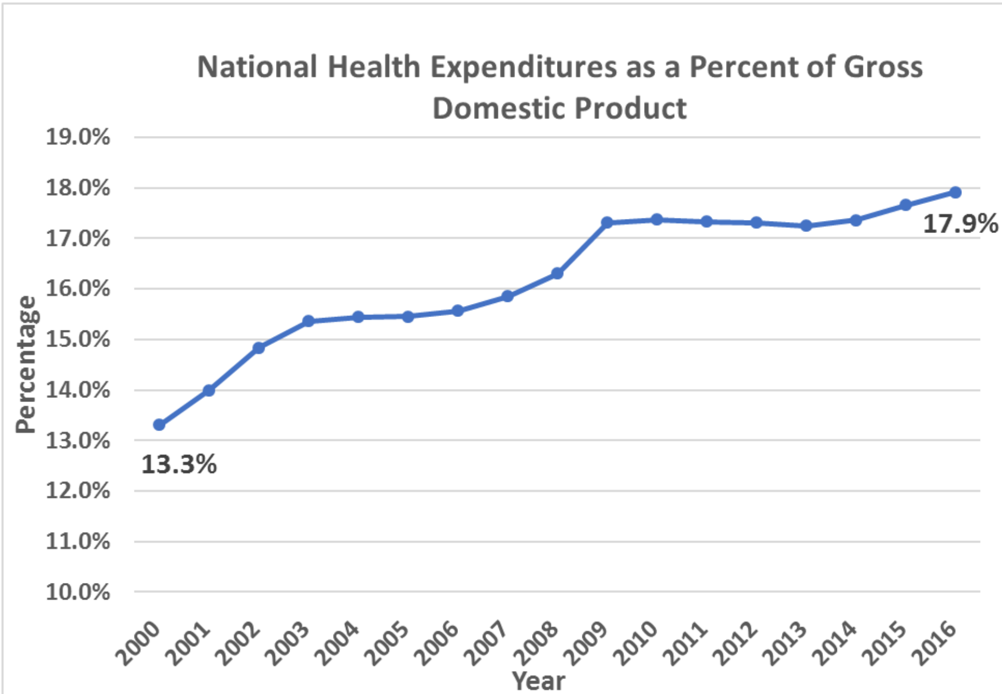

Figure 1.1 National health expenditures as a percentage of Gross Domestic Product [2]

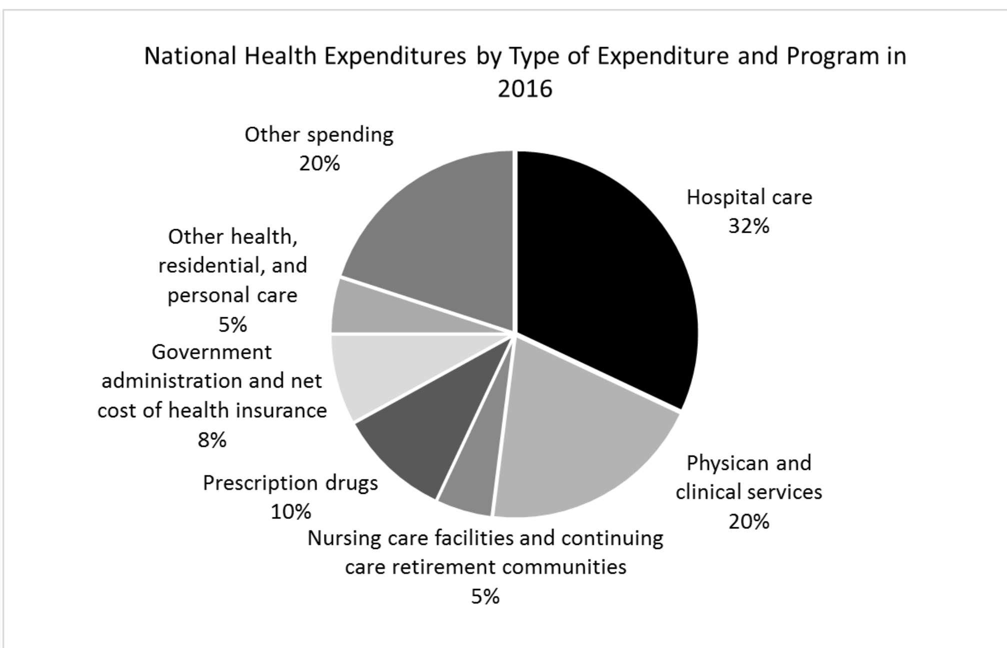

Figure 1. 2 National health expenditures by type of expenditure and program in 2016 [2]

Operating rooms (ORs) are considered the central engine of the hospitals and are responsible for generating more than 40 percent of a hospital's total revenue and consuming almost 30 percent of its resource costs [13, 14, 15, 16,17]. Moreover, ORs have a major effect on the performance of hospitals since they are connected with other hospital departments [18]. Even though ORs are one of the most important cost and revenue centers, they have an average utilization of only 68 percent [14, 19].

Improving the utilization of ORs may generate an additional $5 million annual revenue for hospitals [14]. However, it should be emphasized that OR planning and scheduling is a complex task due to many factors, including limited resources, uncertainty in the surgery durations, conflicting priorities of patients and surgeons, unexpected arrival of emergency patients, and allocating OR time for elective and emergency patients [9]. For this, an effective way of managing the OR planning and scheduling is optimal allocation of available resources to ORs.

In essence, using deterministic assumptions in surgical procedures, such as length of stay, patient flow, and duration of surgical operations (SOs), is not an effective solution in OR planning and scheduling. These surgical procedures have some degree of uncertainty and variability; therefore, there is a need to use stochastic methodologies in OR planning and scheduling [13].

To deal with unexpected arrival of emergency patients, hospitals reserve some OR capacity for urgent surgeries. This can be done in multiple ways. Dedicating certain ORs to emergency patients is the first option (these ORs are called "dedicated rooms"). In this option, an urgent surgery is undergone immediately if the dedicated room is empty. This option might result in low utilization of ORs. The second option is having urgent surgeries in elective ORs (i.e. no "dedicated rooms"). In this option, an emergency patient is taken to an available elective room to have an SO. Elective rooms are available for urgent surgeries before an elective surgery starts or after it finishes. These times or moments, i.e. before or after the operation of elective surgeries, are called

Break-In-Moments (BIMs), as shown in Figure 1. 3. If hospitals choose the second option, urgent surgeries will have to wait until elective surgeries are finished. The third option is the combination of the above-mentioned options in which an emergency patient is taken to a dedicated room to be operated on, otherwise the emergency patient is operated on in an elective room once it is available. Hence, BIMs are the points where urgent surgery can start an operation immediately. In order to minimize total waiting time for emergency patients, these BIMs need to be spread as evenly as possible [20, 21].

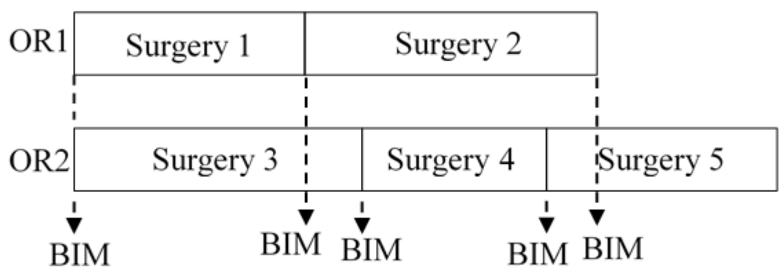

Figure 1. 3 The BIM problem

Improving the total waiting time of patients and decreasing hospital costs are two of the most challenging problems that healthcare providers need to manage, along with the limited resources. While improving the total waiting time and decreasing costs, hospitals need to minimize patients' length of stay, reduce waiting time in hospitals, and optimize the utilization of service rooms [22].

In addition, disruptions may occur in daily schedules due to unpredictable arrival of emergency patients and uncertain durations of surgical procedures. Thus, OR management needs to check and update the OR schedule frequently to determine a new schedule as and when needed [19,23].

1.1. Description of OR Planning and Scheduling Problem

The whole OR planning and scheduling process, also known as the peri-operative process, consists of three stages: pre-operative stage, intra-operative stage, and post-operative stage, as shown in Figure 1.4. Furthermore, Figure 1.4 shows the flow of surgical patients in hospitals. All of the surgery patients move through these three stages. The time period from decision of surgery for a patient to starting the surgical procedure is called the pre-operative stage in which all the preparations to start a surgical procedure, such as collection of patients' information, physical examination, and medical tests, are done. The required time in this stage ranges from minutes, for emergency patients who require an immediate surgical procedure, to days or months for elective patients with a surgery planned in advance. The next stage, intra-operative stage, begins with patients admitted to OR. In this stage, surgeries and all other activities during the surgical procedure are done in the OR. This stage ends with patients discharged from ORs. After the surgery ends, patients are transferred from ORs to recovery areas such as Post Anesthesia Care Units (PACU) or Surgical Intensive Care Units (SICU). The time period patients stay in PACU or SICU is called the post-operative stage, which is the last stage in the peri-operative process [24,25].

OR planning and scheduling problems consist of three hierarchical levels: strategic level (long-term planning), tactical level (medium-term planning), and operational offline/online level (short-term planning). Strategic level or long-term planning addresses long-term structural decision-making problems such as the number and location of ORs, determining regular working hours in ORs, various type of surgeries to be done, and various type of patients to be treated. Once these decisions are made, the OR management needs to provide a Case Mix Plan (CMP), which mainly needs to allocate the time blocks of ORs to specialties or to dedicated surgery groups. The decisions in the strategic level range from a few months to a few years.

The tactical level or medium-term planning deals with developing a Master Surgical Scheduling Plan (MSSP) which determines the number, type and hours of ORs assigned to surgical groups [24,26]. Time horizon for this level ranges from a few weeks to a few months.

The final level, i.e. operational offline/online level, is about developing a Surgery Scheduling and Sequencing Plan (SSSP). This short-term planning consists of two parts; offline and online. In the offline planning, individual patients are scheduled and sequenced in each OR on a daily basis, which is called SSSP. Developing an SSSP usually has two steps. In the first step, patients from a waiting list are assigned to ORs with a date. The second step determines the sequencing of these patients for a certain day in each OR. In the online planning, if any emergency patients are admitted to have a surgery, the current schedule is adjusted to include the emergency patients. This is called rescheduling. Time horizon for the short-term offline planning ranges from a few days to a few weeks, while time horizon for the online planning is a few hours to one day since emergency patients have stochastic arrivals.

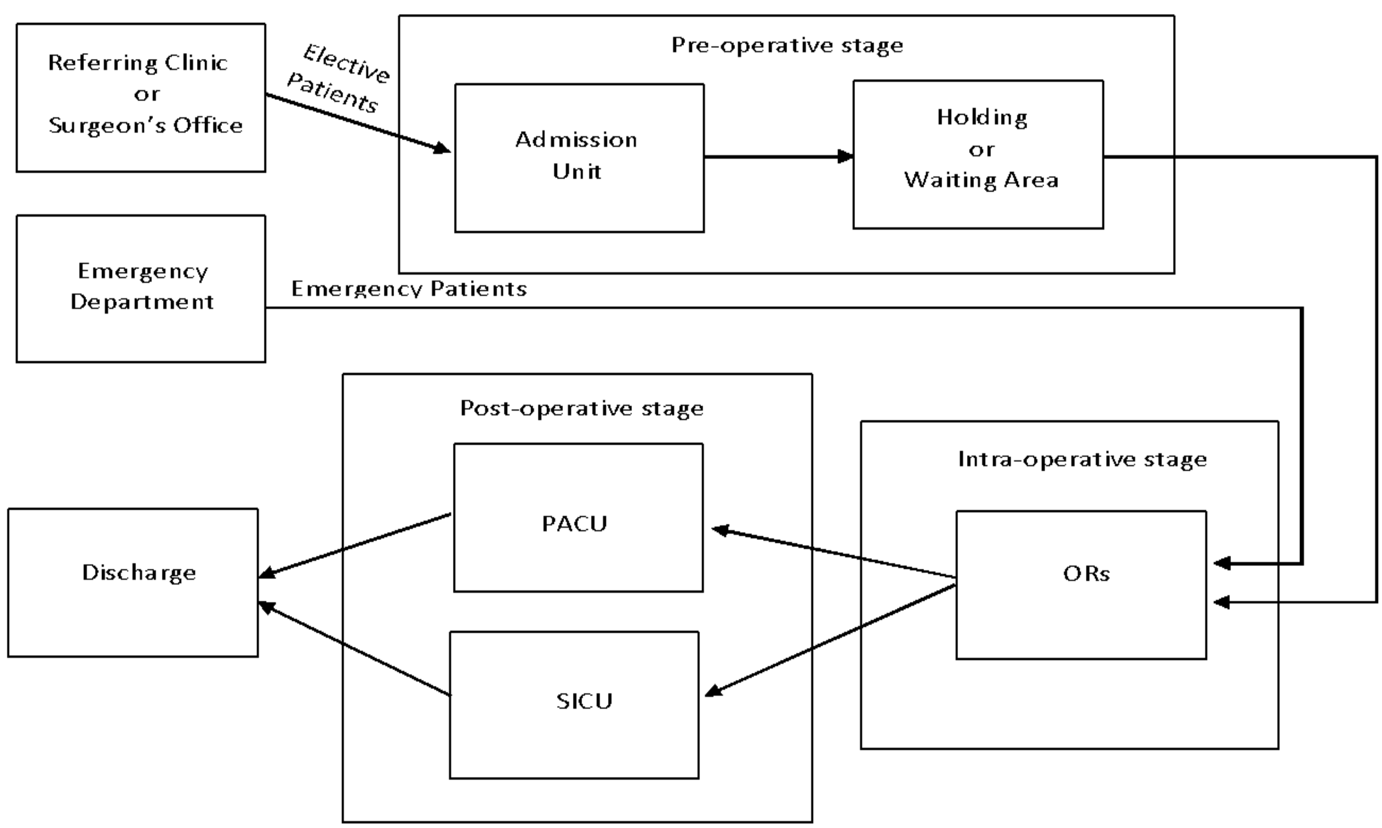

Figure 1.4 Flow of surgical patients in the hospitals

There are two types of patients whom may share the capacity in ORs; (i) elective patients and (ii) emergency patients. Elective patients are known by the hospital and scheduled several days prior to their operation. Elective patients can be either inpatient or outpatient. Patients in the former group are admitted to hospital at least one day before their surgical procedure. After the surgical procedure, they recover in the PACU or SICU. Then, they go back to their inpatient beds to stay until their discharge is authorized by an attending physician. Outpatients are admitted to hospital on the same day of their surgery. After the procedure, outpatients recover in the PACU or SICU until they are discharged. On the other hand, emergency patients have uncertain or stochastic arrivals to hospital with an urgent need for surgery. Any unnecessary care or surgery postponement for emergency patients may lead to serious medical consequences.

1.2. Significance

Even though OR planning and scheduling has a large body of knowledge [14, 18, 27, 28], there are still challenges which need to be addressed, especially in the area of scheduling and rescheduling of the elective patients due to disruptions in the current schedule [27, 29, 30]. There are only a few studies that consider OR scheduling and rescheduling problems and, of these, the majority do not consider OR scheduling and rescheduling problem in the same problem context along with waiting time for emergency patients. The problem of minimizing the total waiting time for emergency patients is investigated in only a few studies. Even so, these studies do not consider the rescheduling of elective patients due to disruptions in the current schedule.

Therefore, to fill the above-mentioned research gaps, the objective of this research is to minimize the total OR costs and to improve the total waiting time for elective and emergency patients in the operational offline/online scheduling stage. This will be accomplished by providing three mathematical models for scheduling and rescheduling of elective patients and scheduling emergency patients using elective and dedicated rooms.

The first mathematical model, which is called the Scheduling Model (SM), schedules and sequences elective patients in elective ORs. Elective patients are known in advance and come from a waiting list. When emergency patients arrive with an unexpected and uncertain arrival time, they disrupt the current elective schedule because emergency patients need to be operated on as soon as possible after they arrive. The other disruption source is changes in the surgical durations. If a surgery takes longer or shorter than expected, the starting time of the following surgeries may need to be changed. Based on these two disruptions, the SM needs to be updated to fix the schedule. The second mathematical model, which is called the Rescheduling Model (RSM), reschedules and resequences elective patients and schedules emergency patients using elective ORs. Emergency patients who require a SO immediately or in a certain time period may not be scheduled using RSM due to unavailable capacity in elective ORs. The third mathematical model, which is called the Rescheduling Dedicated Model (RSDM), reschedules and resequences elective patients and schedules emergency patients using dedicated rooms.

In this way, there are many different performance goals to be considered, such as optimizing utilization, minimizing cost, overtime, length of stay and total waiting times, and maximizing surgeons' preferences, that have been used in the OR planning and scheduling literature. Next, a brief description of the performance goals used in this study is presented.

1.2.1. Patients' Waiting Time

Patients' waiting time for a surgery is considered among the most common performance indicators of OR planning and scheduling. It is directly related to utilization and cost of ORs. In essence, OR planning and scheduling literature shows extensive application of this performance goal. Increasing waiting time of surgeries would negatively affect patient satisfaction and health. Long waiting times can result in patients developing chronic illnesses, which can cause an increased healthcare, cost [17]. The waiting time for surgery is an important factor for patients and, sometimes, this is the main reason in hospital selection. Long waiting times in an OR are one of the issues that healthcare organizations need to fix by optimal allocation of hospital capacity to patient demand [31]. Waiting time for a surgery is also one of the indicators showing how successful a health system is.

1.2.2. Inpatients' Length of Stay

Length of hospital stay for inpatients highly affects the hospital resource utilization and costs. Hospital administrators try to reduce the inpatients' length of stay in the ORs to increase the efficiency by delivering the treatment in a timely manner. In order to increase patient satisfaction and quality of wellbeing, length of stay needs to be minimized.

1.2.3. OR costs

ORs are one of the biggest cost centers in hospitals. Hospital administrations work to minimize this cost to avoid financial crisis. Hospitals need to have an efficient surgical scheduling and sequencing process to increase the availability of ORs and decrease OR costs [32]. There is an extensive body of work that deals with the minimization OR costs.

1.2.4. OR utilization

One of the widely studied objectives in OR planning and scheduling is utilization [10, 33,34, 35, and 36]. The increasing demand of surgeries requires ORs to be fully utilized in order to avoid surgery cancellations. Since ORs are the largest cost centers in hospitals, even a small amount of idle time in ORs will bring considerable costs to hospitals. However, high utilization of ORs may cause to have longer patient waiting times, while allocation of more time for surgeries may increase overtime in ORs [37, 38]. Increasing the number of ORs alone without considering downstream resources, such as PACUs and SICUs, will increase the patients' waiting times and the OR cost when there is unavailable capacity in PACU or SICU. Thus, hospital administrators maximize the utilization of the operating theatre (OT) as a whole.

Despite a large body of literature in OR planning and scheduling, there are still issues that need to be investigated. Especially, while dealing with unpredictable demand for emergency surgeries, hospital management needs to optimize the scheduling in ORs by improving the patients' waiting time and hospital costs. Reactive scheduling, minimizing the length of stay, and improving patient waiting times are some of the new criteria that researchers have started to use in their studies.

In this study, mathematical models are developed to allocate a hospital's limited resources to multiple ORs in order to produce schedules that:

- Minimize the patients' (elective and emergency) total waiting time;
- Minimize the inpatients' length of stay;
- Minimize OR costs;
- Optimize OR utilization.

1.3. Limitations

This section presents some of the limitations to this study. A major limitation is the cost terms of the objective functions of the mathematical models. This study assumes that all of the cost terms are independent and they do not affect each other. Another major limitation is the type of ORs. This study assumes that all of the ORs are identical and surgeries can be performed in any available ORs. However, in real life, ORs are classified into groups based on the surgeries that can be performed in them. Another drawback of the study is the availability of the machines and equipment and their failure and breakdowns. If certain machines and equipment fail or are not available to perform the surgeries, rescheduling of patients is needed.

1.4. Organization of the Dissertation

The organization of this proposal is as follows: Chapter 2 provides the literature review related to OR planning and scheduling problems. The methodology for allocation of resources to ORs under uncertainty is presented in Chapter 3. Chapter 4 presents the results of the study. Chapter 5 discusses the conclusions and future research directions for the OR planning and scheduling problems.

CHAPTER 2: LITERATURE REVIEW

In this chapter, a review of published healthcare literature related to the various performance goals of OR planning and scheduling is presented. The performance indicators of OR planning and scheduling used in this study include surgical length of stay, patients' total waiting time, OR costs, and OR utilization. This chapter reviews these performance indicators and addresses the research gaps in the field of OR planning and scheduling.

2.1.Length of Stay

Even though minimizing inpatients' length of stay decreases healthcare cost and increases inpatients' satisfaction, this goal has not received much attention by researchers. Zhang et al. [39] examined how to allocate OR capacity to minimize inpatients' length of stay by developing a mixed integer programming model. They stated that reducing inpatients' stay time decreases healthcare cost since inpatients use beds and other resources while they stay. Accordingly, their work is the first one to consider the objective of minimizing inpatients' length of stay in an OR capacity planning model. However, they did not consider rescheduling. If any disruptions occur which affect the OR planning and scheduling, rescheduling/optimization is not considered in [39].

Jeang and Chiang [40] minimized idle time and overtime in ORs by modeling the OR scheduling problem as a nonlinear integer programming. The authors considered how to reduce inpatients' length of stay and waiting time of a surgery in their model. Their model allows for rescheduling if minor changes happen. However, their model does not include interval or turnover time between SOs, even though they admit that the turnover time should be about one hour.

Another big limitation of the model is that operating time of SOs is a deterministic variable instead of being a random variable with a probabilistic distribution.

Testi and Tànfani [41] studied OR planning and scheduling by developing a 0-1 linear programming model for the Master Surgical Scheduling Problem (MSSP) and Surgical Case Assignment Problem (SCAP). Their model aims at scheduling patients to ORs according to patients' length of stay with an objective of minimizing overall patient welfare loss due to the negative consequences of excessive waiting. The authors used a block scheduling technique for elective inpatients and no emergency or outpatients were considered. Moreover, they did not include uncertainty in their model for the inpatients' length of stay and surgical durations.

2.2. Patients' Waiting Time

Many researchers have considered reducing patients' waiting time in ORs as the source of most complaints for hospitals stems from long waiting times. Jebali et al. [42] provided a two-step approach for daily OR scheduling which consists of assigning operations to ORs in the first step, then sequencing them as the second step. For this two-step approach, they used a mixed integer programming approach, which minimizes overtime and undertime in the ORs, and patients' waiting time. They believed this objective would increase bed availability and patient satisfaction. They did not consider the uncertainty for the durations of SOs.

Persson and Persson [43] studied the scheduling of elective surgeries in ORs by examining a simulation-optimization approach. In their model, they minimized the cost of using ORs for surgeries, staff working overtime, patients using extra beds, and patients waiting too long before the operation. Their model did not address the utilization of OR, inpatients' length of stay, or uncertainty in the surgery durations.

Denton et al. [19] investigated the optimization of OR overtime, undertime and patient waiting time in the surgery scheduling and sequencing process. They presented a two-stage integer stochastic recourse model using heuristics to sequence the operations in the OR. The authors considered the single isolated OR and uncertainty in surgery durations. Moreover, they did not consider reactive scheduling if any disruptions occur in the current model.

Tànfani and Testi [44] studied the OR planning and scheduling problems, in particular the MSSP, by formulating a binary linear programming model and then using heuristics to solve this model. In their model, the optimization of OR cost, overtime, patients' length of stay and waiting time, available OR equipment, number of surgeons, number of stay and SICU beds were considered. It is to be noted that the authors did not consider any uncertainty in the model and no emergency or outpatients were scheduled.

Denton et al. [35] developed a two-stage stochastic programming model using heuristics to optimize surgery schedules. They considered both open and block scheduling techniques to optimize patient and surgeon waiting time, OR idle time, and OR overtime costs. However, they only included elective patients and a single OR in their stochastic optimization model.

Ozkarahan [45] provided a goal programming model which considers OR utilization, surgeon preferences, SICU capabilities, and block restrictions on a particular day. The main objectives of the Ozkarahan model are to minimize idle time and overtime, and to maximize staff, surgeon, and patient satisfaction. However, this model did not consider the inpatients' length of stay and uncertainty in the duration of surgeries.

Wullink et al. [46] compared two approaches of reserving OR capacity for emergency patients: dedicated ORs only for emergency cases and reserving some of OR capacity for emergency cases. They assumed a block scheduling technique for elective surgery scheduling. The authors collected data from one of the biggest hospitals in Netherlands and used discrete event simulation to compare the patient waiting time, staff overtime, and OR utilization for the two approaches. Based on the comparison of these two approaches, evenly reserving some capacity in

ORs for emergency patients would provide better results with respect to patient waiting times, staff satisfaction, and cost-effectiveness. In this study, isolated ORs without considering downstream resources' such as PACU and SICU were considered.

Gul et al. [47] developed a stochastic integer programming model for the assignment of surgeries into ORs. They considered scheduling and rescheduling of surgeries due to cancellations. They included three objectives in their model: expected cost of surgery cancellations, postponement, and OR overtime. Using a progressive hedging algorithm, the model seeks to find a near optimal solution. The demand and duration of surgeries are considered to be uncertain. However, the authors did not consider length of stay and patients' waiting time. Furthermore, downstream resources, such as PACU and SICU, were not considered in the study.

Heng and Wright [48] provided the benefits of a dedicated OR for emergency surgeries by investigating a real situation at Canada's largest pediatric hospital, known as "The Hospital for Sick Children". They evaluated several performance indicators, namely, OR use, wait times, percentage of cases done within and outside of access targets, off-hours surgery, cancellations, overruns, and length of stay by comparing a 6-month pre-implementation period with a 6-month post-implementation period. Their results show that having a dedicated OR for emergency surgeries improves patients' waiting time by decreasing cancellations and overruns of elective surgeries. Downstream resources, such as PACU and SICU, were not considered in the study.

2.3. Minimize OR Costs

Min and Yih [49] formulated a two-stage stochastic mixed integer programming model for scheduling elective surgery patients with respect to minimization of patient cost and overtime costs under a surgical intensive care unit. The authors assumed that surgery durations, length of stay in SICU, and new demand is random with known distributions. They did not consider the uncertainty in the new arrival of patients and rescheduling if any disruptions occur in the schedule.

Lamiri et al. [50] developed a stochastic optimization model for OR elective surgery planning with uncertain surgical durations while considering uncertain demand for emergency surgery. In this model, elective cases are scheduled over a planning horizon. However, emergency surgeries are scheduled on the day of arrival. Minimization of expected OR overtime costs and elective patient-related costs are considered in their model. The authors did not consider the waiting time, the length of stay of patients, and downstream resource management in their study.

Lamiri et al. [9] studied the OT planning problem to determine the elective surgery planning for ORs under uncertain demand for emergency surgery. They considered a block scheduling technique in the study. They minimized patient-related costs and OR utilization costs, such as overutilization and underutilization costs, by developing a stochastic integer programming model. They utilized a column-generation approach to solve the associated mathematical model, but their work only considered elective patients and assumed elective surgery durations are deterministic.

Fei et al. [51] used a block scheduling strategy to build a weekly planning program for an OT. A binary integer programming model was developed to minimize OR costs due to the cost of operating time and overtime. They used a column-generation approach to solve the model. Two steps are needed for block scheduling strategy. Step one constructs a master surgery schedule which assigns time blocks to surgeons. Step two assigns surgery cases to a surgeon's time blocks. It is assumed that the master surgery schedule is already constructed. No emergency surgeries or uncertainties were considered in the study.

2.4. Optimize OR Utilization

Arnaout and Kulbashian [11] studied the OR scheduling problem with an objective of maximizing the utilization of ORs. By utilizing discrete event simulation, a heuristic model was developed and tested with two other existing heuristic models. They concluded that the heuristic model they developed, which is Longest Expected Processing with Setup Time (LEPST), is the most efficient heuristic model for scheduling of ORs in terms of maximizing the utilization of the ORs. In their work, the authors considered elective patients and isolated ORs. Moreover, they assumed the arrival of patients to be deterministic.

Fei et al. [52] considered an open scheduling strategy to study OT planning problems. They developed an integer programming model to maximize OR utilization and to minimize overtime cost. A column-generation-based heuristic technique was utilized to solve the reformulated associated mathematical model as a set-partitioning model. Using this heuristic algorithm, an efficient solution was obtained in this study. The authors did not include emergency patients in their study. Downstream resources, such as recovery beds, were assumed to be sufficient as well.

Dexter et al. [53] used an open scheduling technique to compare 10 scheduling algorithms for add-on elective OR cases to determine their performance at maximizing the utilization of ORs using computer simulation. In this study, two types of data were collected; hours of open OR time available for add-on cases, and duration of each add-on case. A bin packing algorithm compared 10 scheduling algorithms and they concluded that the best and worst algorithms for elective add-on case scheduling for maximizing the utilization of ORs are “best fit descending with fuzzy constraints” and “worst fit ascending”, respectively. In this study, the authors only considered elective patients.

Hans et al. [54] investigated the problem of assigning elective surgeries to ORs to optimize the OT and minimize overtime in ORs by using an open scheduling strategy. They used a method of “robust surgery loading”, which is based on assigning planned rest time and surgeries to ORs. In this work, constructive heuristics were provided and local search methods for the robust surgery loading problem utilized. Emergency surgeries were not considered in the study.

In study [55], the authors considered scheduling-rescheduling elective patients using a rolling horizon approach. Upstream and downstream units were considered and an MILP with the objectives of minimizing tardiness, idle time and overtime developed. However, surgery durations were assumed to be deterministic and no BIMs were considered.

2.5. Break In Moment (BIM)

Van der Lans et al. [20] developed various operational off-line heuristics to study the BIM problem. Then, they tested these heuristics through a simulation study and compared five different methods in the BIM problem: reserving dedicated rooms for emergency patients, reserving some OR capacity under a subset of ORs without BIM and with BIM, reserving some OR capacity under the whole ORs capacity without BIM and with BIM. The authors concluded that the best option is reserving some OR capacity under all ORs with BIM. They only considered operational off-line level in a BIM problem to minimize the waiting time for emergency patients. However, rescheduling of elective patients upon the arrival of emergency patients was not considered.

Van Essen et al. [21] provided some heuristic and exact solution methods for the BIM problem. They developed an integer programming model to maximize the number of BIMs for the OR scheduling problem. They assumed that elective patients are already assigned in the ORs. Furthermore, they did not consider the rescheduling of the elective patients due to disruptions in the schedule.

2.6. Rescheduling

Erdem [29] studied the OR scheduling and rescheduling problem by developing mathematical optimization models. The first optimization model considered scheduling elective patients to minimize the cost of ORs, hiring additional surgical teams, and some downstream resources, such as PACU beds. The second optimization model considered rescheduling elective patients upon the arrival of emergency patients. Surgical durations were assumed to be stochastic with known probability. However, the author did not consider emergency patients' waiting time in the research.

van Essen et al. [56] developed a decision support system for the OR rescheduling problem by modeling as an Integer Linear Program (ILP). Their ILP considered rescheduling of elective patients due to variability in the surgical durations and unexpected arrival of emergency patients. Their decision support system provides the three best adjusted schedules by minimizing the deviation of the preferences of the all involved stakeholders. However, the ILP considers only one OR at a time.

Heydari and Soudi [57] proposed a two-stage stochastic programming model with recourse to study the OR rescheduling problem. They developed an operational off-line model for the elective patient scheduling by considering the probability of the arrival of emergency patients. Their model minimizes the makespan, overtime and the expected cost of disruption. Surgical durations are assumed to be deterministic. They did not consider the operational on-line scheduling level, which reschedules the elective patients at the time of emergency patients' arrival and schedules the arriving emergency patients.

Stuart and Kozan [58] proposed an optimization model to sequence the elective and non-elective patients in a single OR. They assumed that elective patients are scheduled in advance. Their optimization model reschedules and resequences the elective and non-elective patients due to changes in the surgical durations and unexpected arrival of non-elective patients. They recommended that future work should focus on expanding this study considering multiple ORs.

2.7. Machine Scheduling

OR scheduling problems look like a machine scheduling problem in production systems, where patients are considered jobs and rooms are machines. Despite some common features, this section intends to highlight the major differences between OR scheduling and machine scheduling.

Arnaout [11] studied the OR scheduling problem by maximizing the utilization of the rooms. For this, the author considered how to schedule *n* patients into *m* ORs, where each patient has a certain stochastic surgical duration and there is a preparation phase before each surgery. In essence, patient sequence in the ORs affects the preparation phase. Then, the author translated this OR utilization problem to a sequence-dependent job shop scheduling problem, such as *n* jobs scheduled on *m* identical machines with minimizing the makespan. In the paper, only non-emergency patients were considered.

Zhong et al. [59] adopted a two-stage approach to solve the surgery scheduling problem. The authors considered the surgery scheduling problem as a parallel machine scheduling with multi-machine job, where patients waiting for a surgery were considered as jobs and surgical teams, nurses, and surgical equipment regarded as machines. In the first stage, they assigned surgeons with their surgeries to ORs with an objective of minimizing the makespan for each OR. In the second stage, the authors sequenced surgeons in each OR with an objective of minimizing the cost. Only elective patients with deterministic surgical durations were considered in this study.

Wang et al. [60] studied the OR scheduling problem by developing an MILP model as a resource-constrained machine scheduling problem with machine eligibility constraints. They took a two-step approach to develop the MILP model. In the first step, they assigned jobs to machines , while, in the second step, they sequenced jobs on those machines by considering resource constraints. In their mathematical model, only elective patients with deterministic surgical durations were considered.

Pham and Klinkert [61] translated the OR scheduling problem to a multi-mode blocking job shop (MMBJS) problem and developed a MILP model based on the assumption of *n* jobs to be processed on *m* machines. To perform a job, it is required to have a set of operations and a set of resources needed for each operation, where this set of resources is called modes. Their scheduling problem considered assigning a mode and determining start and finish time for each operation with an objective of minimizing the makespan. Again, only elective patients with deterministic surgical durations were considered in this study.

Fei et al. [62] studied an OR scheduling problem by assigning a set of surgical operations to ORs. They adopted a two-stage approach for developing a mathematical model for the scheduling problem: (i) developing an integer programming model, and (ii) reformulating as a set partitioning problem. The authors only included the elective operations and assumed that surgical teams are available all the time. Even though machine scheduling and OR scheduling have some common features, it is evident from the literature that these two problems are basically dissimilar, as stated below:

- OR scheduling problems include many sources of uncertainty based on surgical durations and unexpected arrival of emergency patients, while machine scheduling problems are usually predetermined processes.
- In an OR scheduling system, surgical teams are a combination of surgeons, nurses, and anesthesiologists while, in machine scheduling systems, machines are normally operated by one technician.
- Managing and satisfying patients' requests and priorities in OR scheduling problems is much more difficult than managing jobs in machine scheduling problems, since human health is much more important than jobs.
- The main goals of the OR scheduling problems are improving the patients' waiting time and minimizing the costs, while machine scheduling problems usually have one main goal of maximizing the profit.
- In an OR scheduling system, if there are no available beds in the recovery area, patients need to stay in the OR until beds become available and this will disrupt the OR scheduling,

while, in machine scheduling, if there are no available machines in the precedent stage, then the semi-finished products will be temporarily stored and this will not disrupt the machine scheduling.

- In an OR scheduling system, patients, surgical teams, and the management have many conflicting requests and priorities and it is much harder to satisfy all of these requests and priorities, while, in machine scheduling, the main priority is maximizing the profit.

For the above-mentioned reasons, this study does not consider the OR scheduling system as a machine scheduling system.

CHAPTER 3: MATHEMATICAL MODELS

This chapter presents the three scheduling and rescheduling optimization models that are explained in Chapter 1. The first model, SM, is dedicated to scheduling and sequencing elective patients. The second model, RSM, considers the rescheduling and resequencing of elective patients upon the arrival of emergency patients and changes in the surgical durations. The third model, RSDM, addresses the rescheduling and resequencing of elective patients and the scheduling of emergency patients using dedicated rooms.

3.1.A Mathematical Model for Scheduling and Sequencing Elective Patients

This section presents a partially stochastic mixed integer linear programming (MILP) model for the SM. A two-stage approach is used to develop this mathematical model. In the first stage, patients are scheduled to days and ORs based on their total waiting time and availability of the ORs. If a patient's waiting time is longer compared to other patients, then that patient will be scheduled earlier. In the second stage, patients are sequenced in the ORs to minimize the total waiting time of emergency patients and the completion time of the last surgery.

3.1.1. Modeling Elements

The mathematical models developed in this study intend to match the demand with the limited resources and take into consideration the system constraints to minimizing the cost. Figure 3.1 shows the demand, resources, goals, and constraints that are considered in the proposed models.

3.1.1.1. Demand

As explained in Chapter 1, patients are classified as either elective or emergency patients. Elective patients are scheduled for surgeries in advance. On the other hand, arrival of emergency patients is uncertain and hospitals either reserve some OR capacity or use elective rooms for those patients.

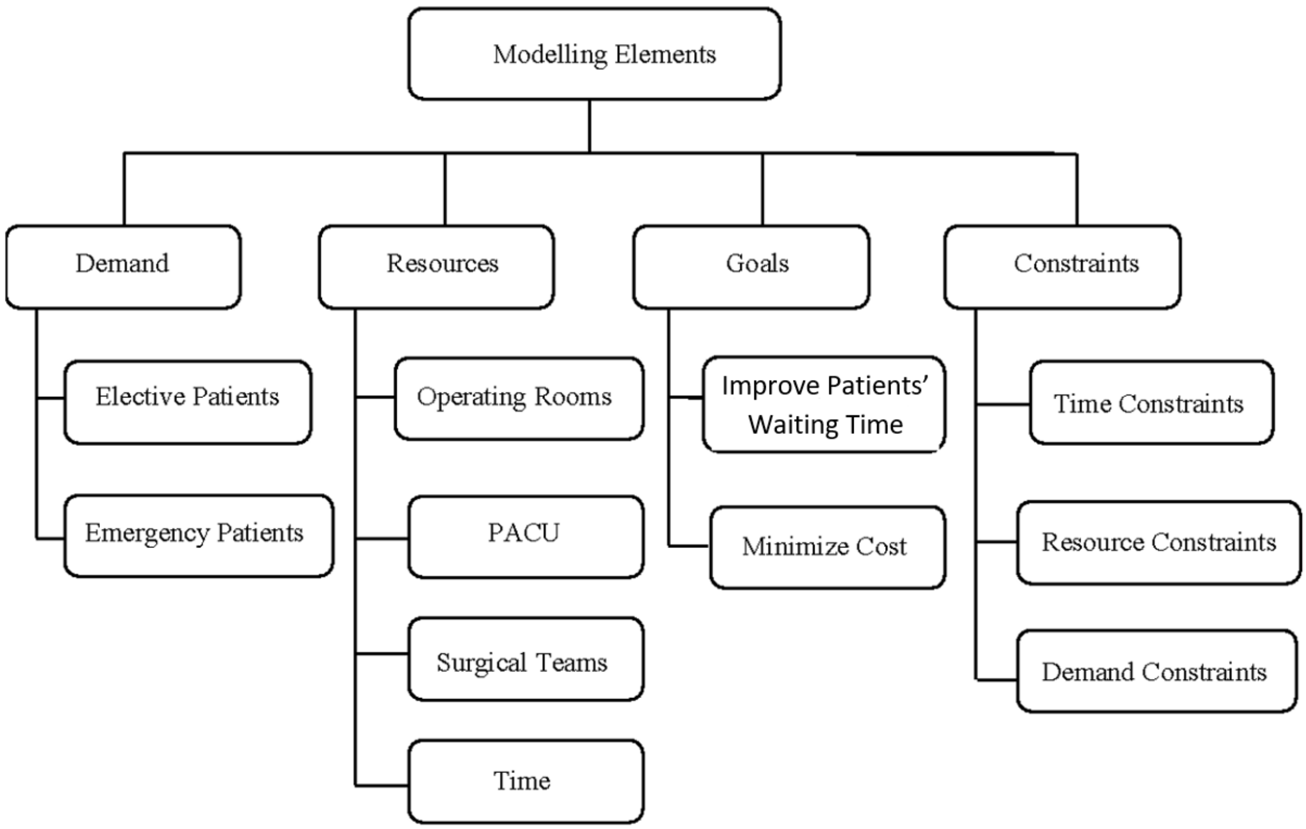

Figure 3.1 Modeling elements

Emergency patients are not considered to have surgeries in the SM, but they are taken into consideration in the elective scheduling as regard minimizing their total waiting time if they arrive at hospital. Surgical durations are assumed to be stochastic with known probability. Therefore, different scenarios are generated.

3.1.1.2. Resources

Four resources are considered in this study to match the patient demand: ORs, PACU or SICU, surgical teams, regular working time, and overtime. As mentioned earlier, the first developed model is intended to match the elective patient demand with ORs availability and minimizes cost of ORs. There are varieties of costs associated with ORs. This model considers the fixed cost of operating ORs, the cost of completing surgeries, and the cost of OR overtime.

After completion of the surgeries in ORs, patients are transferred to PACU or SICU for recovery. The bed capacity in the PACU or SICU is limited. Thus, when there is no available bed in PACU or SICU, their capacity might be expanded to satisfy patient demand. Otherwise, patients have to wait in ORs until there are available beds.

To perform the surgeries, hospitals need to have available surgical teams. Surgical teams include surgeons, surgical assistants, nurses, anesthesiologists, etc. This study takes into account the availability of the surgical teams to perform surgeries.

As mentioned earlier, the time spent in the ORs has two aspects; regular time and overtime. This model aims to schedule patients in the regular time to avoid overtime scheduling, since overtime means extra cost for hospitals.

3.1.1.3.Goals

The proposed mathematical models consider minimizing costs as the primary objective. While minimizing the costs, total waiting time of the patients' needs to be also improved to increase the efficiency in ORs. Therefore, the models in this study consider improving the total waiting time of elective and emergency patients. While minimizing the total waiting time for elective patients, this study also considers the total waiting time of the emergency patients. Assuming the emergency patients will be operated on in the elective rooms when they arrive at hospital, the mathematical models schedule the elective patients in a way that their completion times will be as evenly distributed as possible so as to minimize the total waiting time of emergency patients.

3.1.1.4. Constraints

The constraints applied in the optimization models are as follows:

- Demand Constraints:
  - Every patient has to be scheduled in the planning period or deferred to the next planning period.
  - The next patient will start to have surgery right after the previous patient plus turnover time in the same OR.
  - Patients with higher priority level have to be scheduled earlier than other patients.
  - Patients will be transferred to the PACU or SICU (if the PACU or SICU is available) right after they finish their surgeries.
- Resource Constraints:
  - There are a limited number of resources, ORs, PACU and SICU beds, surgical teams, and time, to perform the surgeries and recover them.
  - Resources cannot be assigned to more than one surgery at a time.
- Time Constraints:
  - There has to be enough time to perform the surgeries when they are scheduled.
  - There has to be a turnover (cleaning and preparation) time between surgeries.

3.1.1.5.Model description

It is assumed that a waiting list of elective patients with the type of surgery they request is available. Using this, in the first stage, patients are assigned to days and ORs. Then, in the second stage, patients are sequenced in the ORs. The availability of the ORs, surgical teams, and PACU and SICU beds are considered in the SM. Overtime constraints for the ORs and PACU and SICU beds are included as well. There is an upper limit for the PACU and SICU beds due to limited number of resources. Surgical durations are assumed to be a stochastic variable with known probability. Hence, different scheduling scenarios are generated for a number of surgical durations. Therefore, each scenario can be considered as a deterministic problem. The OR time needed to schedule for the patients and the number of beds in the PACU or SICU to satisfy the incoming patients are directly affected by those stochastic surgical durations. The SM intends to minimize the overtime in the ORs for each scenario.

Patient priority is another important factor that the SM considers. Some of the patients on the waiting list might need to be scheduled to have a surgery earlier than other patients because of their health conditions. The SM considers these patients as urgent and places them earlier than others. In the model, patient priority levels are characterized by numbers such as 3, 2, and 1. Higher numbers imply higher urgency. If the patients have equal priority level, then they are scheduled by their waiting time, which is another significant factor. Long waiting times will increase hospital costs since patients will be using the hospital resources, such as the beds and doctors, while awaiting surgery. Thus, minimization of elective patients' total waiting time is considered in the SM as an important objective. Waiting time and hospitalization cost of patients per day are assumed to be known and these are used as inputs in the model. Even though minimizing the patients' total waiting time will improve the patient satisfaction levels, it needs to be used along with the patient priority level since a patient with a more critical medical condition needs to be scheduled earlier. In other words, the priority level is favored over the waiting time.

After elective patients finish their surgeries in ORs, they are transferred to PACU or SICU beds for recovery. The number of beds in the PACU and SICU is limited and if there is no available bed, then patients wait in the ORs. Considering downstream resources such as PACU and SICU is highly important as they directly affect the scheduling of ORs. Thus, the SM considers the available capacity in the PACU and SICU in terms of beds and expansion of the capacity with an upper limit, and then tries to minimize the cost of expansion or the overtime in the PACU and SICU. shows the notation for index and parameters used in the SM.

Table 3. 1 Notation for index and parameters of the SM

Indices	
i, i^2	: Elective patient indices; $i, i^2 \in \{1, \dots, I\}$.
j	: SO type index; $j \in \{1, \dots, J\}$.
t, t^2	: Time period indices; $t, t^2 \in \{1, \dots, T\}$.
h, k	: Auxiliary time period indices; $h, k \in T + 3$.
d, d^2	: Day indices; $d \in \{1, \dots, D\}$.
m, m^2	: OR indices; $m, m^2 \in \{1, \dots, N\}$.
w	: Scenario index; $w \in \{1, \dots, W\}$.
Parameters	
FC	: Fixed cost of opening an OR during planning cycle;
MAX_i	: Maximum operation hours for patient i;
COR	: Overtime utilization cost of an OR during planning cycle (cost/hour);
$CPACU$	: Unit expansion cost of PACU during planning cycle (cost/bed);
$BPACU$	: Current capacity of the PACU in terms of beds;
$UPACU$	: Upper limit on the over-utilization of the PACU capacity in terms of beds;
CD	: Cost of deferring a patient to next planning cycle;
CC	: Cost of total completion time for all surgeries in each OR;
CR	: Penalty cost of repeating the completion times for surgeries;
OP_{jw}	: Operation time (hours) for surgery j under scenario w;
PC_j	: Length of stay (hours) at PACU for surgery type j;
RT	: Total number of regular working hours for ORs;
s_{ij}	: Equal to 1 if patient i requests surgery type j, 0 otherwise;
OP_{iw}	: Operation time (hours) time for patient i under scenario w;
PL_i	: Priority level of patient i;
TO	: Turnover time (hours);

P_w	: Probability of scenario w;
WT_i	: Waiting time (days) for patient i;
HS_i	: Hospitalization cost of patient i (cost/day);
M	: A sufficient large number;

The following calculation is used for converting operation hours of surgeries to operation hours of patients.

$$OP_{i,w} = \sum_{j \in J} (s_{ij} * OP_{j,w}), \quad \forall i \in I, w \in W$$

Table 3. 2 shows the notation for decision variables used in the SM.

Table 3. 2 Notation for decision variables of the SM

Decision Variables	
DF_i	: Equal to 1 if patient i is deferred to next planning cycle, 0 otherwise;
C_i	: Surgery completion time for patient i;
$CMAX_{dm}$	: The last surgery completion time on day d in OR m;
WC_{id}	: Waiting cost of patient i on day d;
F_{md}	: Equal to 1 if OR m is open on day d, 0 otherwise;
OT_{mdw}	: Amount of overtime utilization of OR m on day d under scenario w;
$OPACU$	: Amount of additional capacity (beds) placed in PACU;
Y_{idtmw}	: Equal to 1 if patient i has a surgery on day d at time t in OR m under scenario w, 0 otherwise;
X_{idtm}	: Equal to 1 if surgery starts on day d at time t in OR m for patient i, 0 otherwise;
G_{idtw}	: Equal to 1 if a patient i occupies a bed in PACU on day d at time t under scenario w, 0 otherwise;
$z_{ik}, RP_{ii'}$	: Auxiliary decision variables to calculate the BIMs.
$RP_{ii'}$	: Completion time repeats for patients i and i' .

The SM with seven different objectives $(OBJ1 + OBJ2 + OBJ3 + OBJ4 + OBJ5 + OBJ6 + OBJ7)$ is developed as follows:

$$Minimize\ OBJ1 + OBJ2 + OBJ3 + OBJ4 + OBJ5 + OBJ6 + OBJ7$$

$$OBJ1 = \sum_{i \in I} \sum_{d \in D} \sum_{t \in T} \sum_{m \in N} (X_{idtm} * WC_{id})$$

$$OBJ2 = \sum_{i \in I} (CD * DF_i)$$

$$OBJ3 = \sum_{i \in I} \sum_{i' \in I} (CR * RP_{ii'})$$

$$OBJ4 = \sum_{d \in D} \sum_{m \in N} (CC * CMAX_{dm})$$

$$OBJ5 = \sum_{m \in N} \sum_{d \in D} (FC * F_{md})$$

$$OBJ6 = \sum_{m \in N} \sum_{d \in D} \sum_{w \in W} (P_w * COR * OT_{mdw})$$

$$OBJ7 = \sum_{w \in W} (P_w * CPACU * OPACU)$$

Subject to

$$\sum_{d \in D} \sum_{t \in T} \sum_{m \in N} X_{idtm} + DF_i = 1, \qquad \forall i \in I \qquad (1)$$

$$\sum_{i \in I} X_{idtm} \leq 1, \qquad \forall d \in D, t \in T, m \in N \qquad (2)$$

$$\sum_{d \in D} \sum_{t \in T} \sum_{m \in N} \Big(t + MAX_i + \big((d-1) * T\big)\Big) * X_{idtm} = C_i, \qquad \forall i \in I \qquad (3)$$

$$\sum_{t \in T} (t + MAX_i) * X_{idtm} = ORC_{idm}, \qquad \forall i \in I, d \in D, m \in N \qquad (4)$$

$$ORC_{idm} \leq CMAX_{dm}, \qquad \forall i \in I, d \in D, m \in N \qquad (5)$$

$$\sum_{i\in I}\sum_{h} X_{idhm} \leq 1, \forall d \in D, t \in T, m \in N, w \in W, h = \max\big(1, t - OP_{i,w} + 1\big) - TO, \dots, t \quad (6)$$

$$\sum_{i\in I} Y_{idtmw} \leq N, \qquad \forall m \in N, d \in D, t \in T, w \in W \quad (7)$$

$$WC_{id} = HS_i * WT_i * d, \qquad \forall i \in I, d \in D \quad (8)$$

$$Y_{idkmw} \geq s_{ij} * \sum_{m\in N} X_{idtm}, \forall m \in N, i \in I, j \in J, d \in D, w \in W, k = \mathrm{t}, \dots, t + OP_{jw} - 1, \quad (9)$$

$$\sum_{i\in I}\sum_{m\in N} s_{ij} * Y_{idtmw} \leq ST_{jdt}, \qquad \forall j \in J, d \in D, t \in T, w \in W \quad (10)$$

$$\sum_{i\in I}\sum_{t} Y_{idtmw} \leq RT, \qquad \forall m \in N, d \in D, t \in T, w \in W, t \in \{1, \dots, RT\} \quad (11)$$

$$\sum_{i\in I}\sum_{t} Y_{idtmw} = OT_{mdw}, \qquad \forall m \in N, d \in D, t \in T, w \in W, t \in \{RT + 1, \dots, RT + OT\} \quad (12)$$

$$\sum_{i\in I}\sum_{t} Y_{idtmw} = 0, \qquad \forall m \in N, d \in D, w \in W, t \geq RT + OT + 1 \quad (13)$$

$$\sum_{k\in K}\sum_{w\in W} Y_{idkmw} \leq \sum_{j\in J}\sum_{w\in W} \big(s_{ij} * OP_{jw}\big) * \sum_{t\in T} X_{idtm}, \quad \forall d \in D, m \in N, i \in I \quad (14)$$

$$G_{idkw} \geq s_{ij} * \sum_{m\in N} X_{idtm}, \forall i \in I, j \in J, d \in D, t \in T, w \in W,$$

$$k = \mathrm{t} + OP_{jw}, \dots, t + OP_{jw} + PC_j - 1 \quad (15)$$

$$\sum_{i\in I} G_{idtw} \leq BPACU + OPACU, \qquad \forall\, d \in D, t \in T, w \in W \quad (16)$$

$$OPACU \leq UPACU, \quad (17)$$

$$\sum_{k\in I}\sum_{w\in W} G_{idkw} \leq W * PC_j, \qquad \forall\, i \in I, j \in J, d \in D, s_{ij} = 0, 1 \quad (18)$$

$$X_{idtm} \leq F_{md}, \qquad \forall i \in I, m \in N, t \in T, d \in D \quad (19)$$

$$X_{i'd't'm'} * PL_{i'} - X_{idtm} * PL_i \leq M * (1 - X_{idtm}),$$

$$\forall i, i' \in I, m, m' \in N, t, t' \in T, d, d' \in D \quad (20)$$

$$\sum_{k \in K} k * z_{ik} \leq C_i, \qquad \forall i \in I \qquad (21)$$

$$\sum_{k \in K} z_{ik} \leq 1, \qquad \forall i \in I \qquad (22)$$

$$RP_{ii'} \geq z_{ik} + z_{i'k} - 1\,, \qquad \forall i, i' \in I, k \in K \qquad (23)$$

$$RP_{ii'} \geq 1 - \sum_{k \in K} z_{ik} - \sum_{k \in K} z_{i'k}\,, \qquad \forall i, i' \in I \qquad (24)$$

$$D_i, C_i, CMAX_{dm}, WC_{idtm}, F_{md}, X_{idtm}, Y_{idtmw}, OT_{mdw}, RP_{ii'} \geq 0\,, \forall i, i' \in I, m \in N,$$

$$t \in T, d \in D, w \in W \quad (25)$$

$$F_{md}, X_{idtm}, Y_{idtmw}, D_i, z_{ik} \quad binary\,, \quad \forall i \in I, m \in N, t \in T, d \in D, w \in W, k \in K \qquad (26)$$

The objective function of the SM has seven different terms. Explanation of these objectives and constraints are showed in Table 3. 3.

Table 3. 3 Objectives and constraints of the SM

Objectives
OBJ1: This objective function minimizes the waiting cost of elective patients. OBJ2: This objective function minimizes the cost of deferring elective patients to next planning cycle. OBJ3: This objective function minimizes the penalty cost of surgery completion time repeats. Thus, BIMs will be maximized. OBJ4: This objective function minimizes the cost of completion the last surgeries in ORs. OBJ5: This objective function minimizes the cost of opening ORs. OBJ6: This objective function minimizes the cost of overtime in ORs.

OBJ7: This objective function minimizes the cost of overtime in PACU.
Constraints
Constraint (1): This constraint guarantees that every patient will be either scheduled to have a surgery or deferred to next planning period. Constraint (2): This constraint ensures that we cannot have more than 1 patient to start a surgery in any OR at the same time. Constraint (3): This constraint calculates the completion time of surgeries. Constraint (4): This constraint calculates the surgery completion time of patients for each day in each OR. Constraint (5): This constraint shows the last surgery completion time for each day in each OR. Constraint (6): This constraint guarantees that once a patient starts a surgery, we have to wait till that surgery plus turnover time end to start another surgery. Constraint (7): This constraint guarantees that the number of ongoing operations cannot be more than the number of ORs. Constraint (8): This constraint calculates the waiting cost of patients. Constraint (9): This constraint provides the link between the start and continuation of the SOs. Constraint (10): This constraint guarantees that the existing number of surgical teams will be equal or more than the ongoing operations. Constraint (11): This constraint determines the total utilization of the ORs in the planning cycle. Constraint (12): This constraint calculates the amount of overtime utilized in ORs. Constraint (13): This constraint ensures that we cannot have any ongoing operations outside of the planning period. Constraint (14): This constraint ensures that decision variable Y_{idtmw} will be zero if a patient

finishes his/her surgery.

Constraint (15): This constraint shows that patients will transfer and stay for a certain period of time in the PACU.

Constraint (16): This constraint shows that the current plus additional (if needed) capacity in PACU will be enough to satisfy the transferring patients from the ORs.

Constraint (17): This constraint determines the upper limit on the PACU capacity.

Constraint (18): This constraint makes sure that the decision variable of the PACU, G_{idtw}, will be zero if the PACU is empty or there is no patient in it.

Constraint (19): This constraint guarantees that an OR will be closed if there are no ongoing operations in that OR.

Constraint (20): This constraint is priority constraint.

Constraints (21) - (24): These constraints are the BIM constraints that calculate the BIMs.

Constraint (25): This constraint is the non-negativity constraint on all the decision variables.

Constraint (26): This constraint defines each $F_{md}, X_{idtm}, Y_{idtmw}, D_i,\ z_{ik}$ decision variable to be a binary variable.

3.2. A Mathematical Model for Rescheduling and Resequencing Elective Patients

In this section, a partially stochastic MILP model is developed for the RSM. Surgery durations and emergency patient arrivals are the two main sources of uncertainty that may disrupt the existing patient schedule. When disruptions happen in the current schedule, it is modified by postponing, preponing, or canceling previously scheduled patients in order to tackle the disruptions. Even though different scenarios are generated for the surgical durations, they may take shorter or longer than expected. If surgical durations take shorter than expected, the ORs will be empty until the next planned surgery and utilization of the ORs will be decreased. This will increase the cost in the ORs. If surgical durations take longer than expected, this will cause all of the next planned surgeries to start later than the scheduled start time and would increase patients' waiting time. Having surgical durations longer than expected may also cause overtime in the ORs and canceled patients. In both scenarios, having durations shorter or longer than expected, requires rescheduling and resequencing patients in the ORs.

The other source of disruption is emergency patient arrivals. When emergency patients arrive, they need to be operated on as quickly as possible. As discussed earlier, in this study, emergency patients are operated on in elective ORs. The SM maximizes the number of BIMs in order to minimize the total waiting time of emergency patients in the elective patient schedule. The RSM answers the questions of what if surgical durations take shorter or longer and what if emergency patients arrive with urgent need for ORs. When emergency patients arrive, the model allocates them in their closest BIMs to minimize their total waiting time. If more than one emergency patient arrives at the same time, the RSM schedules them based on their urgency levels. Urgency levels are characterized by waiting time of emergency patients. If there is any ongoing or completed surgery at the time of emergency patient arrivals, they are not rescheduled by the RSM because either their surgeries are completed and they are transferred to PACU or SICU or ongoing surgeries cannot be cancelled or postponed. Surgical durations of emergency patients are assumed to be stochastic with known probability and they can have surgeries in any available OR. Table 3. 4 shows the notation for index and parameters used in the RSM.

Table 3. 4 Notation for index and parameters of the RSM

Indices	
i, i^2	: Elective and emergency patient indices; $i, i^2 \in \{1, \dots, I\}$.
j	: SO type index; $j \in \{1, \dots, J\}$.
t, t^2	: Time period indices; $t, t^2 \in \{1, \dots, T\}$.
h, k	: Auxiliary time period indices; $h, k \in T + 3$.
d, d^2	: Day indices; $d \in \{1, \dots, D\}$.

m, m^2	: OR indices; $m, m^2 \in \{1, \dots, N\}$.
w	: Scenario index; $w \in \{1, \dots, W\}$.

Parameters

FC	: Fixed cost of opening an OR during planning cycle;
t_s	: Reference starting time (i.e., the time when the emergency patients arrive);
BIM_{mdt}	: Equal to 1 if there is a BIM in OR m on day d at time t;
FOS_{idtm}	: Equal to 1 if there is an ongoing or finished surgery for patient i on day d at time t in OR m when the emergency patients arrive;
MAX_i	: Maximum operation hours for patient i;
COR	: Overtime utilization cost of an OR during planning cycle (cost/hour);
$CPACU$	: Unit expansion cost of PACU during planning cycle (cost/bed);
$BPACU$	: Current capacity of the PACU in terms of beds;
$UPACU$	: Upper limit on the over-utilization of the PACU capacity in terms of beds;
CD	: Cost of deferring a patient to next planning cycle;
CC	: Cost of total completion time for all surgeries in each OR;
CR	: Penalty cost of repeating the completion times for surgeries;
OP_{jw}	: Operation time (hours) for surgery j under scenario w;
PC_j	: Length of stay (hours) at PACU for surgery type j;
RT	: Total number of regular working hours for ORs;
s_{ij}	: Equal to 1 if patient i requests surgery type j, 0 otherwise;
OP_{iw}	: Operation time (hours) time for patient i under scenario w;
PL_i	: Priority level of patient i;
TO	: Turnover time (hours);
P_w	: Probability of scenario w;
WT_i	: Waiting time (days) for elective patient i;
ECh_i	: Waiting cost for emergency patient i;
HS_i	: Hospitalization cost of elective patient i (cost/day);
M	: A sufficient large number;

The following calculation is used for converting operation hours of surgeries to operation hours of patients.

$$OP_{i,w} = \sum_{j \in J} (s_{ij} * OP_{j,w}), \quad \forall i \in I, w \in W$$

Table 3. 5 shows the notation for decision variables used in the RSM.

Table 3. 5 Notation for decision variables of the RSM

Decision Variables	
D_i	: Equal to 1 if patient i is deferred to next planning cycle, 0 otherwise;
C_i	: Surgery completion time for patient i;
$CMAX_{dm}$	: The last surgery completion time on day d in OR m;
WC_{id}	: Waiting cost of patient i on day d;
F_{md}	: Equal to 1 if OR m is open on day d, 0 otherwise;
OT_{mdw}	: Amount of overtime utilization of OR m on day d under scenario w;
$OPACU$	: Amount of additional capacity (beds) placed in PACU;
Y_{idtmw}	: Equal to 1 if patient (elective and emergency) i has a surgery on day d at time t in OR m under scenario w, 0 otherwise;
X_{idtm}	: Equal to 1 if surgery starts on day d at time t in OR m for patient (elective and emergency) i, 0 otherwise;
G_{idtw}	: Equal to 1 if a patient i occupies a bed in PACU on day d at time t under scenario w, 0 otherwise;
$z_{ik}, RP_{ii'}$	: Auxiliary decision variables to calculate the BIMs.

The RSM with eight different objectives $(OBJ1 + OBJ2 + OBJ3 + OBJ4 + OBJ5 + OBJ6 + OBJ7 + OBJ8)$ is developed as follows:

$$Minimize\ OBJ1 + OBJ2 + OBJ3 + OBJ4 + OBJ5 + OBJ6 + OBJ7 + OBJ8$$

$$OBJ1 = \sum_{i \in I} \sum_{d \in D} \sum_{t \in T} \sum_{m \in N} (X_{idtm} * WC_{id})$$

$$OBJ2 = \sum_{i \in I} (CD * D_i)$$

$$OBJ3 = \sum_{i \in I} \sum_{i' \in I} (CR * RP_{ii'})$$

$$OBJ4 = \sum_{d \in D} \sum_{m \in N} CC * CMAX_{dm}$$

$$OBJ5 = \sum_{m \in N} \sum_{d \in D} (FC * F_{md})$$

$$OBJ6 = \sum_{m \in N} \sum_{d \in D} \sum_{w \in W} (P_w * COR * OT_{mdw})$$

$$OBJ7 = \sum_{w \in W} (P_w * CPACU * OPACU)$$

$$OBJ8 = \sum_{i \geq 9} \sum_{d \in BIM_{mdt}=1} \sum_{t \geq t_s, t \in BIM_{mdt}=1} \sum_{m \in BIM_{mdt}=1} (ECh_i * (t - t_s) * X_{idtm})$$

Subject to

$$\sum_{d \in D} \sum_{t \in T} \sum_{m \in N} X_{idtm} + D_i = 1, \qquad \forall i \in I \tag{1}$$

$$\sum_{d} \sum_{t} \sum_{m} X_{idtm} = 1, \qquad \forall i \in I, i \geq 9, t \geq t_s, d, t, m \in BIM_{mdt} \tag{2}$$

$$X_{idtm} = 1, \qquad i, d, t, m \in FOS_{idtm} \tag{3}$$

$$\sum_{i \in I} X_{idtm} \leq 1, \qquad \forall d \in D, t \in T, m \in N \tag{4}$$

$$\sum_{d \in D} \sum_{t \in T} \sum_{m \in N} \Big(t + MAX_i + \big((d-1) * T\big)\Big) * X_{idtm} = C_i, \qquad \forall i \in I \tag{5}$$

$$\sum_{t \in T} (t + MAX_i) * X_{idtm} = ORC_{idm}, \qquad \forall i \in I, d \in D, m \in N \tag{6}$$

$$ORC_{idm} \leq CMAX_{dm}, \qquad \forall i \in I, d \in D, m \in N \tag{7}$$

$$\sum_{i\in I}\sum_{h} X_{idhm} \leq 1 \text{ ,} \forall d \in D, t \in T, m \in N, w \in W, h = \max\left(1, t - OP_{i,w} + 1\right) - TO, \dots, t \quad (8)$$

$$\sum_{i\in I} Y_{idtmw} \leq N, \qquad \forall m \in N, d \in D, t \in T, w \in W \quad (9)$$

$$WC_{id} = HS_i * WT_i * d, \qquad \forall i \in I, d \in D \quad (10)$$

$$Y_{idkmw} \geq s_{ij} * \sum_{m\in N} X_{idtm} \text{ ,} \forall m \in N, i \in I, j \in J, d \in D, w \in W, k = \mathrm{t}, \dots, t + OP_{jw} - 1, \quad (11)$$

$$\sum_{i\in I}\sum_{m\in N} s_{ij} * Y_{idtmw} \leq ST_{jdt} \text{ ,} \qquad \forall j \in J, d \in D, t \in T, w \in W \quad (12)$$

$$\sum_{i\in I}\sum_{t} Y_{idtmw} \leq RT \text{ ,} \qquad \forall m \in N, d \in D, t \in T, w \in W, t \in \{1, \dots, RT\} \quad (13)$$

$$\sum_{i\in I}\sum_{t} Y_{idtmw} = OT_{mdw} \text{ ,} \qquad \forall m \in N, d \in D, t \in T, w \in W, t \in \{RT + 1, \dots, RT + OT\} \quad (14)$$

$$\sum_{i\in I}\sum_{t} Y_{idtmw} = 0, \qquad \forall m \in N, d \in D, w \in W, t \geq RT + OT + 1 \quad (15)$$

$$\sum_{k\in K}\sum_{w\in W} Y_{idkmw} \leq \sum_{j\in J}\sum_{w\in W} \left(s_{ij} * OP_{jw}\right) * \sum_{t\in T} X_{idtm}, \quad \forall d \in D, m \in N, i \in I \quad (16)$$

$$G_{idkw} \geq s_{ij} * \sum_{m\in N} X_{idtm} \text{ ,} \forall i \in I, j \in J, d \in D, t \in T, w \in W,$$

$$k = \mathrm{t} + OP_{jw}, \dots, t + OP_{jw} + PC_j - 1 \quad (17)$$

$$\sum_{i\in I} G_{idtw} \leq BPACU + OPACU, \qquad \forall\, d \in D, t \in T, w \in W \quad (18)$$

$$OPACU \leq UPACU, \quad (19)$$

$$\sum_{k\in I}\sum_{w\in W} G_{idkw} \leq W * PC_j, \qquad \forall\, i \in I, j \in J, d \in D, s_{ij} = 0{,}1 \quad (20)$$

$$X_{idtm} \leq F_{md} \text{ ,} \qquad \forall i \in I, m \in N, t \in T, d \in D \quad (21)$$

$$X_{i'd't'm'} * PL_{i'} - X_{idtm} * PL_i \leq M * (1 - X_{idtm}),$$

$$\forall i,i' \in I, m,m' \in N, t,t' \in T, d,d' \in D \quad (22)$$

$$\sum_{k \in K} k * z_{ik} \leq C_i, \qquad \forall i \in I \qquad (23)$$

$$\sum_{k \in K} z_{ik} \leq 1, \qquad \forall i \in I \qquad (24)$$

$$RP_{ii'} \geq z_{ik} + z_{i'k} - 1\,, \qquad \forall i,i' \in I, k \in K \qquad (25)$$

$$RP_{ii'} \geq 1 - \sum_{k \in K} z_{ik} - \sum_{k \in K} z_{i'k}\,, \qquad \forall i,i' \in I \qquad (26)$$

$$D_i, C_i, CMAX_{dm}, WC_{idtm}, F_{md}, X_{idtm}, Y_{idtmw}, OT_{mdw}, RP_{ii'} \geq 0\,,$$

$$\forall i,i' \in I, m \in N, t \in T, d \in D, w \in W \quad (27)$$

$$F_{md}, X_{idtm}, Y_{idtmw}, D_i, z_{ik} \quad binary\,, \quad \forall i \in I, m \in N, t \in T, d \in D, w \in W, k \in K \qquad (28)$$

The objective function of the RSM has eight different goals. Explanation of these objectives and constraints are showed in Table 3. 6.

Table 3. 6 Objectives and constraints of the RSM

Objectives
OBJ1: This objective function minimizes the total waiting cost of elective patients. OBJ2: This objective function minimizes the cost of deferring elective patients to next planning cycle. OBJ3: This objective function minimizes the penalty cost of surgery completion time repeats; thus, BIMs will be maximized. OBJ4: This objective function minimizes the cost of completion the last surgeries in ORs. OBJ5: This objective function minimizes the cost of opening ORs. OBJ6: This objective function minimizes the cost of overtime in ORs. OBJ7: This objective function minimizes the cost of overtime in PACU.

OBJ8: This objective function minimizes the total waiting cost of emergency patients.
Constraints
Constraint (1): This constraint guarantees that every patient will be either scheduled to have a surgery or deferred to next planning period. Constraint (2): This constraint ensures that all of the emergency patients will be scheduled to have a surgery in the BIMs. Constraint (3): This constraint guarantees that elective patients who already started or completed their surgeries cannot be rescheduled. Constraint (4): This constraint ensures that we cannot have more than 1 patient to start a surgery in any OR at the same time. Constraint (5): This constraint calculates the completion time of surgeries. Constraint (6): This constraint calculates the surgery completion time of patients for each day in each OR. Constraint (7): This constraint shows the last surgery completion time for each day in each OR. Constraint (8): This constraint guarantees that once a patient starts a surgery, we have to wait till that surgery plus turnover time end to start another surgery. Constraint (9): This constraint guarantees that the number of ongoing operations cannot be more than the number of ORs. Constraint (10): This constraint calculates the waiting cost of patients. Constraint (11): This constraint provides the link between the start and continuation of the SOs. Constraint (12): This constraint guarantees that the existing number of surgical teams will be

equal or more than the ongoing operations.

Constraint (13): This constraint determines the total utilization of the ORs in the planning cycle.

Constraint (14): This constraint calculates the amount of overtime utilized in ORs.

Constraint (15): This constraint ensures that we cannot have any ongoing operations outside of the planning period.

Constraint (16): This constraint ensures that decision variable Y_{idtmw} will be zero if a patient finishes his/her surgery.

Constraint (17): This constraint shows that patients will transfer and stay for a certain period of time in the PACU.

Constraint (18): This constraint shows that the current plus additional (if needed) capacity in PACU will be enough to satisfy the transferring patients from the ORs.

Constraint (19): This constraint determines the upper limit on the PACU capacity.

Constraint (20): This constraint makes sure that the decision variable of the PACU, G_{idtw}, will be zero if the PACU is empty or there is no patient in it.

Constraint (21): This constraint guarantees that an OR will be closed if there are no ongoing operations in that OR.

Constraint (22): This constraint is priority constraint.

Constraints (23) - (26): These constraints are the BIM constraints that calculate the BIMs.

Constraint (27): This constraint is the non-negativity constraint on all the decision variables.

Constraint (28): This constraint defines each $F_{md}, X_{idtm}, Y_{idtmw}, D_i,\ z_{ik}$ decision variable to be a binary variable.

3.3. A Mathematical Model for Rescheduling and Resequencing Elective Patients and Scheduling Emergency Patients by Considering Dedicated Rooms

In this section, a partially stochastic MILP model is developed for the RSDM. Emergency patients normally arrive unexpectedly and need to be operated on as soon as possible. When emergency patients arrive at a hospital, they are checked in based on the severity of their health condition and characterized into emergency levels. Some emergency patients require immediate attention. This MILP model considers three categories for emergency levels. The first category is emergent patients who require immediate operation, such as patients with massive bleeding, major burns, major car accident, or heart attack. The second category is urgent patients who require an operation within two hours, such as patients with head injury (conscious), eye inflammation, or breathing difficulties. The third and last category is non-urgent patients who require an operation on the same day of arrival or within six hours, such as patients with cuts not requiring stitches or minor trauma.

Figure 3. 2 shows the processing diagram of the RSDM. Elective patients come from a referring clinic or surgeon's office. Then, they go to the admission unit to sign in and wait in the waiting area to be taken to an available OR. If they are an elective inpatient, they wait in their inpatient beds until an OR is available for them. Conversely, emergency patients have stochastic arrivals and arrive in a critical health condition. Thus, when emergency patients arrive, they need to be treated within a certain time limit to avoid life-threatening situations.

After surgical emergency patients arrive at a hospital, they are taken to the emergency department where they are characterized and put into emergency levels. If they are emergent patients, elective ORs are checked to see if there is any BIM available for an immediate surgery. Otherwise, emergent patients will be taken to a dedicated room to have surgery immediately. If either there is no immediately available BIM in elective ORs or dedicated rooms are not available immediately, then emergent patients will be transferred to nearby hospitals. By assigning more dedicated rooms, it is possible to minimize the chance of this improper scenario, i.e. transferring patients to nearby hospitals.

The patients who are in the urgent category will be scheduled in elective ORs to have surgeries if there are available BIMs within two hours. Otherwise, those patients will be scheduled in dedicated rooms within two hours. If dedicated rooms or elective ORs are not available within that time, then they will be transferred to a nearby hospital. The non-urgent patients will be scheduled at the BIM in elective ORs within six hours or scheduled in the dedicated rooms within six hours if there is no BIM in elective rooms within that time. If they cannot be scheduled in the elective ORs or dedicated rooms due to unavailability, then they are transferred to nearby hospitals. All of the elective or emergency patients will be transferred to PACU unit after their surgeries and then discharged from the hospital or returned to their inpatient beds.

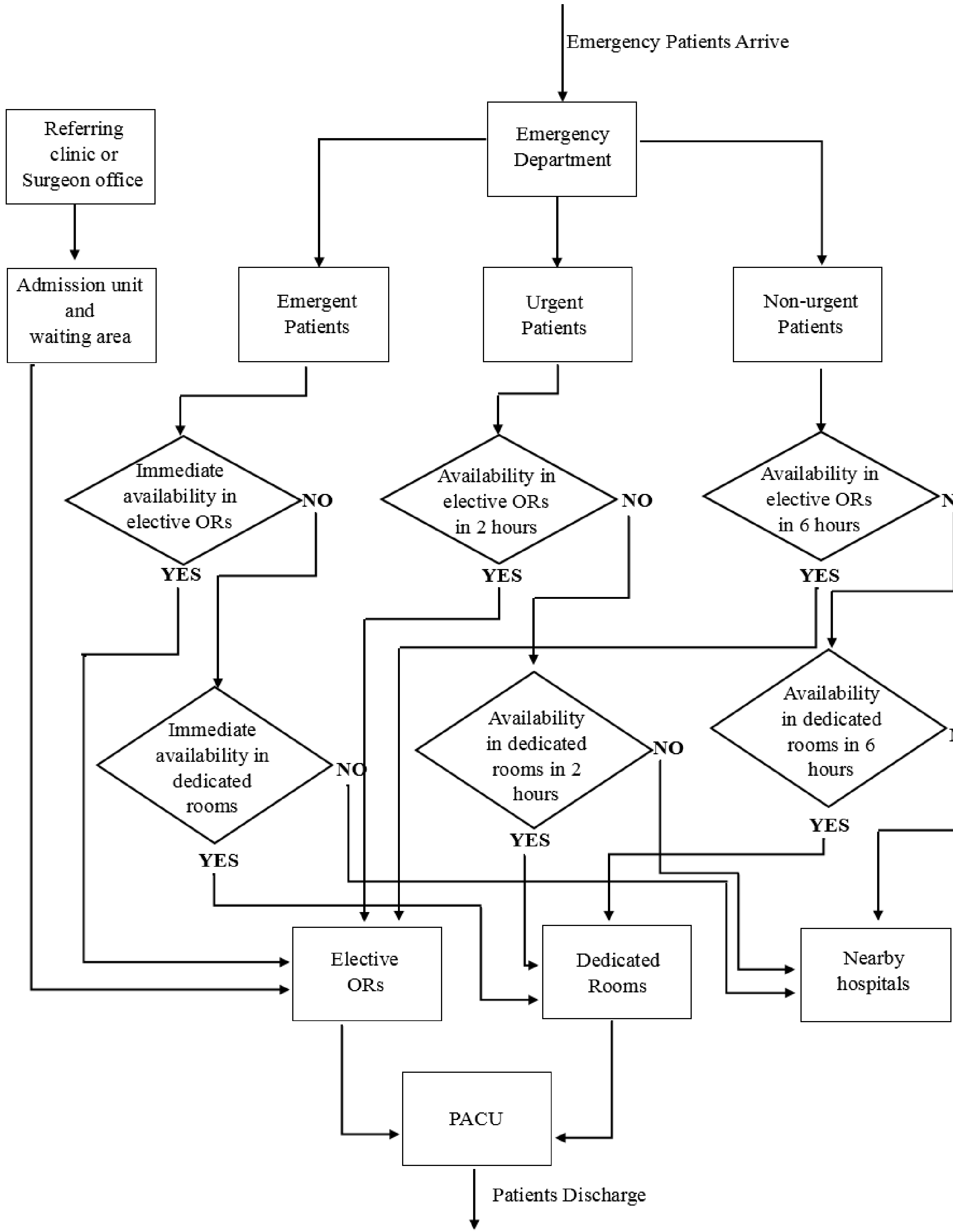

Figure 3. 2 Processing diagram of the RSDM

Table 3. 7 shows the notation for index and parameters used in the RSDM.

Table 3. 7 Notation for index and parameters of the RSDM

Indices	
i, i^2	: Elective and emergency patient indices; $i, i^2 \in \{1, \ldots, I\}$.
j	: SO type index; $j \in \{1, \ldots, J\}$.
t, t^2	: Time period indices; $t, t^2 \in \{1, \ldots, T\}$.
h, k	: Auxiliary time period indices; $h, k \in T + 3$.
d, d^2	: Day indices; $d \in \{1, \ldots, D\}$.
m, m^2	: OR indices; $m, m^2 \in \{1, \ldots, N\}$.
w	: Scenario index; $w \in \{1, \ldots, W\}$.
Parameters	
FC	: Fixed cost of opening an OR during planning cycle;
t_s	: Arrival time for emergency patients;
I_{PE}	: Total number of emergency patients;
N_{RD}	: Total number of dedicated rooms;
CTR_i	: Cost of transfer or loss of revenue for emergency patient i;
ET_P	: Emergent patients
UT_P	: Urgent patients
NUT_P	: Non-urgent patients
$FCDR$	: Fixed cost of dedicated rooms per hour use.
BIM_{mdt}	: Equal to 1 if there is a BIM in OR m on day d at time t;
FOS_{idtm}	: Equal to 1 if there is a finished or ongoing surgery for patient i on day d at time t in OR m when the emergency patients arrive, 0 otherwise;
MAX_i	: Maximum operation hours for patient i;
COR	: Overtime utilization cost of an OR during planning cycle (cost/hour);
$CPACU$	: Unit expansion cost of PACU during planning cycle (cost/bed);
$BPACU$	: Current capacity of the PACU in terms of beds;
$UPACU$	: Upper limit on the over-utilization of the PACU capacity in terms of beds;

CD	: Cost of deferring a patient to next planning cycle;
CC	: Cost of total completion time for all surgeries in each OR;
CR	: Penalty cost of repeating the completion times for surgeries;
OP_{jw}	: Operation time (hours) for surgery j under scenario w;
PC_j	: Length of stay (hours) at PACU for surgery type j;
RT	: Total number of regular working hours for ORs;
OT	: Total number of overtime hours for ORs;
s_{ij}	: Equal to 1 if patient i requests surgery type j, 0 otherwise;
OP_{iw}	: Operation time (hours) time for patient i under scenario w;
PL_i	: Priority level of patient i;
TO	: Turnover time (hours);
P_w	: Probability of scenario w;
WC_{id}	: Waiting cost of elective patient i on day d;
WT_i	: Waiting time (days) for elective patient i;
ECh_i	: Waiting cost for emergency patient i;
HS_i	: Hospitalization cost of elective patient i (cost/day);
M	: A sufficient large number;

The following calculation is used for converting operation hours of surgeries to operation hours of patients.

$$OP_{i,w} = \sum_{j \in J}(s_{ij} * OP_{j,w}), \quad \forall i \in I, w \in W$$

The following equations are used for calculating the cost of transfers or loss of revenue for each levels of emergency patients if they are transferred to nearby hospitals due to unavailable capacity.

$$CTR_i = 60000 * OP_{i,w} \quad , \quad \forall i \in ET_P, w \in W$$

$$CTR_i = 30000 * OP_{i,w} \quad , \quad \forall i \in UT_P, w \in W$$

$$CTR_i = 20000 * OP_{i,w} \quad , \quad \forall i \in NUT_P, w \in W$$

Table 3. 8 shows the notation for decision variables used in the RSDM.

Table 3. 8 Notation for decision variables of the RSDM

Decision Variables	
D_i	: Equal to 1 if an elective patient i is deferred to next planning cycle, 0 otherwise;
C_i	: Surgery completion time for patient i;
$CMAX_{dm}$	: The last surgery completion time on day d in OR m;
F_{md}	: Equal to 1 if an elective OR m is open on day d, 0 otherwise;
OVT_{mdw}	: Amount of overtime utilization of OR m on day d under scenario w;
$OPACU$	: Amount of additional capacity (beds) placed in PACU;
Y_{idtmw}	: Equal to 1 if patient (elective and emergency) i has a surgery on day d at time t in OR m under scenario w, 0 otherwise;
X_{idtm}	: Equal to 1 if a surgery starts on day d at time t in OR (elective or dedicated) m for patient (elective or emergency) i, 0 otherwise;
G_{idtw}	: Equal to 1 if a patient i occupies a bed in PACU on day d at time t under scenario w, 0 otherwise;
$z_{ik}, RP_{ii'}$	: Auxiliary decision variables to calculate the BIMs;
TR_i	: Equal to 1 if an emergency patient i is transferred to nearby hospital, 0 otherwise;
FDR_{mdt}	: Equal to 1 if a dedicated room m is open on day d at time t, 0 otherwise;

The RSDM with ten different objectives $(OBJ1 + OBJ2 + OBJ3 + OBJ4 + OBJ5 + OBJ6 + OBJ7 + OBJ8 + OBJ9 + OBJ10)$ is developed as follows:

$$Minimize\ OBJ1 + OBJ2 + OBJ3 + OBJ4 + OBJ5 + OBJ6 + OBJ7 + OBJ8 + OBJ9 + OBJ10$$

$$OBJ1 = \sum_{i \in I} \sum_{d \in D} \sum_{t \in T} \sum_{m \in N} (X_{idtm} * WC_{id})$$

$$OBJ2 = \sum_{i \in I} (CD * D_i)$$

$$OBJ3 = \sum_{i \in I} \sum_{i' \in I} (CR * RP_{ii'})$$

$$OBJ4 = \sum_{d \in D} \sum_{m \in N} CC * CMAX_{dm}$$

$$OBJ5 = \sum_{m \in N} \sum_{d \in D} (FC * F_{md})$$

$$OBJ6 = \sum_{m \in N} \sum_{d \in D} \sum_{w \in W} (P_w * COR * OVT_{mdw})$$

$$OBJ7 = \sum_{w \in W} (P_w * CPACU * OPACU)$$

$$OBJ8 = \sum_{i \in \{I - I_{PE}+1, \dots, I\}} \sum_{d \in BIM_{mdt}=1} \sum_{t \geq t_s, t \in BIM_{mdt}=1} \sum_{m \in BIM_{mdt}=1} (ECh_i * (t - t_s) * X_{idtm})$$

$$OBJ9 = \sum_{i \in \{I - I_{PE}+1, \dots, I\}} (CTR * TR_i)$$

$$OBJ10 = \sum_{m \in N} \sum_{d \in D} \sum_{t \in T} (FCDR * FDR_{mdt})$$

Subject to

$$\sum_{d \in D} \sum_{t \in T} \sum_{m} X_{idtm} + D_i = 1, \qquad \forall i \in \{1, \dots, I - I_{PE}\}, m \in \{1, \dots, N - N_{RD}\} \qquad (1)$$

$$\sum_{d} \sum_{t} \sum_{m} X_{idtm} + TR_i = 1,$$
$$\forall i \in \{I - I_{PE} + 1, \dots, I\}, t \in \{t_s, \dots, T\}, d, t, m \in BIM_{mdt} \qquad (2)$$

$$X_{idtm} * \ (t - t_s) \leq 0, \qquad \forall\ i \in ET_P, d \in D, t \in \{t_s, \dots, T\}, m \in N \qquad (3)$$

$$X_{idtm} * \ (t - t_s) \leq 2, \qquad \forall\ i \in UT_P, d \in D, t \in \{t_s, \dots, T\}, m \in N \qquad (4)$$

$$X_{idtm} * \ (t - t_s) \leq 6, \qquad \forall\ i \in NUT_P, d \in D, t \in \{t_s, \dots, T\}, m \in N \qquad (5)$$

$$Y_{idtmw} \leq FDR_{mdt}\ , \forall i \in \{I - I_{PE} + 1, \dots, I\}, m \in \{N - N_{RD} + 1, \dots, N\}, t \in T, d \in D \qquad (6)$$

$$X_{idtm} = 1, \qquad i, d, t, m \in FOS_{idtm} \qquad (7)$$

$$\sum_{i \in I} X_{idtm} \leq 1, \qquad \forall d \in D, t \in T, m \in N \qquad (8)$$

$$\sum_{d \in D} \sum_{t \in T} \sum_{m \in N} \Big(t + MAX_i + \big((d-1) * T\big)\Big) * X_{idtm} = C_i, \qquad \forall i \in I \qquad (9)$$

$$\sum_{t \in T} (t + MAX_i) * X_{idtm} = ORC_{idm}, \qquad \forall i \in I, d \in D, m \in N \qquad (10)$$

$$ORC_{idm} \leq CMAX_{dm}, \qquad \forall i \in I, d \in D, m \in N \qquad (11)$$

$$\sum_{i \in I} \sum_{h} X_{idhm} \leq 1, \forall d \in D, t \in T, m \in N, w \in W, h = \max\big(1, t - OP_{i,w} + 1\big) - TO, \dots, t \quad (12)$$

$$\sum_{i \in I} Y_{idtmw} \leq N, \qquad \forall m \in N, d \in D, t \in T, w \in W \qquad (13)$$

$$WC_{id} = HS_i * WT_i * d, \qquad \forall i \in \{1, \dots, I - I_{PE}\}, d \in D \qquad (14)$$

$$Y_{idkmw} \geq s_{ij} * \sum_{m \in N} X_{idtm}, \forall m \in N, i \in I, j \in J, d \in D, w \in W, k = \mathrm{t}, \dots, t + OP_{jw} - 1, \quad (15)$$

$$\sum_{i \in I} \sum_{m \in N} s_{ij} * Y_{idtmw} \leq ST_{jdt}, \qquad \forall j \in J, d \in D, t \in T, w \in W \qquad (16)$$

$$\sum_{i \in I} \sum_{t} Y_{idtmw} \leq RT, \qquad \forall m \in N, d \in D, t \in \{1, \dots, RT\}, w \in W \qquad (17)$$

$$\sum_{i \in I} \sum_{t} Y_{idtmw} = OVT_{mdw}, \qquad \forall m \in N, d \in D, t \in \{RT + 1, \dots, RT + OT\}, w \in W \qquad (18)$$

$$\sum_{i \in I} \sum_{t} Y_{idtmw} = 0, \qquad \forall m \in N, d \in D, t \geq RT + OT + 1, w \in W \qquad (19)$$

$$\sum_{k \in K} \sum_{w \in W} Y_{idkmw} \leq \sum_{j \in J} \sum_{w \in W} \big(s_{ij} * OP_{jw}\big) * \sum_{t \in T} X_{idtm}, \quad \forall d \in D, m \in N, i \in I \qquad (20)$$

$$G_{idkw} \geq s_{ij} * \sum_{m \in N} X_{idtm}, \forall i \in I, j \in J, d \in D, t \in T, w \in W,$$

$$k = \mathrm{t} + OP_{jw}, \dots, t + OP_{jw} + PC_j - 1 \qquad (21)$$

$$\sum_{i\in I} G_{idtw} \leq BPACU + OPACU, \qquad \forall\, d \in D, t \in T, w \in W \qquad (22)$$

$$OPACU \leq UPACU, \qquad (23)$$

$$\sum_{k\in I}\sum_{w\in W} G_{idkw} \leq W * PC_j, \qquad \forall\, i \in I, j \in J, d \in D, s_{ij} = 0{,}1 \qquad (24)$$

$$X_{idtm} \leq F_{md}\,, \qquad \forall i \in I, m \in m \in \{1, \ldots, N - N_{RD}\}, t \in T, d \in D \qquad (25)$$

$$X_{i'd't'm'} * PL_{i'} - X_{idtm} * PL_i \leq M * (1 - X_{idtm}),$$

$$\forall i, i' \in I, m, m' \in N, t, t' \in T, d, d' \in D \qquad (26)$$

$$\sum_{k\in K} k * z_{ik} \leq C_i, \qquad \forall i \in I \qquad (27)$$

$$\sum_{k\in K} z_{ik} \leq 1, \qquad \forall i \in I \qquad (28)$$

$$RP_{ii'} \geq z_{ik} + z_{i'k} - 1\,, \qquad \forall i, i' \in I, k \in K \qquad (29)$$

$$RP_{ii'} \geq 1 - \sum_{k\in K} z_{ik} - \sum_{k\in K} z_{i'k}\,, \qquad \forall i, i' \in I \qquad (30)$$

$$D_i, C_i, CMAX_{dm}, WC_{idtm}, F_{md}, X_{idtm}, Y_{idtmw}, OT_{mdw}, RP_{ii'} \geq 0\,,$$

$$\forall i, i' \in I, m \in N, t \in T, d \in D, w \in W \qquad (31)$$

$$F_{md}, X_{idtm}, Y_{idtmw}, FDR_{mdt}, D_i, z_{ik} \quad binary\,,$$

$$\forall i \in I, m \in N, t \in T, d \in D, w \in W, k \in K \qquad (32)$$

The objective function of the RSDM has ten different goals. Explanation of these objectives and constraints are showed in Table 3. 9.

Table 3. 9 Objectives and constraints of the RSDM

Objectives
OBJ1: This objective function minimizes the total waiting cost of elective patients. OBJ2: This objective function minimizes the cost of deferring elective patients to next planning cycle. OBJ3: This objective function minimizes the penalty cost of surgery completion time repeats; thus, BIMs will be maximized. OBJ4: This objective function minimizes the cost of completion the last surgeries in ORs. OBJ5: This objective function minimizes the cost of opening ORs. OBJ6: This objective function minimizes the cost of overtime in ORs. OBJ7: This objective function minimizes the cost of overtime in PACU. OBJ8: This objective function minimizes the total waiting cost of emergency patients. OBJ9: This objective function minimizes the cost of transfer for emergency patients. OBJ10: This objective function minimizes the usage cost of dedicated rooms.
Constraints
Constraint (1): This constraint guarantees that every elective patient will be either scheduled to have a surgery in elective ORs or deferred to next planning period. Constraint (2): This constraint ensures that all of the emergency patients will be scheduled to have a surgery in the BIMs or transferred to nearby hospitals. Constraint (3): This constraint shows that the waiting time limit for emergent patients is zero. Constraint (4): This constraint shows that the waiting time limit for urgent patients is 2 hours. Constraint (5): This constraint shows that the waiting time limit for non-urgent patients is 6

hours.

Constraint (6): This constraint ensures that dedicated rooms are only used if there is no available capacity in elective rooms for emergency patients.

Constraint (7): This constraint guarantees that elective patients who already started or completed their surgeries cannot be rescheduled.

Constraint (8): This constraint ensures that we cannot have more than 1 patient to start a surgery in any OR at the same time.

Constraint (9): This constraint calculates the completion time of surgeries.

Constraint (10): This constraint calculates the surgery completion time of patients for each day in each OR.

Constraint (11): This constraint shows the last surgery completion time for each day in each OR.

Constraint (12): This constraint guarantees that once a patient starts a surgery, we have to wait till that surgery plus turnover time end to start another surgery.

Constraint (13): This constraint guarantees that the number of ongoing operations cannot be more than the number of ORs.

Constraint (14): This constraint calculates the waiting cost of patients.

Constraint (15): This constraint provides the link between the start and continuation of the SOs.

Constraint (16): This constraint guarantees that the existing number of surgical teams will be equal or more than the ongoing operations.

Constraint (17): This constraint determines the total utilization of the ORs in the planning cycle.

Constraint (18): This constraint calculates the amount of overtime utilized in ORs.

Constraint (19): This constraint ensures that we cannot have any ongoing operations outside of the planning period.

Constraint (20): This constraint ensures that decision variable Y_{idtmw} will be zero if a patient finishes his/her surgery.

Constraint (21): This constraint shows that patients will transfer and stay for a certain period of time in the PACU.

Constraint (22): This constraint shows that the current plus additional (if needed) capacity in PACU will be enough to satisfy the transferring patients from the ORs.

Constraint (23): This constraint determines the upper limit on the PACU capacity.

Constraint (24): This constraint makes sure that the decision variable of the PACU, G_{idtw}, will be zero if the PACU is empty or there is no patient in it.

Constraint (25): This constraint guarantees that an OR will be closed if there are no ongoing operations in that OR.

Constraint (26): This constraint is priority constraint.

Constraints (27) - (30): These constraints are the BIM constraints that calculate the BIMs.

Constraint (31): This constraint is the non-negativity constraint on all the decision variables.

Constraint (32): This constraint defines each $F_{md}, X_{idtm}, Y_{idtmw}, D_i,\ z_{ik}$ decision variable to be a binary variable.

CHAPTER 4: RESULTS and DISCUSSIONS

In this chapter, them SM, RSM, and RSDM are solved to optimality and the results are presented. First, the data needed to solve these MILP models are provided and explained.

4.1.Data

The data related to type and duration of surgeries are taken from Erdem [29], as shown in Table 4. 1.

Table 4. 1 Type and duration of surgeries

Surgical Operation	Duration of operation (min)			Corresponding probability		
Cardio-Vascular (CV)	240			1		
Ear-Nose-Throat (ENT)	60	120		0.6	0.4	
General Surgery	60	120	180	0.2	0.5	0.3
Hand	60			1		
Neurology	240			1		
OB/GYN	120			1		
Ophthalmology	60	120		0.8	0.2	
Orthopedics	120			1		
Podiatry	120			1		
Urology	60	120		0.6	0.4	

The data in also shows the corresponding probability for each type of surgery. The optimization software, LINGO, is unable to find a global optimal solution using the data in in a reasonable time, so the data are converted into a simplified version, which is shown in Table 4. 2.

Table 4. 2 Simplified data

Surgical Operation (SO)	Duration of Operation (DO) (min)	Corresponding Probability (CP)
Cardio-Vascular	240	1
Ear-Nose-Throat	120	1
General Surgery	120 | 180	0.5 | 0.5
Neurology	180	1
OB/GYN	120	1
Ophthalmology	120 | 180	0.5 | 0.5
Orthopedics	240	1
Hand	60	1

Using the data in Table 4. 2, different scenarios are created for the duration of SOs. Table 4. 3 shows the Duration of Operation (DOs) in hours for each scenario and each type of SOs. It is assumed that each scenario has a Corresponding Probability (CP) of 0.25. There are eight patients in the system waiting to have a SO with a Waiting Time (WT) of two days and hospitalization (HS) cost of $300 daily for each patient. Turnover time is equal to one hour for each SO [40]. Priority Level (PL) is 1 for each patient, except Patient 2, who has a higher PL, and equals 2.

Table 4. 3 DOs in hours for each scenario and each type of SOs

SO	DO (hours)				Patients	PL	WT (days)	HS ($/day)	Turnover (hr)
Cardio-Vascular	4	4	4	4	1	1	2	$300	1
Ear-Nose-Throat	2	2	2	2	2	2	2	$300	1
General Surgery	2	2	3	3	3	1	2	$300	1
Neurology	3	3	3	3	4	1	2	$300	1
OB/GYN	2	2	2	2	5	1	2	$300	1
Ophthalmology	2	3	2	3	6	1	2	$300	1
Orthopedics	4	4	4	4	7	1	2	$300	1
Hand	1	1	1	1	8	1	2	$300	1
Scenarios	1	2	3	4					
CP	0.25	0.25	0.25	0.25					

Table 4. 4 shows the planning cycle, which is one day, number of available ORs, which is three, and available time for SOs, which is eight hours for regular time and two hours for overtime. Available number of surgical teams for SOs is shown in Table 4. 5.

Table 4. 4 The planning cycle

		Regular Time								Overtime	
		1	2	3	4	5	6	7	8	**1**	**2**
	ORs	08:00-09:00	09:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	13:00-14:00	14:00-15:00	15:00-16:00	**16:00-17:00**	**17:00-18:00**
Day 1	**OR1**										
	OR2										
	OR3										

Table 4. 5 Available number of surgical teams

	Day 1 Available Number of the Surgical Teams									
SO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Cardio-Vascular	1	1	1	1	1	1	1	1	1	1
Ear-Nose-Throat	1	1	1	1	1	1	1	1	1	1
General Surgery	1	1	1	1	1	1	1	1	1	1
Neurology	1	1	1	1	1	1	1	1	1	1
OB/GYN	1	1	1	1	1	1	1	1	1	1
Ophthalmology	1	1	1	1	1	1	1	1	1	1
Orthopedics	1	1	1	1	1	1	1	1	1	1
Hand	1	1	1	1	1	1	1	1	1	1

Cost of repeating or penalty cost of having the same completion time for SOs is $5,000. Fixed cost of opening an OR is $2,500. The cost of completing the last SO in each OR and the cost of overtime are $1,000 per hour. Current capacity of the PACU is three beds and length of stay is one hour for each SO. Bed expansion cost in PACU is $4,000 per bed and upper limit is one bed. Cost of deferring a patient to the next planning cycle is $15,000.

4.2. Results of the SM

Table 4. 6 provides the scheduling and sequencing results of the SM. Table 4. 7 shows the cost results of each objective considered in the SM. It took 28 minutes and 28 seconds to solve in Lingo 18.

Table 4. 6 Scheduling and sequencing results of the SM

		Regular Time								Overtime	
		1	2	3	4	5	6	7	8	**1**	**2**
	ORs	08:00-09:00	09:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	13:00-14:00	14:00-15:00	15:00-16:00	**16:00-17:00**	**17:00-18:00**
Day 1	**OR1**	Patient 8		Patient 7	Patient 7	Patient 7	Patient 7		Patient 3	Patient 3	*Patient 3*
	OR2	Patient 2	Patient 2		Patient 5	Patient 5		Patient 6	Patient 6	*Patient 6*	
	OR3	Patient 4	Patient 4	Patient 4		Patient 1	Patient 1	Patient 1	Patient 1		

BIM BIM BIM BIM BIM BIM BIM BIM

Table 4. 7 Cost results of the SM

Cost Functions	Cost ($)
Waiting Time	4800
Deferring	0
BIM	0
Completion	30000
ORs	7500
OR Overtime	2000
PACU Overtime	0

4.3. Explanation of the Solution of the SM

It can be seen from Table 4. 6 that some patients, such as patients 3 and 6, require some overtime to finish their SOs. Since Patient 2 has a higher priority level than other patients, that patient starts to have an SO at time 1 in OR 2. None of the patients are deferred to the next planning cycle. The most important part of the solution is minimizing the total waiting time of the emergency patients or maximizing the number of the BIMs. The first BIM is when none of the patients start their SO because, if an emergency patient arrives at that time, that emergency patient can have SO in any of the ORs. The second BIM is when Patient 8 finishes the SO. If an emergency patient arrives at that time, that emergency patient can have SO in OR1 after Patient 8. As seen in Table 4. 7, the cost of total waiting time is $4,800, since all of the eight patients are scheduled on Day 1 with a waiting time of two days and $300 daily costs. There is no patient who is deferred to the next planning cycle, so the cost of deferring is zero. Completion times for all patients are different, which makes the cost of BIM zero. Patient 3 finishes his/her surgery at time 11 in OR1, Patient 6 finishes his/her surgery at time 10 in OR2, and Patient 1 finishes his/her surgery at time 9 in OR 3. Based on these finish times, cost of completion is $30,000. All of the three ORs are used for scheduling, which makes the cost of ORs $7,500. There may be one or two hours of overtime for OR1 and one or zero hours of overtime in OR2, so the cost of overtime is $2,000. Having three beds in PACU will be enough for transferring patients from ORs to recovery area; it is not needed to add extra beds. Thus, the cost of overtime in PACU is zero.

4.4. Results of the RSM

To investigate how the RSM works, it is assumed that some emergency patients arrive and there are some changes in the surgical durations. In addition to data given in the first section regarding the number of patients and the type of surgeries they request, there are two emergency patients, Patients 9 and 10, arriving for an SO at time 3 (09:00-10:00). Waiting cost of emergency Patient 9 is $15,000/hr while the waiting cost of emergency Patient 10 is $5,000/hr. The other disruption is surgical duration for Patient 2, which increases from two hours to three hours. Table 4. 8 shows elective and arriving emergency patients with the type of surgery they request. Table 4. 9 shows the scheduling and sequencing results of the RSM. It is seen from Table 4. 9 that emergency Patient 9 is operated on after Patient 8 in OR1, while emergency patient 10 is operated on after Patient 2 in OR2. Patient 7 is deferred to next planning cycle. As seen in Table 4. 10, the cost of total waiting time for elective patients is $4,200 since one patient, Patient 7, is deferred to next planning cycle. Cost of deferring is $15,000 because of Patient 7. Completion times for Patients 3 and 5 are the same, which makes the cost of BIM $5,000. Patient 1 finishes his/her surgery at time 10 in OR 1, Patient 3 finishes his/her surgery at time 11 in OR2, and Patient 5 finishes his/her surgery at time 11 in OR3. Based on these finish times, cost of completion is

$3,2000. All of the three ORs are used for scheduling, which makes the cost of ORs $7,500. Patient 1 uses one hour of overtime in OR1, Patient 3 uses one or two hours of overtime in OR2, and Patient 5 uses two hours of overtime in OR3, so cost of overtime is $4,500. Cost of overtime in PACU is zero, since having three beds in PACU will be enough for transferring patients from ORs to recovery area.

Table 4. 8 Elective and emergency patients with surgeries they request

SO	DO (hours)				Patients	
Cardio-Vascular	4	4	4	4	1	
Ear-Nose-Throat	3	3	3	3	2	
General Surgery	2	2	3	3	3	
Neurology	3	3	3	3	4	**10**
OB/GYN	2	2	2	2	5	**9**
Ophthalmology	2	3	2	3	6	
Orthopedics	4	4	4	4	7	
Hand	1	1	1	1	8	
CP	**0.25**	**0.25**	**0.25**	**0.25**		

Table 4. 9 Scheduling and sequencing results of the RSM

		Regular Time								Overtime	
		1	2	3	4	5	6	7	8	**1**	**2**
	ORs	08:00-09:00	09:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	13:00-14:00	14:00-15:00	15:00-16:00	**16:00-17:00**	**17:00-18:00**
Day 1	**OR1**	Patient 8		**Patient 9**	**Patient 9**		Patient 1	Patient 1	Patient 1	Patient 1	
	OR2	Patient 2	Patient 2	Patient 2	**Patient 10**	**Patient 10**	**Patient 10**		Patient 3	Patient 3	*Patient 3*
	OR3	Patient 4	Patient 4	Patient 4		Patient 6	Patient 6	*Patient 6*		Patient 5	Patient 5

BIM BIM BIM BIM

Table 4. 10 Cost results of the RSM

Cost Functions	Cost ($)
Elective Waiting Time	4200
Emergency Waiting Time	5000
Deferring	15000
BIM	5000
Completion	32000
ORs	7500
OR Overtime	4500
PACU Overtime	0

4.5. Results of the RSDM

When emergency patients arrive with a SO need, they are categorized into emergency levels based on their health conditions. As explained in Chapter 3, there are three categories, namely, emergent, urgent, and non-urgent for emergency patients. It is assumed that four emergency patients arrive at the same time between 09:00am and 10:00am. After emergency patients arrive, they directly go to the emergency department and are checked for emergency conditions. It is assumed that they are categorized as two of them being emergent, one being urgent, and one being non-urgent. Since emergency patients arrive between 09:00am and 10:00am and spend some time in the emergency department to be checked for the emergency levels, the earliest time they can start for an SO is at 10:00 or time 3. Thus, in the RSDM, it is assumed that emergency patients arrive at time 3. Table 4. 11 shows the elective and arriving emergency patients with the type of surgery they need.

Table 4. 11 Elective and emergency patients with the surgeries they request

SO	DO (hours)				Elective Patients	Emergency Patients		
						Emergent	Urgent	Non-Urgent
Cardio-Vascular	4	4	4	4	1			
Ear-Nose-Throat	2	2	2	2	2	**9**		
General Surgery	2	2	3	3	3			
Neurology	3	3	3	3	4	**11**		
OB/GYN	2	2	2	2	5		**10**	
Ophthalmology	2	3	2	3	6			
Orthopedics	4	4	4	4	7			
Hand	1	1	1	1	8			**12**
CP	**0.25**	**0.25**	**0.25**	**0.25**				

Waiting costs of emergency patients are \$15,000/hr for Patient 9, \$3,000/hr for patient 10, \$5,000/hr for Patient 11, and \$2,000/hr for Patient 12. Cost of transfers or loss of revenues for emergency patients are \$120,000 for patient 9, \$60,000 for Patient 10, \$180,000 for Patient 11, and \$20,000 for Patient 12. Cost of using dedicated rooms per hour is \$10,000. Table 4. 12 shows the scheduling and sequencing results of the RSDM.

Table 4. 12 Scheduling and sequencing results of the RSDM

		Regular Time								**Overtime**	
		1	2	3	4	5	6	7	8	**1**	**2**
	ORs	08:00-09:00	09:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	13:00-14:00	14:00-15:00	15:00-16:00	**16:00-17:00**	**17:00-18:00**
Day 1	**OR1**	Patient 8		**Patient 11**	**Patient 11**	**Patient 11**		Patient 6	Patient 6	*Patient 6*	
	OR2	Patient 2	Patient 2		**Patient 10**	**Patient 10**	**Patient 10**		Patient 3	Patient 3	*Patient 3*
	OR3	Patient 4	Patient 4	Patient 4		**Patient 12**		Patient 5	Patient 5		
	DR			**Patient 9**	**Patient 9**						

BIM BIM BIM BIM BIM

Since Patients 9 and 11 are emergent patients and need to have a surgery immediately, the RSDM schedules Patient 9 in the dedicated room and Patient 11 in the OR1 at time 3 or at 10:00am immediately. Patient 10 is an urgent patient who needs to have a surgery within two hours and is scheduled in the OR2 at time 4 or 11:00am. Patient 12 is a non-urgent patient who needs to have a surgery within six hours and is scheduled in the OR3 at time 5 or 12:00pm. Elective Patients 1 and 7 are deferred to the next planning cycle due to unavailable capacity. Table 4. 13 shows the minimized cost results of the RSDM. A total of six elective patients are scheduled for Day 1 and this makes elective patients' total waiting cost \$3,600. Emergency Patients 9 and 11 are scheduled immediately after they arrive. Emergency Patients 10 and 12 waited one and two hours, respectively, to be scheduled. Thus, emergency patients' total waiting cost is \$7,000. Elective Patients 1 and 7 were deferred to the next planning cycle, so cost of deferring is \$30,000. Cost of transfer is zero, since none of the patients were transferred. Patients 11 and 12 have the same completion times, so cost of BIM is \$5,000. Completion times of ORs are time of 10 for OR1, time of 11 for OR2, time of 9 for OR3, and time of 5 for DR, so cost of completion is \$35,000. All three elective ORs are used, which makes cost of ORs \$7,500. Patient 9 uses two hours of DR and this makes the cost of DR \$10,000. Patient 6 may use one hour of overtime in OR1, patient 3 may use one or two hours of overtime in OR2. Cost of overtime in ORs is $2,000. Cost of PACU overtime is zero, since no extra beds are needed.

Table 4. 13 Cost results of the RSDM

Cost Functions	Cost ($)
Elective Waiting Time	3600
Emergency Waiting Time	7000
Deferring	30000
Transfer	0
BIM	5000
Completion	35000
ORs	7500
DRs	10000
OR Overtime	2000
PACU Overtime	0

4.6. Solving MILP Models using Genetic Algorithm

In this section, Genetic Algorithm (GA) is used to solve the MILP models. Metaheuristic algorithms, such as Genetic Algorithm, Simulated Annealing, Tabu Search, etc., are search algorithms that can provide a proper or close-to-optimal solution for optimization problems. Unlike optimization algorithms, such as Lingo, Lindo, Cplex, Gams, etc., which can provide and guarantee an optimal solution, metaheuristic algorithms do not guarantee an optimal solution. When optimization algorithms are not able to solve or find an optimal solution in a reasonable time due to the size of the problem, then metaheuristic algorithms are applied to find solutions with less computational effort in a reasonable time.

In this study, LINGO 18 is used to find the global optimum solution for the small-scale data, such as 1-day planning cycle with three ORs and eight patients. Even so, it took 28 minutes to get the solution in LINGO 18 with this data. When the problem size is increased for more than 1-day planning cycle, then LINGO 18 is unable to find a good or global optimum solution in a reasonable time, such as one or two hours. Thus, GA is developed to find a good feasible solution for the large-scale data such as 5-day planning cycle with four ORs and 70 patients.

A number of scenarios based on DOs are investigated based on the data shown in and Table 4. 14. Assuming there are 70 elective patients waiting for surgery, Table shows waiting time (days) of the patients and the type of surgery they request. For instance, Patients 1, 2, 3, and 4 are waiting for two days to have Cardio-Vascular surgery, and so on. Hospitalization cost is $300 per day for elective patients. Here, the priority is the same for all patients except Patients 2, 15, 28, and 50 who have higher priority levels. The planning cycle is for five days with four ORs. There is one available surgical team for all times. There will be no change in other data, such as regular and overtime hours, number of beds in PACU, stay time in PACU, etc. Appendix A provides the relevant MATLAB codes for solving MILP Models using Genetic Algorithm.

Table 4. 14 Scenarios for surgical operations

Surgical Operation	**Duration of Operation (hours)**																							
Cardio-Vascular	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
Ear-Nose-Throat	1	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2
General Surgery	1	1	1	1	2	2	2	2	3	3	3	3	1	1	1	1	2	2	2	2	3	3	3	3
Hand	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Neurology	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
OB/GYN	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Ophthalmology	1	1	1	1	2	2	2	2	1	1	1	1	2	2	2	2	1	1	1	1	2	2	2	2
Orthopedics	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Podiatry	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Urology	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
Corresponding Probability	**0.06**	**0.04**	**0.01**	**0.01**	**0.14**	**0.1**	**0.04**	**0.02**	**0.09**	**0.06**	**0.02**	**0.01**	**0.04**	**0.03**	**0.01**	**0.01**	**0.1**	**0.06**	**0.02**	**0.02**	**0.06**	**0.04**	**0.01**	**0.01**

Table 4. 15 Patient waiting time and surgery requests

Surgical Operation	Patients	Waiting Time (days)
Cardio-Vascular	1,2,3,4	2
Ear-Nose-Throat	5	2
	6,7,8	3
	9	4
	10	1
	11,12,13,14	2
General Surgery	15,21,22,23,24,25,31,32,33,34,35	2
	16,17,18,26,27,28	3
	19,29	4
	20,30	1
Hand	36,37,38	3
Neurology	39	4
	40	1
	41	2
OB/GYN	42,43,44,45	2
	46,47,48	3
Ophthalmology	49	4
	50	1
	51	2
Orthopedics	52,53,54,55,56,57,58,59,60,61,62	2
Podiatry	63,64,65,66	2
Urology	67,68,69,70	2

4.7. Results of the SM using Genetic Algorithm

Table 4. 16 through Table 4. 20 show the results for OR scheduling Day 1 through Day 5. Patients shown in Italics have stochastic surgical durations. The red dots provided on the left side of the scheduling for each day show where the BIMs are. Patients 27, 29, 30, 31, 32, 33, 34, 35, 69, and 70 are deferred for the next planning cycle.

Table 4. 16 Operating room scheduling for day 1

Day 1					
		ORs			
	Time	OR1	OR2	OR3	OR4
Regular Time	08:00-09:00	Patient 2	Patient 15	Patient 8	Patient 56
	09:00-10:00	Patient 2	*Patient 15*	*Patient 8*	
	10:00-11:00	Patient 2	*Patient 15*		Patient 10
	11:00-12:00	Patient 2		Patient 63	*Patient 10*
	12:00-13:00		Patient 21	Patient 63	
	13:00-14:00	Patient 50	*Patient 21*		Patient 40
	14:00-15:00	*Patient 50*	*Patient 21*	Patient 67	Patient 40
	15:00-16:00			*Patient 67*	Patient 40
Overtime	16:00-17:00	Patient 45			Patient 40
	17:00-18:00	Patient 45			

Table 4. 17 Operating room scheduling for day 2

Day 2					
		ORs			
	Time	OR1	OR2	OR3	OR4
Regular Time	08:00-09:00	Patient 28	Patient 6	Patient 58	Patient 1
	09:00-10:00	*Patient 28*	*Patient 6*		Patient 1
	10:00-11:00	*Patient 28*		Patient 11	Patient 1
	11:00-12:00		Patient 39	*Patient 11*	Patient 1
	12:00-13:00	Patient 20	Patient 39		
	13:00-14:00	*Patient 20*	Patient 39	Patient 36	Patient 66
	14:00-15:00	*Patient 20*	Patient 39		Patient 66
	15:00-16:00			Patient 38	
Overtime	16:00-17:00				Patient 68
	17:00-18:00			Patient 54	*Patient 68*

Table 4. 18 Operating room scheduling for day 3

Day 3					
		ORs			
	Time	**OR1**	**OR2**	**OR3**	**OR4**
Regular Time	**08:00-09:00**	Patient 7	Patient 60	Patient 16	Patient 46
	09:00-10:00	*Patient 7*		*Patient 16*	Patient 46
	10:00-11:00		Patient 12	*Patient 16*	
	11:00-12:00	Patient 51	*Patient 12*		Patient 19
	12:00-13:00	*Patient 51*		Patient 3	*Patient 19*
	13:00-14:00		Patient 37	Patient 3	*Patient 19*
	14:00-15:00	Patient 24		Patient 3	
	15:00-16:00	*Patient 24*	Patient 53	Patient 3	Patient 42
Overtime	**16:00-17:00**	*Patient 24*			Patient 42
	17:00-18:00		Patient 57		

Table 4. 19 Operating room scheduling for day 4

Day 4					
		ORs			
	Time	**OR1**	**OR2**	**OR3**	**OR4**
Regular Time	**08:00-09:00**	Patient 9	Patient 61	Patient 17	Patient 47
	09:00-10:00	*Patient 9*		*Patient 17*	Patient 47
	10:00-11:00		Patient 13	*Patient 17*	
	11:00-12:00	Patient 64	*Patient 13*		Patient 22
	12:00-13:00	Patient 64		Patient 41	*Patient 22*
	13:00-14:00		Patient 4	Patient 41	*Patient 22*
	14:00-15:00	Patient 25	Patient 4	Patient 41	
	15:00-16:00	*Patient 25*	Patient 4	Patient 41	Patient 43
Overtime	**16:00-17:00**	*Patient 25*	Patient 4		Patient 43
	17:00-18:00				

Table 4. 20 Operating room scheduling for day 5

Day 5					
		ORs			
	Time	OR1	OR2	OR3	OR4
Regular Time	08:00-09:00	Patient 62	Patient 18	Patient 5	Patient 48
	09:00-10:00		*Patient 18*	*Patient 5*	Patient 48
	10:00-11:00	Patient 14	*Patient 18*		
	11:00-12:00	*Patient 14*		Patient 49	Patient 23
	12:00-13:00		Patient 52	*Patient 49*	*Patient 23*
	13:00-14:00	Patient 26			*Patient 23*
	14:00-15:00	*Patient 26*	Patient 55	Patient 65	
	15:00-16:00	*Patient 26*		Patient 65	Patient 44
Overtime	16:00-17:00		Patient 59		Patient 44
	17:00-18:00				

Table 4. 21 provides the minimized costs for each objective of the SM for the large-scale data.

Table 4. 21 Cost results of the SM

Cost Functions	Cost ($)
Waiting Time	57000
Deferring	150000
BIM	770000
Completion	116000
ORs	50000
OR Overtime	90000
PACU Overtime	0

4.8. Results of the RSM using Genetic Algorithm

To obtain the results of the RSM, we assume that some emergency patients arrive at some point in the planning cycle. Thus, the genetic algorithm will provide rescheduling and resequencing of elective patients and scheduling of emergency patients. In addition to the objectives of the first model, minimizing the total waiting time of the emergency patients will be considered. Assuming three emergency patients arrive between 09:00am and 10:00am, where they belong to Cardio-Vascular, General Surgery, and Ophthalmology. Waiting costs of Cardio-Vascular, General Surgery, and Ophthalmology emergency patients are $15,000/hr, $5,000/hr, and $10,000/hr, respectively. Table 4. 22 shows elective patients with waiting time and arriving emergency patients requesting a type of surgery. Table 4. 23 through Table 4. 27 show the results for OR rescheduling and resequencing of elective patients and scheduling of emergency patients from Day 1 to Day 5.

Table 4. 22 Elective and emergency patients with surgery requests

Surgical Operation	Elective Patients	Waiting Time (days)	Emergency Patients
Cardio-Vascular	1,2,3,4	2	71
Ear-Nose-Throat	5	2	
	6,7,8	3	
	9	4	
	10	1	
	11,12,13,14	2	
General Surgery	15,21,22,23,24,25,31,32,33,34,35	2	73
	16,17,18,26,27,28	3	
	19,29	4	
	20,30	1	
Hand	36,37,38	3	
Neurology	39	4	
	40	1	
	41	2	
OB/GYN	42,43,44,45	2	
	46,47,48	3	
Ophthalmology	49	4	72
	50	1	
	51	2	
Orthopedics	52,53,54,55,56,57,58,59,60,61,62	2	
Podiatry	63,64,65,66	2	
Urology	67,68,69,70	2	

Table 4. 23 Operating room rescheduling for day 1

Day 1					
		ORs			
	Time	**OR1**	**OR2**	**OR3**	**OR4**
Regular Time	**08:00-09:00**	Patient 2	Patient 15	Patient 8	Patient 56
	09:00-10:00	Patient 2	*Patient 15*	*Patient 8*	
	10:00-11:00	Patient 2	*Patient 15*	**Patient 72**	**Patient 71**
	11:00-12:00	Patient 2	**Patient 73**	***Patient 72***	**Patient 71**
	12:00-13:00		***Patient 73***		**Patient 71**
	13:00-14:00	Patient 50	***Patient 73***	Patient 28	**Patient 71**
	14:00-15:00	*Patient 50*		*Patient 28*	
	15:00-16:00		Patient 12	*Patient 28*	Patient 25
Overtime	**16:00-17:00**	Patient 14	*Patient 12*		*Patient 25*
	17:00-18:00	*Patient 14*			*Patient 25*

Table 4. 24 Operating room rescheduling for day 2

Day 2					
		ORs			
	Time	**OR1**	**OR2**	**OR3**	**OR4**
Regular Time	**08:00-09:00**	Patient 21	Patient 6	Patient 58	Patient 3
	09:00-10:00	*Patient 21*	*Patient 6*		Patient 3
	10:00-11:00	*Patient 21*		Patient 68	Patient 3
	11:00-12:00		Patient 39	*Patient 68*	Patient 3
	12:00-13:00	Patient 20	Patient 39		
	13:00-14:00	*Patient 20*	Patient 39	Patient 36	Patient 66
	14:00-15:00	*Patient 20*	Patient 39		Patient 66
	15:00-16:00			Patient 38	
Overtime	**16:00-17:00**	Patient 69			Patient 11
	17:00-18:00	*Patient 69*		Patient 54	*Patient 11*

Table 4. 25 Operating room rescheduling for day 3

Day 3					
		ORs			
	Time	**OR1**	**OR2**	**OR3**	**OR4**
Regular Time	**08:00-09:00**	Patient 42	Patient 57	Patient 16	Patient 46
	09:00-10:00	*Patient 42*		*Patient 16*	Patient 46
	10:00-11:00		Patient 12	*Patient 16*	
	11:00-12:00	Patient 51	*Patient 12*		Patient 19
	12:00-13:00	*Patient 51*		Patient 1	*Patient 19*
	13:00-14:00		Patient 37	Patient 1	*Patient 19*
	14:00-15:00	Patient 24		Patient 1	
	15:00-16:00	*Patient 24*	Patient 53	Patient 1	Patient 7
Overtime	**16:00-17:00**	*Patient 24*			Patient 7
	17:00-18:00		Patient 60		

Table 4. 26 Operating room rescheduling for day 4

Day 4					
		ORs			
	Time	**OR1**	**OR2**	**OR3**	**OR4**
Regular Time	**08:00-09:00**	Patient 9	Patient 61	Patient 17	Patient 47
	09:00-10:00	*Patient 9*		*Patient 17*	Patient 47
	10:00-11:00		Patient 13	*Patient 17*	
	11:00-12:00	Patient 64	*Patient 13*		Patient 22
	12:00-13:00	Patient 64		Patient 41	*Patient 22*
	13:00-14:00		Patient 4	Patient 41	*Patient 22*
	14:00-15:00	Patient 25	Patient 4	Patient 41	
	15:00-16:00	*Patient 25*	Patient 4	Patient 41	Patient 43
Overtime	**16:00-17:00**	*Patient 25*	Patient 4		Patient 43
	17:00-18:00				

Table 4. 27 Operating room rescheduling for day 5

Day 5					
		ORs			
	Time	OR1	OR2	OR3	OR4
Regular Time	08:00-09:00	Patient 62	Patient 18	Patient 5	Patient 48
	09:00-10:00		*Patient 18*	*Patient 5*	Patient 48
	10:00-11:00	Patient 29	*Patient 18*		
	11:00-12:00	*Patient 29*		Patient 49	Patient 23
	12:00-13:00	*Patient 29*	Patient 52	*Patient 49*	*Patient 23*
	13:00-14:00				*Patient 23*
	14:00-15:00	Patient 26	Patient 55	Patient 65	
	15:00-16:00	*Patient 26*		Patient 65	Patient 44
Overtime	16:00-17:00	*Patient 26*	Patient 59		Patient 44
	17:00-18:00				

It is to be noted that emergency patients 71 and 72 are scheduled at the time of their arrivals after elective patients 56 and 8 in OR5 and OR4, respectively, while emergency Patient 73 waits for one hour to be scheduled after elective Patient 15 in OR3. Table 4. 28 provides the minimized costs for each objective of the RSM with large-scale data.

Table 4. 28 Cost results of the RSM

Cost Functions	Cost ($)
Elective Waiting Time	55000
Emergency Waiting Time	5000
Deferring	180000
BIM	750000
Completion	119000
ORs	50000
OR Overtime	85000
PACU Overtime	0

4.9. Results of the RSDM using Genetic Algorithm

Some emergency patients arrive at hospital in a serious health condition and may require an operation immediately or within a short time. When elective ORs are not available immediately or within a short time, dedicated rooms are used to handle these emergency patients with serious health conditions. Table 4. 29 shows the elective patients with their waiting time and three types of emergency patients with their surgery requests.

Table 4. 29 Elective and emergency patients with surgery requests for the RSDM

Surgical Operation	Elective Patients	Waiting Time (days)	Emergent Patients	Urgent Patients	Non-urgent Patients
Cardio-Vascular	1,2,3,4	2	71		
Ear-Nose-Throat	5	2			
	6,7,8	3			
	9	4			
	10	1			
	11,12,13,14	2			
General Surgery	15,21,22,23,24,25,31,32,33,34,35	2	72		
	16,17,18,26,27,28	3			
	19,29	4			
	20,30	1			
Hand	36,37,38	3			
Neurology	39	4		73	
	40	1			
	41	2			
OB/GYN	42,43,44,45	2			
	46,47,48	3			
Ophthalmology	49	4			
	50	1			
	51	2			
Orthopedics	52,53,54,55,56,57,58,59,60,61,62	2			74
Podiatry	63,64,65,66	2			
Urology	67,68,69,70	2			

In addition to RSM, RSDM is about minimizing the transfer cost of emergency patients and the cost of using dedicated rooms. If there is no available capacity for arriving emergency patients, they need to be transferred to nearby hospitals and RSDM minimizes this cost. If there is available capacity in elective rooms when emergency patients arrive, they will be taken to those rooms, otherwise they go to the dedicated rooms. RSDM also minimizes the cost of using dedicated rooms. We assume that four emergency patients, two emergent, one urgent, and one non-urgent, arrive between 10:00am and 11:00am or at time 4.

Table 4. 30 shows the results of RSDM. Since Patients 71 and 72 are emergent, they are scheduled immediately in DR1 and DR2, respectively. Patient 73 is an urgent patient and scheduled in elective OR2 after Patient 15. Patient 74 is a non-urgent patient and scheduled in elective OR1 after Patient 2. Here, only the rescheduling for Day 1 is shown since emergency patients arrive on this day. Table 4. 31 shows the cost results of the RSDM.

Table 4. 30 Operating room rescheduling with dedicated rooms for day 1

Day 1							
		Elective ORs				**Dedicated Rooms**	
	Time	**OR1**	**OR2**	**OR3**	**OR4**	**DR1**	**DR2**
Regular Time	**08:00-09:00**	Patient 2	Patient 15	Patient 8	Patient 56		
	09:00-10:00	Patient 2	*Patient 15*	*Patient 8*			
	10:00-11:00	Patient 2	*Patient 15*		Patient 10		
	11:00-12:00	Patient 2		Patient 63	*Patient 10*	**Patient 71**	**Patient 72**
	12:00-13:00		**Patient 73**	Patient 63		**Patient 71**	***Patient 72***
	13:00-14:00	**Patient 74**	**Patient 73**		Patient 38	**Patient 71**	***Patient 72***
	14:00-15:00	**Patient 74**	**Patient 73**	Patient 24		**Patient 71**	
	15:00-16:00		**Patient 73**	*Patient 24*	Patient 68		
Overtime	**16:00-17:00**	Patient 7		*Patient 24*	*Patient 68*		
	17:00-18:00	*Patient 7*					

Table 4. 31 Cost results of RSDM

Cost Functions	**Cost ($)**
Elective Waiting Time	57000
Emergency Waiting Time	7000
Deferring	150000
Transfer	0
BIM	770000
Completion	119000
ORs	50000
DRs	35000
OR Overtime	85000
PACU Overtime	0

4.10. A Comparison Between Lingo and GA

In this section of the study, a comparison between Lingo and GA is provided for the small-scale data since Lingo is unable to solve the large-scale data. The same small-scale data as explained in Chapter 4.1 are used for the GA. Table 4. 32 and Table 4. 33 show the results of SM using GA.

Table 4. 32 Results of the SM using GA

		Regular Time								**Overtime**	
		1	2	3	4	5	6	7	8	**1**	**2**
	ORs	08:00-09:00	09:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	13:00-14:00	14:00-15:00	15:00-16:00	**16:00-17:00**	**17:00-18:00**
Day 1	**OR1**	Patient 1	Patient 1	Patient 1	Patient 1		Patient 3	Patient 3	*Patient 3*		
	OR2	Patient 4	Patient 4	Patient 4		Patient 6	Patient 6	*Patient 6*		Patient 8	
	OR3	Patient 2	Patient 2		Patient 5	Patient 5		Patient 7	Patient 7	Patient 7	Patient 7

BIM BIM BIM BIM BIM BIM BIM BIM

Table 4. 33 Cost results of the SM using GA

Cost Functions	Cost($)
Waiting Time	4800
Deferring	0
BIM	0
Completion	30000
ORs	7500
OR Overtime	2000
PACU Overtime	0

As seen in Table 4. 32 and Table 4. 33, results of the SM using Lingo and GA are same except the sequencing of the patients in the ORs and the location of the BIMs. Then, the RSM is solved using GA and Table 4. 34 and Table 4. 35 show the results.

Table 4. 34 Results of the RSM using GA

		Regular Time								Overtime	
		1	2	3	4	5	6	7	8	**1**	**2**
	ORs	08:00-09:00	09:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	13:00-14:00	14:00-15:00	15:00-16:00	**16:00-17:00**	**17:00-18:00**
Day 1	**OR1**	Patient 1	Patient 1	Patient 1	Patient 1		Patient 3	Patient 3	*Patient 3*		Patient 8
	OR2	Patient 4	Patient 4	Patient 4	**Patient 9**	**Patient 9**		Patient 6	Patient 6	*Patient 6*	
	OR3	Patient 2	Patient 2	Patient 2		**Patient 10**	**Patient 10**	**Patient 10**		Patient 5	Patient 5

BIM BIM BIM BIM

Table 4. 35 Cost results of the RSM using GA

Cost Functions	**Cost($)**
Elective Waiting Time	4200
Emergency Waiting Time	25000
Deferring	15000
BIM	5000
Completion	32000
ORs	7500
OR Overtime	3500
PACU Overtime	0

Based on the results of the RSM using GA, waiting time of emergency patients is more but OR overtime is less than Lingo results. Finally, the RSDM is solved using GA and Table 4. 36 and Table 4. 37 show the results.

Table 4. 36 Results of the RSDM using GA

		Regular Time								Overtime	
		1	2	3	4	5	6	7	8	**1**	**2**
	ORs	08:00-09:00	09:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	13:00-14:00	14:00-15:00	15:00-16:00	**16:00-17:00**	**17:00-18:00**
Day 1	**OR1**	Patient 1	Patient 1	Patient 1	Patient 1		**Patient 12**		Patient 3	Patient 3	*Patient 3*
	OR2	Patient 4	Patient 4	Patient 4		**Patient 10**	**Patient 10**	**Patient 10**		Patient 8	
	OR3	Patient 2	Patient 2	**Patient 11**	**Patient 11**	**Patient 11**		Patient 7	Patient 7	Patient 7	Patient 7
	DR			**Patient 9**	**Patient 9**						

BIM BIM BIM BIM BIM

Table 4. 37 Cost results of the RSDM using GA

Cost Functions	Cost($)
Elective Waiting Time	3600
Emergency Waiting Time	12000
Deferring	30000
Transfer	0
BIM	5000
Completion	37000
ORs	7500
DRs	10000
OR Overtime	4500
PACU Overtime	0

Table 4. 36 and Table 4. 37 show that emergency patients' waiting time, overtime, and costs are increased using GA.

Solving the three MILP models with the small-scale data using Lingo gives better results, in comparison with GA since Lingo guarantees an optimal solution, but GA does not. However, there is a need for GA for the large-scale data since Lingo is not able to provide a feasible or optimal solution in a reasonable time.

CHAPTER 5: CONCLUSION and FUTURE WORK

This study focuses on OR planning and scheduling problems and has three levels, strategic level, tactical level, and operational offline/online level. This study considers operational online level. In this level, elective patients are scheduled and sequenced in ORs. Emergency patients arrive unexpectedly and need to be scheduled as soon as possible in order to improve the total waiting time. In addition, when emergency patients arrive, they disrupt the current schedule, so rescheduling of elective patients and scheduling of emergency patients is required.

In this study, Mixed Integer Linear Programming (MILP) models are developed to handle the OR scheduling and rescheduling problem. The SM considers scheduling and sequencing of elective patients in the ORs by minimizing cost of ORs and downstream units, and improving the total waiting time of elective and emergency patients. When patients finish their surgeries in the ORs, they are transferred to downstream recovery units, such as PACU or SICU, before their discharge. If there is no available capacity in PACU or SICU, patients stay in ORs after their surgeries until there is available capacity in the recovery units. This will cause a delay in scheduling patients in ORs since non-transferred patients will be using OR resources.

The MILP models developed in this study consider minimizing the cost in recovery units since ORs and recovery departments are interconnected units. Cost of overtime in ORs, cost of opening ORs, and cost of postponing patients to next planning cycle are other cost factors minimized by the MILP models. Even though the SM developed in this study does not schedule emergency patients, it minimizes the total waiting time of emergency patients by maximizing the number of BIMs in the elective patient schedule. When the number of BIMs are maximized and emergency patients arrive unexpectedly, the RSM developed in this study schedules and sequences emergency patients and reschedules elective patients considering all the cost and waiting factors.

In addition, the RSM minimizes the total waiting time of emergency patients to make sure they are operated on as soon as possible. Some emergency patients may require an operation immediately or within a short time. Scheduling these types of emergency patients in dedicated rooms may be needed. The RSDM developed in this study considers having dedicated rooms in addition to elective rooms in case some emergency patients may require an immediate operation. For this, the RSDM reschedules and reassigns elective patients and schedules and sequences emergency patients by taking dedicated rooms into consideration. Cost of using dedicated rooms and cost of transfer of patients to nearby hospitals if there is no available capacity are additional cost factors considered by the RSDM.

To find the global optimum solution for these three MILP models, optimization software package Lingo 18 is used. The planning cycle for the short-term planning ranges from a few days to a few weeks. It should be noted that Lingo 18 is able to find a global optimum solution for a small version of the short-term planning cycle, i.e. 1-day and eight patients. However, Lingo 18 and other optimization software packages are not able to solve these MILP models for a regular planning cycle of 1-week or more with at least 70 patients in a reasonable time. Therefore, a metaheuristic algorithm was applied to find a proper feasible solution for a larger version OR scheduling, i.e. 5-day and 70 patients.

There are two disruption sources, namely having shorter or longer durations for SOs and arrival of the emergency patients. Both are considered in this study. Future work includes considering another disruption source, such as no show-up by patients, for rescheduling the MILP model. When patients complete their SO in ORs, they are transferred to PACU to recover. However, some patients require a higher-than-normal level of care and they need to be transferred to SICU from ORs. The current MILP models only consider patients transferring to PACU. The SICU is left for future study.

REFERENCES

1. Colwill, J. M., Cultice, J. M., & Kruse, R. L. (2008). Will generalist physician supply meet demands of an increasing and aging population? *Health Affairs*, *27*(3), w232-w241.
2. Centers for Medicare and Medicaid Services. (2012). National health expenditure data; last accessed December 25, 2018; available at *http://www.cms.hhs.gov/NationalhealthexpendData/downloads/tables.pdf*.
3. Healthcare expenditure statistics; last accessed December 25, 2018; available at https://ec.europa.eu/eurostat/statistics-explained/index.php/Healthcare_expenditure_statistics
4. Lawrence, D. (2005). Bridging the quality chasm. *Building a better delivery system*.
5. Paz, J. (2010). *Process Improvement for the Department of Surgery Sierra Vista Medical Center* (Doctoral dissertation, California Polytechnic State University San Luis Obispo).
6. Andel, C., Davidow, S. L., Hollander, M., & Moreno, D. A. (2012). The economics of health care quality and medical errors. *Journal of health care finance*, *39*(1), 39.
7. Kohn, L. T., Corrigan, J. M., & Donaldson, M. S. (2000). To err is human: building a safer health system. A report of the Committee on Quality of Health Care in America, Institute of Medicine.
8. Agdestein, S. D. (2012). Operating Room Scheduling Problem: Considering the uncertain arrivals of the emergency patients and the capacity limits of the pre-op and post-op facilities.
9. Lamiri, M., Xie, X., & Zhang, S. (2008). Column generation approach to operating theater planning with elective and emergency patients. *IIE Transactions*, *40*(9), 838-852.
10. Anderson, D. R. (2013). The impact of resource management on hospital efficiency and quality of care.

11. Arnaout, J. P. M., & Kulbashian, S. (2008, December). Maximizing the utilization of operating rooms with stochastic times using simulation. In *Proceedings of the 40th conference on winter simulation* (pp. 1617-1623). Winter Simulation Conference.

12. Carter, M. (2002). Diagnosis: mismanagement of resources. *OR MS TODAY*, *29*(2), 26-33.

1. Davila, M. P. (2013). A Methodology for Scheduling Operating Rooms Under Uncertainty (Doctoral dissertation, University of South Florida).

2. Erdogan SA, Denton BT (2010). Wiley Encyclopedia of Operation Research & Management Science.

3. Everett J (2002) A decision support simulation model for the management of an elective surgery waiting system. Health Care Manag Sci 5:89–95.

4. Fei, H., Meskens, N., & Chu, C. (2010). A planning and scheduling problem for an operating theatre using an open scheduling strategy. *Computers & Industrial Engineering*, *58*(2), 221-230.

5. Globerman, S., Esmail, N., Day, B., & Henderson, D. R. (2013). Reducing Wait Times for Health Care: What Canada Can Learn from Theory and International Experience. *Fraser Institute, October*.

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

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

8. van der Lans, M., Hans, E. W., Hurink, J. L., Wullink, G., van Houdenhoven, M., & Kazemier, G. (2006). Anticipating urgent surgery in operating room departments.

9. Van Essen, J. T., Hans, E. W., Hurink, J. L., & Oversberg, A. (2012). Minimizing the waiting time for emergency surgery. *Operations Research for Health Care*, *1*(2), 34-44.

10. McLaughlin, M. M. (2012). *A MODEL TO EVALUATE EFFICIENCY IN OPERATING ROOM PROCESSES* (Doctoral dissertation, The University of Michigan).

11. Healthcare Financial Management Association. (2003). Achieving operating room efficiency through process integration. *Healthcare financial management: journal of the Healthcare Financial Management Association*, *57*(3), suppl-1.

12. Abedini, A., Li, W., & Ye, H. (2017). An optimization model for operating room scheduling to reduce blocking across the perioperative process. *Procedia Manufacturing*, *10*, 60-70.

13. Batun, S. (2011). *Scheduling multiple operating rooms under uncertainty* (Doctoral dissertation, University of Pittsburgh).

14. Cardoen, B., Demeulemeester, E., & Beliën, J. (2006). Optimizing a multiple objective surgical case-scheduling problem.

15. Abdelrasol, Z., Harraz, N., & Eltawil, A. (2014). Operating room scheduling problems: A survey and a proposed solution framework. In *Transactions on Engineering Technologies* (pp. 717-731). Springer, Dordrecht.

16. Guerriero, F., & Guido, R. (2011). Operational research in the management of the operating theatre: a survey. *Health care management science*, *14*(1), 89-114.

17. Erdem, E. (2013). *Optimization models for scheduling and rescheduling elective surgery patients under the constraint of downstream units* (Doctoral dissertation, North Dakota State University).

18. Addis, B., Carello, G., Grosso, A., & Tànfani, E. (2016). Operating room scheduling and rescheduling: a rolling horizon approach. *Flexible Services and Manufacturing Journal*, *28*(1-2), 206-232.

19. Rahimi, S. A., Jamshidi, A., Ait-kadi, D., & Bartolome, A. R. (2014, January). Applied Methods in Prioritization of Patients in Surgery Waiting Lists. In *IIE Annual Conference. Proceedings* (p. 1857). Institute of Industrial Engineers-Publisher.

20. Roland, B., Di Martinelly, C., & Riane, F. (2006, October). Operating theatre optimization: a resource-constrained based solving approach. In *Service Systems and Service Management, 2006 International Conference on* (Vol. 1, pp. 443-448). IEEE.

21. Chaabane, S., Meskens, N., Guinet, A., & Laurent, M. (2008). Comparison of two methods of operating theatre planning: application in Belgian hospital.*Journal of Systems Science and Systems Engineering*, *17*(2), 171-186.

22. Denton, B. T., Rahman, A. S., Nelson, H., & Bailey, A. C. (2006, December). Simulation of a multiple operating room surgical suite. In *Proceedings of the 38th conference on Winter simulation* (pp. 414-424). Winter Simulation Conference.

23. Denton, B., Viapiano, J., & Vogl, A. (2007). Optimization of surgery sequencing and scheduling decisions under uncertainty. *Health care management science,10*(1), 13-24.

24. Dexter, F., Macario, A., Traub, R. D., Hopwood, M., & Lubarsky, D. A. (1999). An operating room scheduling strategy to maximize the use of operating room block time:

computer simulation of patient scheduling and survey of patients' preferences for surgical waiting time. *Anesthesia & Analgesia*, *89*(1), 7-20.

25. Li, Q. (2010). *Multi-objective Operating Room Planning and Scheduling* (Doctoral dissertation, Arizona State University).

26. Tyler, D. C., Pasquariello, C. A., & Chen, C. H. (2003). Determining optimum operating room utilization. Anesthesia & Analgesia, 96(4), 1114-1121.

27. Zhang, B., Murali, P., Dessouky, M. M., & Belson, D. (2009). A mixed integer programming approach for allocating operating room capacity. *Journal of the Operational Research Society*, *60*(5), 663-673.

28. Jeang, A., & Chiang, A. J. (2012). Economic and quality scheduling for effective utilization of operating rooms. *Journal of medical systems*, *36*(3), 1205-1222.

29. Testi, A., & Tànfani, E. (2009). Tactical and operational decisions for operating room planning: Efficiency and welfare implications. *Health Care Management Science*, *12*(4), 363-373.

30. Jebali, A., Alouane, A. B. H., & Ladet, P. (2006). Operating rooms scheduling. *International Journal of Production Economics*, *99*(1), 52-62.

31. Persson, M., & Persson, J. A. (2009). Health economic modeling to support surgery management at a Swedish hospital. *Omega*, *37*(4), 853-863.

32. Tànfani, E., & Testi, A. (2010). A pre-assignment heuristic algorithm for the Master Surgical Schedule Problem (MSSP). *Annals of Operations Research*, *178*(1), 105-119.

33. Ozkarahan, I. (2000). Allocation of surgeries to operating rooms by goal programing. *Journal of Medical Systems*, *24*(6), 339-378.

34. Wullink, G., Van Houdenhoven, M., Hans, E. W., van Oostrum, J. M., van der Lans, M., & Kazemier, G. (2007). Closing emergency operating rooms improves efficiency. *Journal of Medical Systems*, *31*(6), 543-546.

35. Gul, S., Denton, B. T., & Fowler, J. W. (2012). A multi-stage stochastic integer programming model for surgery planning.

36. Heng, M., & Wright, J. G. (2013). Dedicated operating room for emergency surgery improves access and efficiency. *Canadian Journal of Surgery*, *56*(3), 167.

37. Min, D., & Yih, Y. (2010). Scheduling elective surgery under uncertainty and downstream capacity constraints. *European Journal of Operational Research*, *206*(3), 642-652.

38. Lamiri, M., Dreo, J., & Xie, X. (2007, September). Operating room planning with random surgery times. In *Automation Science and Engineering, 2007. CASE 2007. IEEE International Conference on* (pp. 521-526). IEEE.

39. Fei, H., Meskens, N., & Chu, C. (2006, October). An operating theatre planning and scheduling problem in the case of a" block scheduling" strategy. In *Service Systems and Service Management, 2006 International Conference on* (Vol. 1, pp. 422-428). IEEE.

40. Fei, H., Chu, C., & Meskens, N. (2009). Solving a tactical operating room planning problem by a column-generation-based heuristic procedure with four criteria. *Annals of operations research*, *166*(1), 91-108.

41. Dexter, F., Macario, A., & Traub, R. D. (1999). Which algorithm for scheduling add-on elective cases maximizes operating room utilization? Use of bin packing algorithms and fuzzy constraints in operating room management. *Anesthesiology*, *91*(5), 1491-1500.

42. Hans, E., Wullink, G., Van Houdenhoven, M., & Kazemier, G. (2008). Robust surgery loading. *European Journal of Operational Research*, *185*(3), 1038-1050.

43. Vali Siar, M. M., Gholami, S., & Ramezanian, R. (2017). Multi-period and multi-resource operating room scheduling and rescheduling using a rolling horizon approach: a case study. *Journal of Industrial and Systems Engineering*, *10*, 97-115.

44. van Essen, J. T., Hurink, J. L., Hartholt, W., & van den Akker, B. J. (2012). Operating room rescheduling. Working paper, Beta Research School for Operations Management and Logistics, University of Twente.

45. Heydari, M., & Soudi, A. (2016). Predictive/reactive planning and scheduling of a surgical suite with emergency patient arrival. *Journal of medical systems*, *40*(1), 30.

46. Stuart, K., & Kozan, E. (2012). Reactive scheduling model for the operating theatre. *Flexible Services and Manufacturing Journal*, *24*(4), 400-421.

47. Zhong, L., Luo, S., Wu, L., Xu, L., Yang, J., & Tang, G. (2014). A two-stage approach for surgery scheduling. *Journal of Combinatorial Optimization*, *27*(3), 545-556.

48. Wang, S., Su, H., & Wan, G. (2015). Resource-constrained machine scheduling with machine eligibility restriction and its applications to surgical operations scheduling. *Journal of Combinatorial Optimization*, *30*(4), 982-995.

49. Pham, D. N., & Klinkert, A. (2008). Surgical case scheduling as a generalized job shop scheduling problem. *European Journal of Operational Research*, *185*(3), 1011-1025.

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

APPENDIX A. GENETIC ALGORITHM MATLAB CODE FOR THE OPERATING ROOM SCHEDULING

```
clc;
tic

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Begin of Reading data
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

fid = fopen('Operating-Room-Scheduling-Data_modified.txt');
% fid = fopen('Operating Room Scheduling Data-Large-Scale-modified2.txt');

fgetl(fid); fgetl(fid);

Total_number_of_patients = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % Total number
of patients waiting for a surgery
fgetl(fid);

Total_time = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % Total time (hours) in planning
cycle
fgetl(fid);

Regular_time = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % Regular time (hours) in
each Operating Room
fgetl(fid);

Overtime_time = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % Overtime (hours) in each
Operating Room
fgetl(fid);

Number_of_days = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % Number of days in
planning cycle
fgetl(fid);

Number_of_surgery_type = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % Number of
surgery type
fgetl(fid);

Total_number_of_Operating_Rooms = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); %
Total number of Operating Rooms
fgetl(fid);

Turnover_time = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % Turnover (clean) time
between consecutive surgeries
fgetl(fid);

total_number_of_regular = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % total number of
regular working hours for each operating room
```

fgetl(fid);

cost_of_deferring = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % cost of deferring (not scheduling) patients
fgetl(fid);

cost_of_last_surgery = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % cost of last surgery completion time in operating rooms
fgetl(fid);

Cost_of_overtime_per_hour = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % Cost of overtime per hour in operating rooms
fgetl(fid);

fixed_cost_of_opening = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % fixed cost of opening (running) an operating room
fgetl(fid);

Priority_level = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % Priority level for patients, the bigger the number meand the higher the priority level
fgetl(fid);

waiting_time = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % waiting time (days) for patients to have a surgery
fgetl(fid);

cost_of_waiting_per_day = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % cost of waiting per day for patients to have a surgery
fgetl(fid);

Operation_time = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % Operation time (hours) for each surgery type
fgetl(fid);

Recovery_stay_time = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % Recovery stay time (hours) for surgery types
fgetl(fid);

Current_number_of_beds = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % Current number of beds in the recovery area
fgetl(fid);

extra_beds = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % extra beds that can be used for overtime in the recovery are
fgetl(fid);

cost_of_using_extra_beds = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % cost of using extra beds in the recovery area

```
fgetl(fid);

penalty_cost = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match')); % Penalty cost of having same
completion (finish) time for surgeries
fgetl(fid); fgetl(fid); fgetl(fid);

s = [];
for i = 1:Total_number_of_patients
   c = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match'));
   s = [s; c];
end

fgetl(fid); fgetl(fid); fgetl(fid); fgetl(fid); fgetl(fid); fgetl(fid);
Team_availability = [];
for i = 1:Number_of_surgery_type
   c = str2double(regexp(fgetl(fid), '\d+\.?\d*', 'match'));
   Team_availability = [Team_availability; c];
end

fclose(fid)

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% End of Reading data
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Starting the algorithm
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Max_recovery = Current_number_of_beds + extra_beds;

list_patients_Priority = [];
Priority_level_c = Priority_level;

for i = 1:Total_number_of_patients
   list_patients_Priority = [list_patients_Priority; i];
end
for i = 1:Total_number_of_patients
   for j = (i + 1):Total_number_of_patients
      if (Priority_level_c(i) < Priority_level_c(j))
         cop = Priority_level_c(i);
         Priority_level_c(i) = Priority_level_c(j);
         Priority_level_c(j) = cop;

         cop = list_patients_Priority(i);
         list_patients_Priority(i) = list_patients_Priority(j);
         list_patients_Priority(j) = cop;
      end
   end
end
```

```
Operation_time_patients = [];
for i = 1:Total_number_of_patients
   for j = 1:Number_of_surgery_type
      if (s(i, j) == 1)
         Operation_time_patients = [Operation_time_patients j];
         break;
      end
   end
end

Population = [];
Fitness = [];
Sbest = {};
Best_cost = 100000000000000;
nbr_population = 75;
% Generate initial population

for nbr_p = 1:nbr_population

   Sol = [];
   Sol_patients = [];
   Sol_Size = [];
   patients = [];

   for i = 1:Number_of_days
      for j = 1:Total_number_of_Operating_Rooms
         Sol_patients = [Sol_patients; 0];
         Sol(i, j) = size(Sol_patients, 1);
         Sol_Size(i, j) = 0;
      end
   end

   % Add patients into list_pat
   list_patients = [];

   for i = 1:Total_number_of_patients
      list_patients = [list_patients; list_patients_Priority(i)];
      Patient_data_structure = [0 0 0 0]; % nbr of patient ;Type of surgery;Duration of
surgery;Start_hour;End_hour;
      patients = [patients; Patient_data_structure];
   end

   Recovery_room = [];
   for i = 1:Number_of_days
      for j = 1:Total_time
         Recovery_room(i, j) = 0;
      end
```

```
end

Team_current = [];
for i = 1:Number_of_surgery_type
   for j = 1:(Total_time * Number_of_days)
      Team_current(i, j) = 0;
   end
end

Room_availability = [];
for i = 1:Total_number_of_Operating_Rooms
   for j = 1:(Total_time * Number_of_days)
      Room_availability(i, j) = 0;
   end
end

for i = 1:size(list_patients, 1)
   P = list_patients(i);
   entry = 0;
   Can = [];
   Cane_Size = 0;
   for j = 1:Number_of_days
      entry2 = 0;
      for h = 1:Total_number_of_Operating_Rooms
         for t = 1:Total_time
            last = Sol(j, h);
            str = 0;
            str1 = 0;
            indM = - 1;
            if (Sol_Size(j, h) > 0)

               testMin = t - 1;
               for mm = 1:Sol_Size(j, h)
                  if (testMin > patients(Sol_patients(last, mm), 3))
                     indM = mm;
                     break;
                  end
               end
               if (indM > - 1)
                  str1 = Turnover_time;
               end
               str = Turnover_time;
            end

            EndTime = (t - 1) + str + str1 + Operation_time(Operation_time_patients(P)); %E +
str + Operation_time(Operation_time_patients(P));

            if (EndTime > Total_time)
```

```
            EndTime = Total_time;
          end

          test_of_avaibalility_of_room = true;
          for ip = t:EndTime
            % fprintf('Operating Rooms %d   %d   %d  %d:\n',j,
Operation_time(Operation_time_patients(P)),ip,EndTime);
            if ((Room_availability(h, ip + (Total_time * (j - 1)))) > 0)
              test_of_avaibalility_of_room = false;
            end
          end

          % test of avaibalility of team
          test_of_avaibalility_of_team = true;

          for ip = t:EndTime
            % fprintf('Operating Rooms %d   %d   %d  %d:\n',j,
Operation_time(Operation_time_patients(P)),ip,EndTime);
            if ((Team_current(Operation_time_patients(P), ip + (Total_time * (j - 1))) + 1) >
Team_availability(Operation_time_patients(P), ip + (Total_time * (j - 1))))
              test_of_avaibalility_of_team = false;
            end
          end

          if (test_of_avaibalility_of_team == true && test_of_avaibalility_of_room == true
&& Recovery_room(j, EndTime) < Max_recovery && ((t - 1 + str +
Operation_time(Operation_time_patients(P))) <= Total_time))
            Start_hour = t - 1 + str; %E + str;
            if (indM == - 1)
              Start_hour = t - 1;
            end

            End_hour = Start_hour + Operation_time(Operation_time_patients(P));
            Can1 = [j h last Start_hour End_hour t];
            Can = [Can; Can1];
            Cane_Size = Cane_Size + 1;

            break; %go to the next room
          end
        end
      end
    end

    if (Cane_Size > 0)

      start_min = Can(1, 4);
      rand = 1; %randi([1 Cane_Size]);
      for gh = 2:Cane_Size
```

```
            if (Can(gh, 4) < start_min)
                start_min = Can(gh, 4);
                rand = gh;
            end
        end

        Sol_Size(Can(rand, 1), Can(rand, 2)) = Sol_Size(Can(rand, 1), Can(rand, 2)) + 1;
        Sol_patients(Can(rand, 3), Sol_Size(Can(rand, 1), Can(rand, 2))) = P;
        patients(Sol_patients(Can(rand, 3), Sol_Size(Can(rand, 1), Can(rand, 2))), 1) = P;
        patients(Sol_patients(Can(rand, 3), Sol_Size(Can(rand, 1), Can(rand, 2))), 2) =
Operation_time(Operation_time_patients(P));
        patients(Sol_patients(Can(rand, 3), Sol_Size(Can(rand, 1), Can(rand, 2))), 3) = Can(rand,
4);
        patients(Sol_patients(Can(rand, 3), Sol_Size(Can(rand, 1), Can(rand, 2))), 4) = Can(rand,
5);

        Recovery_room(Can(rand, 1), Can(rand, 5)) = Recovery_room(Can(rand, 1), Can(rand,
5)) + 1;

        for ip = (Can(rand, 4) + 1):Can(rand, 5)
            Team_current(Operation_time_patients(P), (ip + (Total_time * (Can(rand, 1) - 1)))) =
Team_current(Operation_time_patients(P), (ip + (Total_time * (Can(rand, 1) - 1)))) + 1;
        end

        for ip = (Can(rand, 4) + 1):Can(rand, 5)
            Room_availability(Can(rand, 2), (ip + (Total_time * (Can(rand, 1) - 1)))) =
Room_availability(Can(rand, 2), (ip + (Total_time * (Can(rand, 1) - 1)))) + 1;
        end
    end
end

% Removed all patients which are inserted into Sol from list_pat;
total_patient_reste = Total_number_of_patients;
for i = 1:Total_number_of_patients
    for j = 1:size(list_patients, 1)
        if (patients(i, 1) == list_patients(j) && patients(i, 1) ~= 0 && patients(i, 2) ~= 0 &&
patients(i, 4) ~= 0)
            list_patients(j) = [];
            total_patient_reste = total_patient_reste - 1;
            break;
        end
    end
end

%fprintf('\nTotal Cost: %f\n', Total_cost);
S1 = Transformation_1(Sol_Size, Sol, Sol_patients, patients, Total_number_of_patients,
Number_of_days, Total_number_of_Operating_Rooms);
```

```
    S = {S1 Sol_Size Sol Sol_patients patients list_patients Recovery_room total_patient_reste};
% {Sol_Size  Sol  Sol_patients  patients  list_patients Recovery_room total_patient_reste};
    Total_cost = Objective_function(S, Total_number_of_patients, Number_of_days,
Total_number_of_Operating_Rooms, cost_of_waiting_per_day, cost_of_deferring,
cost_of_last_surgery, fixed_cost_of_opening, Cost_of_overtime_per_hour, Regular_time,
cost_of_using_extra_beds, Current_number_of_beds, Total_time, penalty_cost, waiting_time);

    %disp(S1)
    Population = [Population; S];
    Fitness = [Fitness; Total_cost];

    if (Total_cost < Best_cost)
        Sbest = S;
        Best_cost = Total_cost;
    end

    %Show(S, Total_cost, Number_of_days, Total_number_of_Operating_Rooms,
Operation_time, Operation_time_patients);

end

fprintf('\nTotal Cost: %f\n', Best_cost);

Max_nbr_iteration = 1000;
ite = 0;

while (ite < Max_nbr_iteration)
    ite = ite + 1;
    % Crossover
    New_Population = [];
    New_fitness = [];
    for i = 1:(nbr_population / 2)
        % Select randomly two solutions S1 & S2 from Population[]
        rand1 = randi([1 nbr_population]);
        S11 = Population(rand1, :);
        S1 = S11{1, 1};
        rand2 = randi([1 nbr_population]);
        S22 = Population(rand2, :);
        S2 = S22{1, 1};

        % Select randomly two point
        p1 = randi([1 size(S1, 1)]);
        p2 = randi([1 size(S2, 1)]);

        % Let child1 & child2
        Child1 = [];
        Child2 = [];
        for m = 1:Total_number_of_patients
```

```
    Cop = [0 0];
    Child1 = [Child1; Cop];
    Child2 = [Child2; Cop];
end

for j = 1:p1
    Child1(j, 1) = S1(j, 1);
    Child1(j, 2) = S1(j, 2);

    Child2(j, 1) = S2(j, 1);
    Child2(j, 2) = S2(j, 2);
end

for j = p2:size(S1, 1)
    Child1(j, 1) = S1(j, 1);
    Child1(j, 2) = S1(j, 2);

    Child2(j, 1) = S2(j, 1);
    Child2(j, 2) = S2(j, 2);
end
for j = (p1 + 1):p2
    Child2(j, 1) = S1(j, 1);
    Child2(j, 2) = S1(j, 2);

    Child1(j, 1) = S2(j, 1);
    Child1(j, 2) = S2(j, 2);
end

% mutation
r = randi([1 100]);
if (r < 5)
    r1 = randi([1 size(Child1, 1)]);
    r2 = randi([1 size(Child1, 1)]);

    Cop1 = Child1(r1, 1);
    Cop2 = Child1(r1, 2);

    Child1(r1, 1) = Child1(r2, 1);
    Child1(r1, 2) = Child1(r2, 2);

    Child1(r2, 1) = Cop1;
    Child1(r2, 2) = Cop2;

    r1 = randi([1 size(Child2, 1)]);
    r2 = randi([1 size(Child2, 1)]);

    Cop1 = Child2(r1, 1);
    Cop2 = Child2(r1, 2);
```

```
        Child2(r1, 1) = Child2(r2, 1);
        Child2(r1, 2) = Child2(r2, 2);

        Child2(r2, 1) = Cop1;
        Child2(r2, 2) = Cop2;
      end

      C1 = Transformation_2(Child1, list_patients_Priority, Total_number_of_patients,
Number_of_days, Total_number_of_Operating_Rooms, Total_time, Operation_time,
Operation_time_patients, Team_availability, Number_of_surgery_type, Max_recovery,
Turnover_time);
      C2 = Transformation_2(Child2, list_patients_Priority, Total_number_of_patients,
Number_of_days, Total_number_of_Operating_Rooms, Total_time, Operation_time,
Operation_time_patients, Team_availability, Number_of_surgery_type, Max_recovery,
Turnover_time);

      C1_F = Objective_function(C1, Total_number_of_patients, Number_of_days,
Total_number_of_Operating_Rooms, cost_of_waiting_per_day, cost_of_deferring,
cost_of_last_surgery, fixed_cost_of_opening, Cost_of_overtime_per_hour, Regular_time,
cost_of_using_extra_beds, Current_number_of_beds, Total_time, penalty_cost, waiting_time);

      C2_F = Objective_function(C2, Total_number_of_patients, Number_of_days,
Total_number_of_Operating_Rooms, cost_of_waiting_per_day, cost_of_deferring,
cost_of_last_surgery, fixed_cost_of_opening, Cost_of_overtime_per_hour, Regular_time,
cost_of_using_extra_beds, Current_number_of_beds, Total_time, penalty_cost, waiting_time);
      %disp("C1");
      %Show(C1, C1_F, Number_of_days, Total_number_of_Operating_Rooms,
Operation_time, Operation_time_patients);
      %disp("C2");
      %Show(C2, C2_F, Number_of_days, Total_number_of_Operating_Rooms,
Operation_time, Operation_time_patients);

      New_Population = [New_Population; C1];
      New_fitness = [New_fitness; C1_F];

      New_Population = [New_Population; C2];
      New_fitness = [New_fitness; C2_F];

   end

   Total = [];
   Total_fitness = [];
   for i = 1:size(Population, 1)
      Total = [Total; Population(i, :)];
      Total_fitness = [Total_fitness; Fitness(i)];
   end
   for i = 1:size(New_Population, 1)
```

```
        Total = [Total; New_Population(i, :)];
        Total_fitness = [Total_fitness; New_fitness(i)];
    end

    for i = 1:nbr_population
        ind = 0;
        min = 100000000000000;
        for j = 1:size(Total, 1)
            if (min > Total_fitness(j))
                min = Total_fitness(j);
                ind = j;
            end
        end
        Population(i, :) = Total(ind, :);
        Fitness(i) = Total_fitness(ind);
        Total(ind, :) = [];
        Total_fitness(ind) = [];
    end

    Sb = Population(1, :);
    Sb_fitness = Fitness(1);
    %fprintf('\nIteration %d => Cost = %f', ite, Sb_fitness);
    if (Sb_fitness < Best_cost)
        Sbest = Sb;
        Best_cost = Sb_fitness;
        fprintf('\nIteration %d => Cost = %f', ite, Best_cost);
    end

end

% Show(Sbest,Number_of_days,Total_number_of_Operating_Rooms);
fprintf('\nBest Solution\n');
Show(Sbest, Best_cost, Number_of_days, Total_number_of_Operating_Rooms, Operation_time,
Operation_time_patients);
Objective_function_detail(Sbest, Total_number_of_patients, Number_of_days,
Total_number_of_Operating_Rooms, cost_of_waiting_per_day, cost_of_deferring,
cost_of_last_surgery, fixed_cost_of_opening, Cost_of_overtime_per_hour, Regular_time,
cost_of_using_extra_beds, Current_number_of_beds, Total_time, penalty_cost, waiting_time);

function S1 = Transformation_1(Sol_Size, Sol, Sol_patients, patients,
Total_number_of_patients, Number_of_days, Total_number_of_Operating_Rooms)
    S1 = [];
    for i = 1:Total_number_of_patients
        S11 = [0 0];
        S1 = [S1; S11];
    end

    for i = 1:Number_of_days
```

```
        for j = 1:Total_number_of_Operating_Rooms
            for h = 1:Sol_Size(i, j)
                E = patients(Sol_patients(Sol(i, j), h), 1);
                S1(E, 1) = i;
                S1(E, 2) = j;
            end
        end
    end

end

function S = Transformation_2(S1, list_patients_Priority, Total_number_of_patients,
Number_of_days, Total_number_of_Operating_Rooms, Total_time, Operation_time,
Operation_time_patients, Team_availability, Number_of_surgery_type, Max_recovery,
Turnover_time)

    Sol = [];
    Sol_patients = [];
    Sol_Size = [];
    patients = [];
    total_patient_reste = Total_number_of_patients;

    for i = 1:Number_of_days
        for j = 1:Total_number_of_Operating_Rooms
            Sol_patients = [Sol_patients; 0];
            Sol(i, j) = size(Sol_patients, 1);
            Sol_Size(i, j) = 0;
        end
    end

    % Add patients into list_pat
    list_patients = [];

    for i = 1:Total_number_of_patients
        list_patients = [list_patients; list_patients_Priority(i)];
        Patient_data_structure = [0 0 0 0]; % nbr of patient ;Type of surgery;Duration of
surgery;Start_hour;End_hour;
        patients = [patients; Patient_data_structure];
    end

    Recovery_room = [];
    for i = 1:Number_of_days
        for j = 1:Total_time
            Recovery_room(i, j) = 0;
        end
    end

    Team_current = [];
```

```
    for i = 1:Number_of_surgery_type
        for j = 1:(Total_time * Number_of_days)
            Team_current(i, j) = 0;
        end
    end

    Room_availability = [];
    for i = 1:Total_number_of_Operating_Rooms
        for j = 1:(Total_time * Number_of_days)
            Room_availability(i, j) = 0;
        end
    end

    for i = 1:Total_number_of_patients
        P = list_patients(i);
        insert = 0;
        if (S1(P, 1) ~= 0)
            total_patient_reste = total_patient_reste - 1;
            str = 0;
            E = 0;
            if (Sol_Size(S1(P, 1), S1(P, 2)) > 0)
                str = Turnover_time;
                E = patients(Sol_patients(Sol(S1(P, 1), S1(P, 2)), Sol_Size(S1(P, 1), S1(P, 2))), 4);
            end
            EndTime = E + str + Operation_time(Operation_time_patients(P));

            if (Total_time < EndTime)
                EndTime = Total_time;
            end
            test_of_avaibalility_of_room = true;
            for ip = (E + 1):EndTime
                % fprintf('Operating Rooms %d   %d   %d  %d:\n',j,
Operation_time(Operation_time_patients(P)),ip,EndTime);
                if ((Room_availability(S1(P, 2), ip + (Total_time * (S1(P, 1) - 1)))) > 0)
                    test_of_avaibalility_of_room = false;
                end
            end

            % test of avaibalility of team
            test_of_avaibalility_of_team = true;

            for ip = (E + 1):EndTime
                % fprintf('Operating Rooms %d   %d   %d  %d:\n',j,
Operation_time(Operation_time_patients(P)),ip,EndTime);
                if ((Team_current(Operation_time_patients(P), ip + (Total_time * (S1(P, 1) - 1))) + 1)
> Team_availability(Operation_time_patients(P), ip + (Total_time * (S1(P, 1) - 1))))
                    test_of_avaibalility_of_team = false;
                end
```

```
        end
        if (test_of_avaibalility_of_team == true && test_of_avaibalility_of_room == true &&
Recovery_room(S1(P, 1), EndTime) < Max_recovery && ((E + str +
Operation_time(Operation_time_patients(P))) <= Total_time))

            Sol_Size(S1(P, 1), S1(P, 2)) = Sol_Size(S1(P, 1), S1(P, 2)) + 1;
            Sol_patients(Sol(S1(P, 1), S1(P, 2)), Sol_Size(S1(P, 1), S1(P, 2))) = P;
            patients(P, 1) = P;
            patients(P, 2) = Operation_time(Operation_time_patients(P));
            patients(P, 3) = E + str;
            patients(P, 4) = EndTime;

            Recovery_room(S1(P, 1), EndTime) = Recovery_room(S1(P, 1), EndTime) + 1;
            for ip = (E + 1):EndTime
                Team_current(Operation_time_patients(P), ip + (Total_time * (S1(P, 1) - 1))) =
Team_current(Operation_time_patients(P), ip + (Total_time * (S1(P, 1) - 1))) + 1;
            end

            for ip = (E + 1):EndTime
                Room_availability(S1(P, 2), (ip + (Total_time * (S1(P, 1) - 1)))) =
Room_availability(S1(P, 2), (ip + (Total_time * (S1(P, 1) - 1)))) + 1;
            end

            insert = 1;
        end

    end
    if (insert ~= 1)
        Can = [];
        Cane_Size = 0;
        for j = 1:Number_of_days
            for h = 1:Total_number_of_Operating_Rooms
                for t = 1:Total_time
                    last = Sol(j, h);
                    str = 0;
                    str1 = 0;
                    indM = - 1;
                    if (Sol_Size(j, h) > 0)
                        testMin = t - 1;
                        for mm = 1:Sol_Size(j, h)
                            if (testMin > patients(Sol_patients(last, mm), 3))
                                indM = mm;
                                break;
                            end
                        end
                        if (indM > - 1)
                            str1 = Turnover_time;
                        end
```

```
          str = Turnover_time;
        end

        EndTime = (t - 1) + str + str1 + Operation_time(Operation_time_patients(P)); %E
+ str + Operation_time(Operation_time_patients(P));

        if (EndTime > Total_time)
          EndTime = Total_time;
        end

        test_of_avaibalility_of_room = true;
        for ip = t:EndTime
          % fprintf('Operating Rooms %d   %d   %d  %d:\n',j,
Operation_time(Operation_time_patients(P)),ip,EndTime);
          if ((Room_availability(h, ip + (Total_time * (j - 1)))) > 0)
            test_of_avaibalility_of_room = false;
          end
        end

        % test of avaibalility of team
        test_of_avaibalility_of_team = true;

        for ip = t:EndTime
          days = (ip + (Total_time * (j - 1)));
          % fprintf('Operating Rooms %d   %d   %d  %d:\n',j,
Operation_time(Operation_time_patients(P)),ip,EndTime);
          if ((Team_current(Operation_time_patients(P), days) + 1) >
Team_availability(Operation_time_patients(P), days))
            test_of_avaibalility_of_team = false;
          end
        end

        if (test_of_avaibalility_of_team == true && test_of_avaibalility_of_room == true
&& Recovery_room(j, EndTime) < Max_recovery && ((t - 1 + str +
Operation_time(Operation_time_patients(P))) <= Total_time))
          Start_hour = t - 1 + str;

          if (indM == - 1)
            Start_hour = t - 1;
          end

          End_hour = Start_hour + Operation_time(Operation_time_patients(P));

          End_hour = Start_hour + Operation_time(Operation_time_patients(P));
          Can1 = [j h last Start_hour End_hour t];
          Can = [Can; Can1];
          Cane_Size = Cane_Size + 1;
```

```
                    %Sol_Size(j, h) = Sol_Size(j, h) + 1;
                    %Sol_patients(last, Sol_Size(j, h)) = P;
                    %patients(Sol_patients(last, Sol_Size(j, h)), 1) = P;
                    %patients(Sol_patients(last, Sol_Size(j, h)), 2) =
Operation_time(Operation_time_patients(P));
                    %patients(Sol_patients(last, Sol_Size(j, h)), 3) = Start_hour;
                    %patients(Sol_patients(last, Sol_Size(j, h)), 4) = End_hour;

                    %Recovery_room(j, EndTime) = Recovery_room(j, EndTime) + 1;

                    %for ip = (Start_hour + 1):EndTime
                    %Team_current(Operation_time_patients(P), ip + (Total_time * (j - 1))) =
Team_current(Operation_time_patients(P), ip + (Total_time * (j - 1))) + 1;
                    %end
                    %for ip = (Start_hour + 1):EndTime
                    %Room_availability(h, (ip + (Total_time * (j - 1)))) = Room_availability(h, (ip
+ (Total_time * (j - 1)))) + 1;
                    %end

                    break;
                end

            end
        end
    end

    if (Cane_Size > 0)

        start_min = Can(1, 4);
        rand = 1; % randi([1 Cane_Size]);
        for gh = 2:Cane_Size
            if (Can(gh, 4) < start_min)
                start_min = Can(gh, 4);
                rand = gh;
            end
        end

        Sol_Size(Can(rand, 1), Can(rand, 2)) = Sol_Size(Can(rand, 1), Can(rand, 2)) + 1;
        Sol_patients(Can(rand, 3), Sol_Size(Can(rand, 1), Can(rand, 2))) = P;
        patients(Sol_patients(Can(rand, 3), Sol_Size(Can(rand, 1), Can(rand, 2))), 1) = P;
        patients(Sol_patients(Can(rand, 3), Sol_Size(Can(rand, 1), Can(rand, 2))), 2) =
Operation_time(Operation_time_patients(P));
        patients(Sol_patients(Can(rand, 3), Sol_Size(Can(rand, 1), Can(rand, 2))), 3) =
Can(rand, 4);
        patients(Sol_patients(Can(rand, 3), Sol_Size(Can(rand, 1), Can(rand, 2))), 4) =
Can(rand, 5);
```

```
            Recovery_room(Can(rand, 1), Can(rand, 5)) = Recovery_room(Can(rand, 1),
Can(rand, 5)) + 1;

            for ip = (Can(rand, 4) + 1):Can(rand, 5)
               Team_current(Operation_time_patients(P), (ip + (Total_time * (Can(rand, 1) - 1))))
= Team_current(Operation_time_patients(P), (ip + (Total_time * (Can(rand, 1) - 1)))) + 1;
            end

            for ip = (Can(rand, 4) + 1):Can(rand, 5)
               Room_availability(Can(rand, 2), (ip + (Total_time * (Can(rand, 1) - 1)))) =
Room_availability(Can(rand, 2), (ip + (Total_time * (Can(rand, 1) - 1)))) + 1;
            end
         end

      end
   end

   % Removed all patients which are inserted into Sol from list_pat;
   total_patient_reste = Total_number_of_patients;
   for i = 1:Total_number_of_patients
      for j = 1:size(list_patients, 1)
         if (patients(i, 1) == list_patients(j) && patients(i, 1) ~= 0 && patients(i, 2) ~= 0 &&
patients(i, 4) ~= 0)
            list_patients(j) = [];
            total_patient_reste = total_patient_reste - 1;
            break;
         end
      end
   end

   % disp(Sol_patients)
   % disp(total_patient_reste)
   S = {S1 Sol_Size Sol Sol_patients patients list_patients Recovery_room total_patient_reste};
end

function Show(C1, Total_cost, Number_of_days, Total_number_of_Operating_Rooms,
Operation_time, Operation_time_patients)

   Sol_Size = C1{1, 2};
   Sol = C1{1, 3};
   Sol_patients = C1{1, 4};
   patients = C1{1, 5};
   list_patients = C1{1, 6};
   Recovery_room = C1{1, 7};
   total_patient_reste = C1{1, 8};

   fprintf('\nPatient not inserted\n')
```

```
    for j = 1:total_patient_reste
        P = list_patients(j);
        fprintf('    Patient %d: Operation time: %d \n', P,
Operation_time(Operation_time_patients(P)));
    end
    % Show Resultat
    for i = 1:Number_of_days
        fprintf('\nDay %d:\n', i);
        for j = 1:Total_number_of_Operating_Rooms
            fprintf('    Operating Rooms %d:\n', j);

            for h = 1:Sol_Size(i, j)
                min = patients(Sol_patients(Sol(i, j), h), 3);
                index = h;

                for hh = (h + 1):Sol_Size(i, j)
                    if (min > patients(Sol_patients(Sol(i, j), hh), 3))
                        min = patients(Sol_patients(Sol(i, j), hh), 3);
                        index = hh;
                    end
                end

                P = Sol_patients(Sol(i, j), index);
                Sol_patients(Sol(i, j), index) = Sol_patients(Sol(i, j), h);
                Sol_patients(Sol(i, j), h) = P;
                fprintf('        Patient %d: Operation time: %d, Start hour: %d, End hour: %d\n',
patients(P, 1), patients(P, 2), patients(P, 3), patients(P, 4));
            end

        end
    end
    fprintf('\nTotal Cost: %f\n', Total_cost);

end

% %%%%%%%%%%%%%% Objective function
function Total_cost = Objective_function(S, Total_number_of_patients, Number_of_days,
Total_number_of_Operating_Rooms, cost_of_waiting_per_day, cost_of_deferring,
cost_of_last_surgery, fixed_cost_of_opening, Cost_of_overtime_per_hour, Regular_time,
cost_of_using_extra_beds, Current_number_of_beds, Total_time, penalty_cost, waiting_time)

    Sol_Size = S{1, 2};
    Sol = S{1, 3};
    Sol_patients = S{1, 4};
    patients = S{1, 5};
    list_patients = S{1, 6};
    Recovery_room = S{1, 7};
    total_patient_reste = S{1, 8};
```

```
    Wait_cost = 0;
    for i = 1:Total_number_of_patients
        Additional_time = 0;
        for j = 1:Number_of_days
            Additional_time = Additional_time + 1;
            for k = 1:Total_number_of_Operating_Rooms
                for m = 1:Sol_Size(j, k)
                    if (Sol_patients(j, m) == i)
                        %Wait_cost = Wait_cost + cost_of_waiting_per_day(j, i);
                        Wait_cost = Wait_cost + (cost_of_waiting_per_day(1, i) * (waiting_time(1, i) +
Additional_time - 1));
                        break;
                    end
                end
            end
        end
    end

    %fprintf('\nOBJ1: The waiting cost of elective patients: %f\n', Wait_cost);

    Def_cost = 0;
    for i = 1:Total_number_of_patients
        for j = 1:total_patient_reste
            if (list_patients(j) == i)
                Def_cost = Def_cost + cost_of_deferring;
                break;
            end
        end
    end

    % fprintf('\nOBJ2: The cost of deferring elective patients to next planning cycle: %f\n',
Def_cost);

    Rep_cost = 0;
    EndHour = [];
    for i = 1:Total_number_of_patients
        EndHour = [EndHour; - 1];
    end
    for i = 1:Number_of_days
        for j = 1:Total_number_of_Operating_Rooms
            for m = 1:Sol_Size(i, j)
                EndHour(Sol_patients(Sol(i, j), m)) = patients(Sol_patients(Sol(i, j), m), 4);
            end
        end
    end

    for i = 1:Total_number_of_patients
```

```
    for i1 = (i + 1):Total_number_of_patients
      if ((EndHour(i) ~= - 1) && (EndHour(i) == EndHour(i1)))
        Rep_cost = Rep_cost + penalty_cost;
      end
    end
  end
  % fprintf('\nOBJ3: The penalty cost of surgery completion time repeats: %f\n', Rep_cost);

  Cmax_cost = 0;
  for i = 1:Number_of_days
    for j = 1:Total_number_of_Operating_Rooms
      if (Sol_Size(i, j) > 0)
        e = patients(Sol_patients(Sol(i, j), Sol_Size(i, j)), 4);
        Cmax_cost = Cmax_cost + (e * cost_of_last_surgery);
      end
    end
  end
  % fprintf('\nOBJ4: The cost of completion the last surgeries in ORs: %f\n', Cmax_cost);

  RO_cost = 0;
  for j = 1:Number_of_days
    for k = 1:Total_number_of_Operating_Rooms
      if (Sol_Size(j, k) > 0)
        RO_cost = RO_cost + fixed_cost_of_opening;
      end
    end
  end

  % fprintf('\nOBJ5: The cost of opening ORs: %f\n', RO_cost);

  Over_cost = 0;
  for i = 1:Number_of_days
    for j = 1:Total_number_of_Operating_Rooms
      if (Sol_Size(i, j) > 0)
        e = patients(Sol_patients(Sol(i, j), Sol_Size(i, j)), 4);
        Over_cost = Over_cost + (max((e - Regular_time), 0) * Cost_of_overtime_per_hour);
      end
    end
  end

  % fprintf('\nOBJ6: The cost of overtime in ORs: %f\n', Over_cost);

  Over_ra = 0;
  for j = 1:Number_of_days
    for hour = 1:Total_time
      if (Recovery_room(j, hour) > Current_number_of_beds)
        Over_ra = Over_ra + ((Recovery_room(j, hour) - Current_number_of_beds) *
cost_of_using_extra_beds);
```

```
      end
    end
  end

  % fprintf('\nOBJ7: The cost of overtime in PACU: %f\n', Over_ra);

  Total_cost = Wait_cost + Def_cost + Rep_cost + Cmax_cost + RO_cost + Over_cost +
Over_ra;
end

% %%%%%%%%%%%%%% Objective function detail
function Total_cost = Objective_function_detail(S, Total_number_of_patients, Number_of_days,
Total_number_of_Operating_Rooms, cost_of_waiting_per_day, cost_of_deferring,
cost_of_last_surgery, fixed_cost_of_opening, Cost_of_overtime_per_hour, Regular_time,
cost_of_using_extra_beds, Current_number_of_beds, Total_time, penalty_cost, waiting_time)

  Sol_Size = S{1, 2};
  Sol = S{1, 3};
  Sol_patients = S{1, 4};
  patients = S{1, 5};
  list_patients = S{1, 6};
  Recovery_room = S{1, 7};
  total_patient_reste = S{1, 8};

  Wait_cost = 0;
  for i = 1:Total_number_of_patients
    Additional_time = 0;
    for j = 1:Number_of_days
      Additional_time = Additional_time + 1;
      for k = 1:Total_number_of_Operating_Rooms
        for m = 1:Sol_Size(j, k)
          if (Sol_patients(j, m) == i)
            %Wait_cost = Wait_cost + cost_of_waiting_per_day(j, i);
            Wait_cost = Wait_cost + (cost_of_waiting_per_day(1, i) * (waiting_time(1, i) +
Additional_time - 1));
            break;
          end
        end
      end
    end
  end

  fprintf('\nOBJ1: The waiting cost of elective patients: %f\n', Wait_cost);

  Def_cost = 0;
  for i = 1:Total_number_of_patients
    for j = 1:total_patient_reste
      if (list_patients(j) == i)
```

```
            Def_cost = Def_cost + cost_of_deferring;
            break;
        end
    end
end

fprintf('\nOBJ2: The cost of deferring elective patients to next planning cycle: %f\n',
Def_cost);

Rep_cost = 0;
EndHour = [];
for i = 1:Total_number_of_patients
    EndHour = [EndHour; - 1];
end
for i = 1:Number_of_days
    for j = 1:Total_number_of_Operating_Rooms
        for m = 1:Sol_Size(i, j)
            EndHour(Sol_patients(Sol(i, j), m)) = patients(Sol_patients(Sol(i, j), m), 4);
        end
    end
end

for i = 1:Total_number_of_patients
    for i1 = (i + 1):Total_number_of_patients
        if ((EndHour(i) ~= - 1) && (EndHour(i) == EndHour(i1)))
            Rep_cost = Rep_cost + penalty_cost;
        end
    end
end
fprintf('\nOBJ3: The penalty cost of surgery completion time repeats: %f\n', Rep_cost);

Cmax_cost = 0;
for i = 1:Number_of_days
    for j = 1:Total_number_of_Operating_Rooms
        if (Sol_Size(i, j) > 0)
            e = patients(Sol_patients(Sol(i, j), Sol_Size(i, j)), 4);
            Cmax_cost = Cmax_cost + (e * cost_of_last_surgery);
        end
    end
end
fprintf('\nOBJ4: The cost of completion the last surgeries in ORs: %f\n', Cmax_cost);

RO_cost = 0;
for j = 1:Number_of_days
    for k = 1:Total_number_of_Operating_Rooms
        if (Sol_Size(j, k) > 0)
            RO_cost = RO_cost + fixed_cost_of_opening;
        end
```

```
        end
    end

    fprintf('\nOBJ5: The cost of opening ORs: %f\n', RO_cost);

    Over_cost = 0;
    for i = 1:Number_of_days
        for j = 1:Total_number_of_Operating_Rooms
            if (Sol_Size(i, j) > 0)
                e = patients(Sol_patients(Sol(i, j), Sol_Size(i, j)), 4);
                Over_cost = Over_cost + (max((e - Regular_time), 0) * Cost_of_overtime_per_hour);
            end
        end
    end

    fprintf('\nOBJ6: The cost of overtime in ORs: %f\n', Over_cost);

    Over_ra = 0;
    for j = 1:Number_of_days
        for hour = 1:Total_time
            if (Recovery_room(j, hour) > Current_number_of_beds)
                Over_ra = Over_ra + ((Recovery_room(j, hour) - Current_number_of_beds) *
cost_of_using_extra_beds);
            end
        end
    end

    fprintf('\nOBJ7: The cost of overtime in PACU: %f\n', Over_ra);

    Total_cost = Wait_cost + Def_cost + Rep_cost + Cmax_cost + RO_cost + Over_cost +
Over_ra;

    fprintf('\nTotal cost: %f\n', Total_cost);
end

% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% End of the algorithm
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```