Int. J. Production Economics 141 (2013) 646–658

Contents lists available at SciVerse ScienceDirect

# Int. J. Production Economics

journal homepage: www.elsevier.com/locate/ijpe

# Appointment scheduling of outpatient surgical services in a multistage operating room department

Alireza Saremi [a], Payman Jula [b,*], Tarek ElMekkawy [c], G. Gary Wang [d]

[a] Department of Mechanical and Manufacturing Eng., University of Manitoba, Winnipeg, MB, Canada R3T 2N2
[b] Beedie School of Business, Simon Fraser University, 8888 University drive, Vancouver, BC, Canada V5A1S6
[c] Department of Mechanical and Manufacturing Eng., University of Manitoba, Winnipeg, MB, Canada R3T 2N2
[d] Mechatronic Systems Engineering, Simon Fraser University, Surrey, BC, Canada V3T 0A3

ARTICLE INFO

*Article history:*
Received 14 August 2011
Accepted 6 October 2012
Available online 13 October 2012

*Keywords:*
Outpatient surgery scheduling
Simulation-based optimization
Mathematical programming
Tabu search

ABSTRACT

This article addresses the appointment scheduling of outpatient surgeries in a multistage operating room (OR) department with stochastic service times serving multiple patient types. We discuss many challenges, such as the limited availability of multiple resources (e.g., staff, operating rooms, surgeons, and recovery beds), and the compatibility of patient and surgeon types. In addition, availability of surgeons is restricted by time window constraints. Three simulation-based optimization methods have been proposed to minimize the patients' wait time, patients' completion time, and number of surgery cancellations. The first method is simulation-based tabu search (STS). It combines discrete-event simulation and tabu search to schedule surgery cases. The second and third methods are integer programming enhanced tabu search (IPETS) and binary programming enhanced tabu search (BPETS). IPETS and BPETS improve on STS by incorporating integer programming and binary programming models, respectively. This article includes a case study of an OR department in a major Canadian hospital. We further expand the actual data obtained in the case study to cover a wide range of parameters in sets of test problems, and provide analysis on the efficiency and effectiveness of the proposed methods in comparison with several scheduling rules. Finally, comments on the applications of the proposed methods are provided.



## 1. Introduction

Operating room (OR) departments are in a constant battle to use their limited resources in order to serve a maximum number of patients. Appointment scheduling plays an important role in this context, by providing a smooth flow of patients while minimizing the patient waiting time, completion time, and number of cancellations in OR departments. In this research, appointment scheduling refers to the determination of the time at which each patient should arrive at the OR department, and waiting time is the time that a patient spends in the facility waiting to be served. Completion time refers to the time that the last patient leaves the OR department. Case cancellations, simply called cancellations here, refer to scheduled surgery cases that are cancelled due to the lack of time or resources. Scheduling in this environment is a challenging task due to the stochastic service times and constrained resources. Surgery scheduling is not only restricted by the availability of resources, but also constrained by the compatibility requirements (e.g., only a specific surgeon type can serve a patient type). In this study, we focus on outpatient surgeries, in which patients leave the system on the same day after receiving the service.

This research focuses on minimizing the waiting time of patients, and the completion time of OR department while monitoring the cancellation of scheduled cases. As reported by several studies such as Gül et al. (2011), and Klassen and Yoogalingam (2009), in this environment, improving one measure often leads to the deterioration of other criteria. For example, minimizing waiting time may decrease the utilization, or increase the completion time and number of cancellations in the OR department.

Previous studies have applied optimization or simulation methods to schedule surgery cases. Typically, optimization methods use analytical approaches to achieve optimal (or near optimal) solutions. These approaches have difficulty addressing large-complex systems and, therefore, have often focused on elements of the system, or have overly simplified the system. For instance, many optimization methods consider only single stage systems with Exponential or Erlang distributions for service

* Corresponding author. Tel.: +778 782 3564; fax: +778 782 4920.
*E-mail addresses:* umsaremi@cc.umanitoba.ca (A. Saremi), pjula@sfu.ca (P. Jula), tmekkawy@cc.umanitoba.ca (T. ElMekkawy), gary_wang@sfu.ca (G.G. Wang).

0925-5273/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.ijpe.2012.10.004

times. On the other hand, simulation methods are capable of addressing complexities of large systems. Hence, simulation literature has considered detailed multistage systems with constraints on resources, accounting for several environmental factors such as patient priorities, unpunctual patients, and different service time distributions. However, simulation approaches are time-consuming and often do not deliver a competitive optimization strategy. Therefore, a gap exists in the literature questing for efficient and effective methods to address the challenges in outpatient surgery scheduling. The *efficiency* of a method refers to the amount of computation time required by the method to produce meaningful results, while *effectiveness* addresses the quality of solutions generated.

In this study, we integrate the *discrete-event simulation* model, hereafter called simulation model, with metaheuristics, propose three simulation-based optimization methods, and further improve the performance of the proposed methods using mathematical programming (MP). The proposed methods address the problem of appointment scheduling of a predetermined number of patients of different types with stochastic durations in a multistage OR department. We consider the availability of several resources including ORs, recovery beds, and human resources such as surgeons. Furthermore, other constraints are considered in our model, such as the compatibility of resources and the number of available surgeons for each surgeon type. In addition, each surgeon is constrained by a time window, which indicates his/her availability in the scheduling horizon.

The first method, termed simulation-based tabu search (STS), integrates simulation with tabu search. The second method, integer programming enhanced tabu search (IPETS), improves the tabu search by incorporating an integer programming model. The third method, binary programming enhanced tabu search (BPETS) uses a binary programming model along with a heuristic and simulation-based tabu search to solve the problem.

In order to evaluate the performance of our methods, a number of test problems have been developed based on our findings in the OR department under study over an extended range of three major factors, namely, the number of patients, number of ORs, and coefficient of variability (CV) of service time. We then analyze the proposed methods based on their performance in terms of solution quality and the computation time. Furthermore, we examine the application of several scheduling rules (such as shortest/longest processing time, etc.) in comparison with the proposed methods. Based on this study, we provide insight into the applications of the proposed scheduling approaches to assist practitioners. Additionally, we study the application of BPETS in a case study of an OR department in a major Canadian hospital and compare the results with those of the actual schedules used in the OR department for several days.

The remainder of the paper is organized as follows: Section 2 discusses the relevant literature of outpatient surgery scheduling problem. Section 3 states the problem definition. Section 4 depicts the architecture of the proposed methods. Section 5 describes the design of experiments for testing and presents analysis of the test results. Section 6 presents the case study of an OR department. Finally, Section 7 discusses the conclusions and future work.

## 2. Literature review

In this section, we divide the relevant articles into two categories, optimization and simulation. Few articles combine simulation and optimization for patient appointment scheduling (e.g., Klassen and Yoogalingam, 2009). The existing literature can further be categorized based on their applications in clinics, or OR departments. For a comprehensive review of literature, readers are encouraged to refer to Cayirli and Veral (2003) for appointment scheduling of outpatient clinics, and to Blake and Carter (1997) and Cardoen et al. (2010) for surgery scheduling.

Many works in the optimization category use analytical methods to schedule appointments in healthcare. Although analytical methods can propose optimal schedules, they cannot easily model all the details and constraints in a complex environment. Therefore, they have focused on elements of the system, or have overly simplified the system. For instance, many optimization methods considered only single stage systems with Exponential or Erlang distributions for service times (Cayirli and Veral, 2003). Klassen and Yoogalingam (2009) pointed out that most of the proposed analytical methods are only valid for problems dealing with a small number of patients.

Within the optimization category, queuing theory has been widely used to solve clinic appointment scheduling problems. Most articles in this domain assume steady state behavior for the system, which is hardly achievable in healthcare environments (Cayirli and Veral, 2003).

In addition to the queuing theory, researchers have used mathematical programming (MP) as an analytical method to tackle the appointment scheduling problem. Hsu et al. (2003) developed a deterministic two-stage no-wait flow shop model for appointment scheduling of an ambulatory surgery clinic. Guinet and Chaabane (2003) developed a no-wait flow shop method for scheduling surgery cases. Pham and Klinkert (2008) proposed a deterministic MP model based on a multi-blocking job shop scheduling problem with the goal of minimizing makespan in surgery-case scheduling. Min and Yih (2010) proposed a stochastic programming model for case scheduling. They considered OR and surgical intensive care units that include several specialties. However, the model did not consider the intake procedure and other resources such as nurses, surgeons and equipment. They solved their model using a sample average approximation method.

Lamiri et al. (2009) developed a stochastic programming model for surgery planning to minimize elective patients' assignment costs and expected overtime costs. They considered elective and emergency cases and presented an "almost-exact" Monte Carlo simulation method. They studied the performance of their method in comparison with several heuristic and metaheuristic approaches (such as simulated annealing and tabu search). They reported that although for small to medium size test problems, their method has a better performance than the heuristic and metaheuristics methods, the computation time was significantly higher. For large problems, however, tabu search provided better solutions than those provided by the almost-exact method with a reasonable amount of time.

In brief, although MP has been used in several studies and delivered promising results, most MP methods (except for stochastic programming) do not address the stochastic nature of service times in outpatient clinic scheduling. Although stochastic programming can accommodate stochastic service time, they are usually analytically intractable, and suffer from long computation time. Similar to other analytical methods, MP lacks the capabilities required to capture all the intricacies that arise in complex-large systems.

Simulation is another approach that has been used to study the appointment scheduling problem. In contrast to analytical methods, simulation has the flexibility to model large and complex systems. Dexter et al. (1999) used simulation to address general surgery scheduling. They proposed a method to assign a time block to the surgeons and schedule patients to improve utilization of operating rooms. Marcon and Dexter (2006) used simulation to analyze the impact of different sequencing rules on OR utilization and workload of post anesthesia care units (PACU). Lowery and Davis (1999) used a simulation model to examine the effects of the surgery schedule and variability in surgery durations on the number of required beds. Overall, simulation does not include any optimization strategy, and therefore researchers have to pre-specify all potential solutions. In addition, simulation approaches may require long computation time that is not appealing for quick decision-making.

Few articles have considered the combination of simulation and optimization methods. Simulation-based optimization enjoys the flexibility of simulation in modeling complex system, while systematically seeks optimal solutions through its optimization component. Denton et al. (2006) considered an endoscopy suite and used simulation, which includes only two types of patients, and one surgeon type to address the problem. They used simulated annealing as an optimization tool to schedule the surgery cases in order to minimize facility overtime and patient waiting time. Klassen and Yoogalingam (2009) considered a single stage outpatient clinic and used OptQuest® to decide on the arrival time of patients. They studied dome patterns in appointment scheduling and suggested that practitioners can employ "a plateau-dome" type rule in many different environments. Gül et al. (2011) presented a simulation-based multi-objective genetic algorithm for the appointment scheduling of an outpatient procedure center. They found that the metaheuristics approaches did not offer schedules that are superior to the rules for a scheduling period of one day. This interesting finding is studied and discussed later in the current study.

In summary, existing analytical methods have difficulty in addressing large and complex systems, and therefore, mainly have focused on the elements of the system or simplified models of the system. Simulation methods can address many complexities in large systems, but are time-consuming and often do not deliver a competitive optimization strategy. Therefore, we observed a lack of efficient and effective methods in existing literature, which provide optimal or near-optimal solutions while encompassing details of real complex OR departments. In these OR departments, usually care providers deal with multistage systems that are constrained by limited resources while serving patients of different arrival and service time distributions.

This work tries to address the patient appointment scheduling problem under the following conditions, which eliminate many assumptions currently used in literature and should bring the problem closer to reality. We address the challenges of a multistage OR department that serves different types of patients with possibly different stochastic service times at each stage. In contrast to most existing articles, our approach addresses the stochastic service time of patients at each stage regardless of the type of their probability distribution function. We consider an OR department which includes multiple ORs and multiple recovery beds. In addition to resource availability constraints, the compatibility of resources is considered (i.e., each patient type can be only served by a specific surgeon type). Each surgeon follows a time window constraint, which determines the number of available surgeons of each type at each time block. The schedule is generated according to the Master Surgery Schedule (MSS) decided by management.

Based on simulation models, we introduce new optimization approaches to efficiently search for reliable solutions. Furthermore, we apply the proposed method to a case study OR department and compared its performance with the actual schedules. Finally, we study the application of several scheduling rules in appointment scheduling and their impacts on the waiting time and completion time. This study results in insight for practitioners who seek practical approaches for patient scheduling.

## 3. Problem description

In a typical OR department, each patient goes through three stages: pre-operation (surgery preparation), surgery, and recovery. In the first stage, an OR nurse identifies the patient and extracts patient's charts and information such as lab results and consent forms. This stage includes different preparation procedures required for each type of surgery such as taking drugs and anesthetics, having blood tests, and waiting for the medicine to take effect. The surgery stage includes anesthesia and operation. The last stage accounts for the procedures and the time required for patient's recovery. Fig. 1 represents the three stages of an OR department.

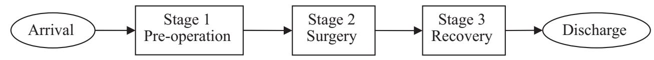


**Fig. 1.** Stages of an OR department.

We target the appointment scheduling of a specified number of patients in order to minimize the waiting time of patients, completion time, and number of cancellations. The completion time refers to the time that the last served patient leaves the post-anesthesia care unit (PACU). The cancellations account for the patients who could not be served due to the lack of time or resources. The number of patients and patient types are predetermined in the higher-level planning according to the available resources. The problem considers several patient types with stochastic service time at each stage whose arrivals are punctual. The stages in the department (except for the first stage) work according to the first-come–first-Served rule. The first stage admits the patients according to the schedule. Each type of patient is served by a specific specialty, i.e., the patient is served by a specific surgeon type, which intensifies the importance of resource compatibility in this problem.

Each type of surgeon encompasses a number of doctors of the same specialty. The number of available surgeons for each surgeon type per time block is provided by surgeon schedules, which is determined based on the surgeons' availability time window. The surgeon schedules are generated using master surgery scheduling (MSS), which is often developed by management, in the tactical level. In addition, the problem considers resources such as available ORs and available beds in PACU.

To provide a better understanding of the problem, consider scheduling of five patients of three different types. The specification of these patients is provided in Table 1.

In the interest of brevity, we used LOGN, TRIA, GAMM, BETA, WEIB, and EXPO to represent lognormal, triangular, gamma, beta, weibull, and exponential distributions, respectively. A specific type of surgeon serves each patient type. Assume that the department in this example includes two ORs that are shared by four surgeons. There are two beds in the preparation holding area, and two PACU beds. Surgeons' availability and type are provided in Fig. 2. Considering the patients' type, the availability of the surgeons, and the number of resources, we consider a possible schedule as depicted in Fig. 2. The circles represent patients and indicate the time that the patient is expected to arrive.

## 4. Methodology

This article proposes three simulation-based tabu search methods for outpatient scheduling in OR departments. The proposed STS method integrates simulation with tabu search. IPETS and BPETS methods improve on STS by incorporating an integer and binary programming models, respectively, with STS. The integer and binary programming in the last two methods solve the deterministic version of the problem. Tabu search utilizes the result of deterministic models as the initial solution. All proposed methods share the simulation model and tabu search components. In order to examine the applicability of

**Table 1**
Specification of patients.

| Patient | Patient type | Pre-operation time | Surgery time | Recovery time |
|---|---|---|---|---|
| 1 | Cardiac | LOGN(48.8, 12.4) | 391 × BETA(2.94, 4.96) | 240+EXPO(71.5) |
| 2 | Neurology | TRIA(14.5, 50, 77.5) | 490 × BETA(2.16, 6.3) | 45+270 × BETA(1.76, 4.06) |
| 3 | Orthopedic | 14.5+GAMM(17.7, 1.39) | LOGN(132, 200) | 40+WEIB(93.2, 1.22) |
| 4 | Orthopedic | 14.5+GAMM(17.7, 1.39) | LOGN(132, 200) | 40+WEIB(93.2, 1.22) |
| 5 | Neurology | TRIA(14.5, 50, 77.5) | 490 × BETA(2.16, 6.3) | 45+270 × BETA(1.76, 4.06) |

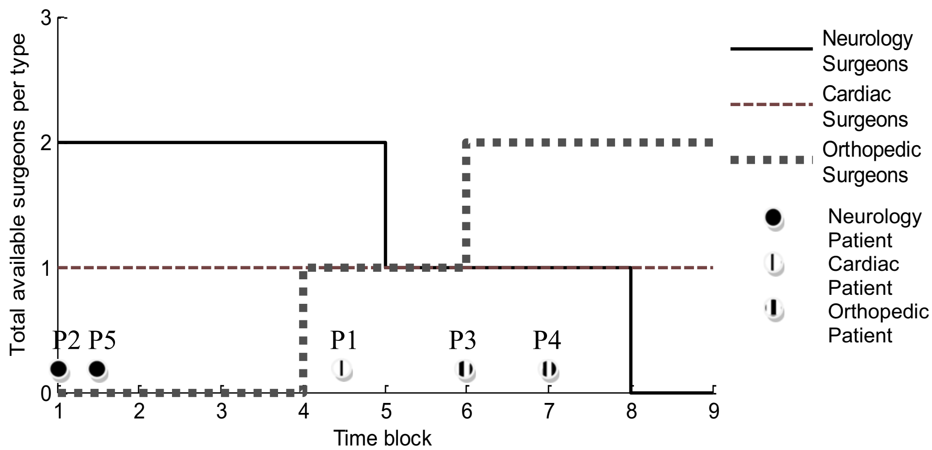


**Fig. 2.** Number of available surgeons in a session and a possible schedule.

proposed methods, we studied the performance of BPETS method on a simulation model built based on the actual data of an OR department in a Canadian hospital.

Furthermore, in this section, we study the application of scheduling rules in appointment scheduling of an OR department, which is then compared with the proposed methods.

### 4.1. Tabu search

Tabu search (TS) (Glover, 1989) is a metaheuristic method that has been successfully used to solve many optimization problems. This method iteratively proposes solutions using heuristic procedures while employing a flexible memory to guide the exploration in the solution space. Employing the memory and examining properties of solutions, it determines the search direction. In each iteration, a neighborhood of solutions is built based on a seed solution. This neighborhood consists of all solutions generated from the seed solution modified by the heuristic procedure. In this paper, we only describe the details of tabu search that is specific to this research. Readers are recommended to refer to Glover and Laguna (1997) for the detail of tabu search algorithms.

Tabu search in this work starts by determining an initial solution. In the STS method, we use an arbitrary schedule of surgeries as the initial solution. IPETS and BPETS, however, improve the performance of tabu search by starting the search from a more promising initial solution (generated by IP). The neighborhood of solutions is generated based on the determined initial solution by means of swap and insertion heuristics. In other words, the neighborhood consists of different surgery schedules that are generated based on the initial schedule. In STS, a number of solutions in the neighborhood are evaluated by simulation. The best solution based on the result of evaluations is nominated as the seed solution for the next iteration.

In order to reduce the number of simulation runs, BPETS and IPETS do not evaluate all solutions. They utilize a heuristic called *deterministic scheduling module* (DSM) which approximates the average waiting time and completion time to rank the solutions before evaluating them by simulation. Solutions in the neighborhood are ranked based on their objective value and a selected number of best solutions are put in the *candidate list*. The solutions in the candidate list are then evaluated using the simulation model. The best solution, based on the simulation model results, is then nominated as the seed solution for the next iteration.

In the next step, the nominated solution is checked against the tabu list and is used as the seed for the next iteration as long as it is not a tabu solution. However, a tabu solution may be used for the next iteration if it satisfies aspiration conditions. The aspiration rule used in this article allows tabu solutions that are better than the best known solution. Tabu search iteratively continues the described procedure till the termination condition is satisfied, which is set as the maximum allowed number of iterations in the proposed methods.

Although tabu search is a powerful optimization tool, its application does not guarantee reaching the optimal solution. Furthermore, it hardly uses the knowledge of the problem. In order to develop an efficient and effective optimization method, employing other methods to get a good initial solution seems beneficial. This strategy is what we do in IPETS and BPETS.

In order to incorporate the average waiting time of patients, along with the completion time of the facility, and the number of cancellations in the objective function of tabu search, we use a linear weighted-sum of these terms. Expression (1) presents the objective function used by tabu search, where $n$ is the number of patients, and $w_i$ is the waiting time of patient $i$. The objective function includes the completion time, $m$, and the number of cancellations, $\nu$. Coefficients $\alpha$, $\eta$ and $\xi$ are the weights that are assigned based on the empirical data or managerial preferences to reflect the relative importance of each term.

$$\alpha \sum_{i=1}^{n} \frac{w_i}{n} + \eta \times m + \xi \times \nu \tag{1}$$

### 4.2. Integer programming enhanced simulation-based tabu search (IPETS)

IPETS incorporates integer programming (IP) with simulation-based tabu search. The IP model represents the deterministic version of the problem, which uses the mean of service time distributions as the service time to construct the initial solution neighborhood of tabu search. IPETS uses the appointment schedule yielded from solving IP model. The integer programming model proposed in this work is an extension of the model by Jula and Leachman (2010), which is adapted for appointment scheduling.

In the IP model, the scheduling horizon is divided into a number of equal-length time grids. Each time block consists of one or more time grids. The discrete timeline assumes that the length of service times is an integer multiple of the basic time grid length. In addition, it assumes that each process starts only at the beginning of a time grid. The output of the IP model provides a surgery schedule with the minimum wait time and completion time assuming deterministic service times. The notation of the model is presented as follows:

Notation:

- $t$ discrete time index, $t=1,\ldots, T$, where $T$ is the time horizon and number of time grids in each day;
- $j$ stage index, $j=1, 2, 3$, where stage 1, 2, and 3 represent the pre-operation (holding area), operation (ORs), and post-operation (PACU) stages. Please note that the model is not restricted to the three stages and can be extended for the departments with more than three stages;
- $p$ patient type index, $p=1,\ldots, P$, where $P$ is the number of patient types;
- $s$ surgeon type index, $s=1,\ldots, S$, where $S$ is the number of surgeon types;
- $B_s$ set of patient types who can be served by surgeon type $s$. $p\in B_s$ means that the patient type $p$ can only be served by the surgeon type $s$.

Parameters:

- $o_t$ total capacity of operating rooms at time $t$;
- $l_{s,t}$ number of available surgeons of type $s$ at time $t$;
- $d_{j,p}$ service duration of patient type $p$ in stage $j$;
- $I_{j,p}$ initial number of patients of type $p$ in the line, waiting to be served at stage $j$ (including the patients who have appointments in the first stage);
- $R_j$ number of available resources in stage $j$ at the beginning of the scheduling horizon;
- $M$ an arbitrarily large number;
- $\gamma_p$ penalty coefficient of waiting a single time grid for a patient of type $p$;
- $\beta$ penalty coefficient of operating the clinic per time grid;
- $n_p$ number of nurses required for a patient of type $p$ in PACU.

Variables:

- $x_{j,t,p}$ number of patients of type $p$ at stage $j$ to start being processed at time $t$;
- $q_{j,t,p}$ number of patients of type $p$ who are waiting to be served at stage $j$ at time $t$;
- $X_{j,t,p}$ cumulative number of patients of type $p$ at stage $j$, whose treatment started by time $t$;
- $r_{j,t}$ number of available idle resources at stage $j$ at time $t$; each stage has its dedicated resource;
- $m$ last time block, in which all patients have been discharged (completion time);
- $y_t$ One if some patient is discharged at time $t$; 0 otherwise.

The optimization model is expressed as follows:

1. Objective function:

$$\text{Minimizing} \sum_j \sum_t \sum_p \gamma_p q_{j,t,p} + \beta m \tag{2}$$

2. Queue balance constraints:

$$q_{j,t,p} = I_{j,p} - X_{j,t,p} + X_{j-1,t-d_{j-1,p},p} \quad \forall j,t,p. \tag{3}$$

3. Cumulative variables definition:

$$X_{j,t,p} = \sum_{\tau=1}^{t} x_{j,\tau,p} \quad \forall j,t,p. \tag{4}$$

4. Capacity constraints:

$$r_{j,t} = R_j - \sum_p X_{j,t,p} + \sum_p X_{j,t-d_{j,p},p} \quad \forall j=1,2, \forall t, \tag{5}$$

$$r_{3,t} = R_3 - \sum_p n_p X_{3,t,p} + \sum_p n_p X_{3,t-d_{3,p},p} \quad \forall t. \tag{6}$$

5. Surgeon availability constraint:

$$\sum_{p\in B_s} \left( X_{2,t,p} - X_{2,t-d_{2,p},p} \right) \le l_{s,t} \quad \forall t,s. \tag{7}$$

6. OR availability constraint:

$$\sum_p \left( X_{2,t,p} - X_{2,t-d_{2,p},p} \right) \le o_t \quad \forall t. \tag{8}$$

7. Number of patients that have to be served:

$$X_{3,T-d_{3,p},p} = \sum_j I_{j,p} \quad \forall p. \tag{9}$$

8. The completion time indicator constraint:

$$M y_t \ge \sum_p x_{3,t-d_{3,p},p} \quad \forall t, \tag{10}$$

$$m \ge t \times y_t \quad \forall t. \tag{11}$$

9. Initial conditions and integer constraints:
$x_{j,t,p}$ and $r_{j,t}$ are non-negative integer variables. The values for $X_{j,t,p}$, $x_{j,t,p}$, and $r_{j,t}$ should be pre-specified for $t<0$ if applicable.

Expression (2) minimizes the total penalized sum of patients' waiting time and completion time. Eq. (3) defines the relationship among stages and maintains the patient flow within the system. It states that the number of patients in the queue of each stage is equal to the number of patients who were initially there plus the patients who entered the stage subtracted by patients who left so far. Eq. (4) determines the cumulative variables that address the total number of patients processed in each stage. Eq. (5) controls the number of available general resources in all stages through time. Resources such as receptionists and surgery preparation nurses are handled by this constraint. Eq. (6) addresses resource capacity of PACU. Expression (7) requires that there are enough number of surgeons of each type needed for surgery at each time block, considering the surgeons' availability and schedule. Expression (8) ensures that the total number of ongoing surgeries at each time block does not exceed the available number of ORs. Eq. (9) stipulates that all patients will be served by the end of the session. Expression (10) and (11) determine the last patients' discharge time block (completion time).

The Initial condition states that for equations such as Eqs. (3–5), the time block index is determined by terms in which $t$ is subtracted by the service time of patients. Having such a term in the time block index results in negative values when $t$ is smaller than service time. In this case, we need to assign zero values to the variables with negative time block index as an initial condition.

In contrast to the simulation model, IP model assumes that the service times must be multiples of a basic time grid. Note that the basic time grid of 15 min is used in our experiments. Using shorter time grids will result in an increase in the number of variables in the model, which increases the computation efforts to solve the model.

Although the IP model presents optimal surgery schedules for small to medium size problems, it may not efficiently provide optimal solutions for large problems due to huge computational efforts needed to solve the problems. Therefore, we relaxed the integrality constraints on the number of patients in the integer programming model and propose a new binary programming based algorithm in the next section.

### 4.3. Binary programming simulation-based tabu search (BPETS)

BPETS incorporates binary linear programming (BLP) with simulation-based tabu search. The BLP model used in this method is similar to the IP model used in IPETS while relaxing the integer assumption on the variables (excluding binary variable $y_t$). Consequently, its final results may contain non-integer values in schedules. For example, it may suggest that a non-integer number of patients appear at several time blocks. To remedy this problem, we develop a heuristic, termed *integer appointment constructor* (IAC), which takes the result of BLP model as the input and converts it to a feasible integer schedule as the output. IAC considers the schedule generated by the BLP model for each type of patients, and constructs appointments accordingly. The details of IAC are as follows:

| | |
|---|---|
| $p$ | patient type index; $p=1,\ldots,P$, where $P$ is the number of patient types; |
| $n_p^0$ | number of patients of type $p$; |
| $\bar{S}_t=\{\bar{s}_t\}$ | integer arrival schedule array for type $p$ patients, in which the number of arriving patients at time $t$, $\bar{s}_t$, is integer; |
| $\underline{S}_p=\{\underline{s}_t\}$ | non-integer arrival schedule array for type $p$ patients, in which the number of arriving patients at time $t$, $\underline{s}_t$, is not integer; |
| $n_p^n$ | the sum of all items of $\underline{S}_p$; |
| $n_c$ | counter for non-integer arrivals. |

**Algorithm 1. (Integer appointment constructor)**

**Step 0:** Increment $p$ by one; if $p \le P$, determine the number of patients in patient type $p$, $n_p^0$ from input string; otherwise stop.
**Step 1:** Determine $\bar{S}_p$and $\underline{S}_p$.
**Step 2:** Determine $n_p^n$; set $n_c=0$; set $t^*=0$.
**Step 3:** If $n_p^n=0$, go to Step 0; otherwise, for $t=1,\ldots,T$, set $n_c=n_c+\underline{s}_t$ until $n_c$ is an integer; set $t^*=t$;
**Step 4:** for $t=1, \ldots, t^*$
select the $n_c$ number of highest values of $\underline{s}_t$; increment the equivalent items in $\bar{S}_p$ by one, and for
$t=1, ,\ldots,t^*$, set $\underline{s}_t=0$; go to Step 2.

Fig. 3 shows an example of IAC algorithm assuming all patients are of the same type. Here, $\underline{S}_p=0.15, 0.55, 0, 0, 0.30, 0.33, 0, 0, 0.66, 0,$ and in the first iteration $n_p^n=2$ and $t^*=5$ with

| Time | Arrivals |
|---|---|
| 1 | 1.15 |
| 2 | 0.55 |
| 3 | 0 |
| 4 | 0 |
| 5 | 0.30 |
| 6 | 0.33 |
| 7 | 0 |
| 8 | 0 |
| 9 | 0.66 |
| 10 | 1 |

| Time | Arrival |
|---|---|
| 1 | 1 |
| 2 | 0 |
| 3 | 0 |
| 4 | 0 |
| 5 | 0 |
| 6 | 0 |
| 7 | 0 |
| 8 | 0 |
| 9 | 0 |
| 10 | 1 |

| Time | Arrival |
|---|---|
| 1 | 0.15 |
| 2 | 0.55 |
| 3 | 0 |
| 4 | 0 |
| 5 | 0.30 |
| 6 | 0.33 |
| 7 | 0 |
| 8 | 0 |
| 9 | 0.66 |
| 10 | 0 |

| Time | Arrival |
|---|---|
| 1 | 1 |
| 2 | 1 |
| 3 | 0 |
| 4 | 0 |
| 5 | 0 |
| 6 | 0 |
| 7 | 0 |
| 8 | 0 |
| 9 | 1 |
| 10 | 1 |

**Fig. 3.** An example of IAC heuristic.

$n_c=1$. Then for $t=1,\ldots, 5$, $\underline{s}_t=0$ and $\bar{s}_2=1$. Then, in the second iteration $n_p^n=1$, and $t^*=9$ with $n_c=1$. For $t=1,\ldots,9$, $\underline{s}_t=0$ and $\bar{s}_9=1$.The two largest $\underline{s}_t$are located in the 2nd and 9th positions of $\bar{S}_p$ in the first and second iteration. The algorithm is terminated in the third iteration as $n_p^n=0$.

### 4.4. Simulation model

The proposed methods use a *discrete-event simulation* model to evaluate the schedules generated by optimization components. The simulation model encompasses three stages of the OR department including holding area, ORs, and PACU, as depicted in Fig. 1. At the beginning of each simulation run, the simulation model accepts, as inputs, the patient schedule and parameters of the problem. Problem parameters include the number of patient types, number of patients in each patient type, information on patient service times at each stage, number of surgeon types, schedule of each surgeon type, number of ORs, and number of PACU beds.

Using stochastic service time in the simulation model, we perform 30 replications of each simulation run for each schedule. Based on several experiments, we determined that 30 replications is an appropriate tradeoff between computation time and half width of the confidence intervals. Over the 30 replications, the simulation model tallies and reports the results as the final output. The simulation model has been validated using actual data of 24 days of the OR department described in Section 6, considering the service times derived from the actual data.

### 4.5. Scheduling rules

In this section, we attempt to study the application of several scheduling rules in appointment scheduling of patients to establish a reference for comparing the results. Many practitioners rely on a first-come–first-serve rule to assign the patient appointments based on available spots in the schedule. In order to address the variability of service times in appointment scheduling rules, we use the job hedging approach, introduced by Yellig and Mackulak (1997) and implemented by Gül et al. (2011) in scheduling of an outpatient clinic. In this approach, the adjusted service time, $S'=\mu+\alpha\sigma$, is used for scheduling, where $\mu$ is the mean of service time, $\sigma$ is the standard deviation of service time, and $\alpha\in[0,1]$ is a real number. Furthermore, in this section we propose a new sequencing rule inspired by the finding of Robinson and Chen (2003). We also consider examining sequencing patients based on adjusted service times in addition to solely rely on the mean of service time for scheduling.

We used the following scheduling rules: (a) increasing mean of service time (shortest processing time, SPT), (b) decreasing mean of service time (longest processing time, LPT), (c) increasing variance of service time (SVR), (d) increasing coefficient of variability of service time (SCV), and (e) dome-shape rule (DSR). In addition to using mean of service time, we considered applying SPT and DSR rules on adjusted service times, termed stochastic SPT (SSPT) and stochastic DSR (SDSR), respectively. SVR sorts

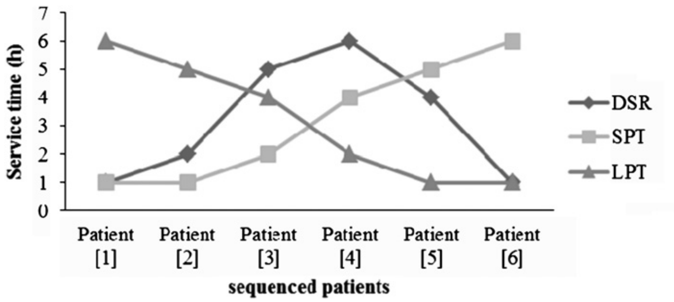


**Fig. 4.** DSR, LPT and SPT scheduling rules.

patients increasingly according to their variance of service time. SCV sorts the patients increasingly according to coefficient of variation of their service time.

The DSR first decreasingly sorts the patients based on their service time. Then, it starts generating the patients' sequence by placing the patient with the largest service time in the middle of sequence and putting patients with less service time before and after it alternatively. Robinson and Chen (2003) suggested in their article that optimal appointment intervals present a dome pattern for patients' service time in appointment scheduling of an outpatient clinic. We adopted this insight from outpatient clinic scheduling and extended them to surgery scheduling. For instance, if we consider six patients and their associated service times of {1, 4, 2, 1, 5, 6}, the sequence resulted by DSR will be {P1, P3, P5, P6, P2, P4}. Whereas, applying SPT and LPT results in {P1, P4, P3, P2, P5, P6} and {P6, P5, P2, P3, P4, P1}, respectively. Fig. 4 compares outcomes of three scheduling rules of DSR, SPT, and LPT for these patients. In Fig. 4, the brackets represent the order of patients' arrival. For example, "Patient[3]" represents the third patient arriving. The resulted sequence indicates that DSR places the patients with longer service times in the middle of sequence while sets the ones with shorter service time in the beginning or end of the sequence.

Considering a specific patient sequence, the next step is the appointment time determination for each patient. As our study indicates that the OR stage is the bottleneck of the system, we use the OR service time to determine the appointment times. In order to determine the appointment time, we need to define following items:

| | |
|---|---|
| $i$ | patient index, $i=1,\ldots,N$; where $N$ is number of patients; |
| $l$ | OR index, $l=1,\ldots,n$; where $n$ is the number of ORs; |
| $c_l$ | variable which represents the earliest time that $l$th OR becomes idle; |
| $s'_i$ | adjusted service time of $i$th patient which is defined earlier in this section. |

The earliest available OR is noted by $k$, and the time this OR becomes available is $c_k=\min(c_l)$, $l=1$ to $n$. The appointment time for patient is $A_i=c_k+s'_t$. The value of $c_k$ is updated to $c_k+s'_i$ after the patient is assigned to the $k$th OR.

## 5. Experiments and results

In this study, the Rockwell Arena$^{\text{TM}}$ version 12 software was used for the simulation model; the proposed tabu method was implemented with Microsoft Visual C♯, which was integrated with Arena$^{\text{TM}}$. The integer and binary programming models have been developed and solved by GAMS$^{\text{TM}}$ 225 software tool. We use 2.53 GHz Intel® Core 2 Duo CPU with 3 GB RAM to perform experiments.

In order to develop a number of test problems, we change the levels of several factors that are expected to be the most effective on the performance of the algorithms. Based on our preliminary experiments, three factors have been identified—the number of patients, number of operating rooms and coefficient of variance of service.

**Table 2**
Specification of test problem based on patient types.

| Total no. of patients | No. of PACU beds | No. of patients per patient type | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | PT 1 | PT 2 | PT 3 | PT 4 | PT 5 | PT 6 | PT 7 | PT 8 | PT 9 | PT 10 |
| 15 | 6 | 3 | 3 | 3 | 3 | 3 | | | | | |
| 33 | 10 | 4 | 2 | 4 | 5 | 5 | 4 | 4 | 5 | | |
| 50 | 14 | 3 | 4 | 4 | 5 | 3 | 4 | 5 | 4 | 7 | 6 |

To analyze the performance of the proposed methods with respect to the size of the problems, we generated problems with 15, 33, and 50 patients per day. This range not only covers but also exceeds the range of the number of patients served in the OR department that has been studied in the case study section. Depending on the type of surgeries, these problems may represent patient load of a medium to large OR department. Table 2 presents the configuration of test problems and includes the number of patients of each type and the number of resources. PT stands for patient type in the table.

The second factor employed in the development of the test problems is the number of operating rooms. We selected two different values for each test problem. Test problems with 15 patients include four and five ORs, while eight and nine ORs are used for the test problems with 33 patients. For the test problems with 50 patients, 9 and 10 operating rooms are used.

Since the variance of service times plays a significant role in the magnitude of actual service times at each stage, we use coefficient of variability (CV) as the third factor involved in the design of test problems. CV is defined as the ratio of variance to the mean of service time. We use the same CV for the service time in the surgery stage. Three different levels of CV have been used for the test problems, namely 0.5, 1, and 2. These values of CV are selected based on the extended range of the CV of service time distributions obtained in our case study for different patient types. In the design of test problems, we used the same type of patients who were available in our case study. For each patient type, the service times of Stages 1 and 3 are taken from the case study. We use the lognormal distribution function in the surgery stage as suggested by several studies, see for example Zhou and Dexter (1998). In the surgery stage, the mean is taken from the actual data obtained in the case study, and variance is modeled by a given level of CV. The proposed methods, however, are not limited to the chosen types of distribution functions.

We conducted two studies to evaluate the performance of our methods. First, we compared the three proposed methods based on the quality of their solutions and the computation time. We assess the quality of solutions based on three criteria of average waiting time, completion time, and number of cancellations. The second study includes comparing the performance of the proposed methods with a set of scheduling rules.

### 5.1. Performance study of proposed methods

Based on the selected factors and the levels for each factor, 18 test problems are generated. Each test problem runs ten times, using a different random seed each time. In total, 180 runs are performed for each method. Each run is limited to 300 function evaluations.

We examined the methods based on their efficiency and effectiveness. We measure efficiency based on the CPU time (in seconds) required by each method for completing 300 function evaluations. The effectiveness is measured based on the average waiting time, completion time, and number of cancellations resulted by each method.

Fig. 5 compares the average waiting time of patients yielded by the proposed methods. We observe that as the size of the problem grows, the gap between the performance of the methods which benefit from a MP model (IPETS and BPETS), and STS becomes more significant, and the methods which use the MP model present superior results. Considering medium and large problems, IPETS delivers better results than STS for all instances, and competitive results compared to BPETS.

Fig. 6 compares the performance of our methods in terms of completion time. The completion times of the methods enhanced by MP are significantly less than those offered by STS. In this figure, we contrasted a set of problems with the same number of patients and ORs by drawing an oval around them. Similar ovals can be constructed for other sets with the same number of patients and ORs. The result suggests that the CV of surgery service time plays a significant role in the completion times and increasing the CV significantly increases the completion time.

Table 3 compares three proposed methods in terms of effectiveness classified based on the total number of patients, number

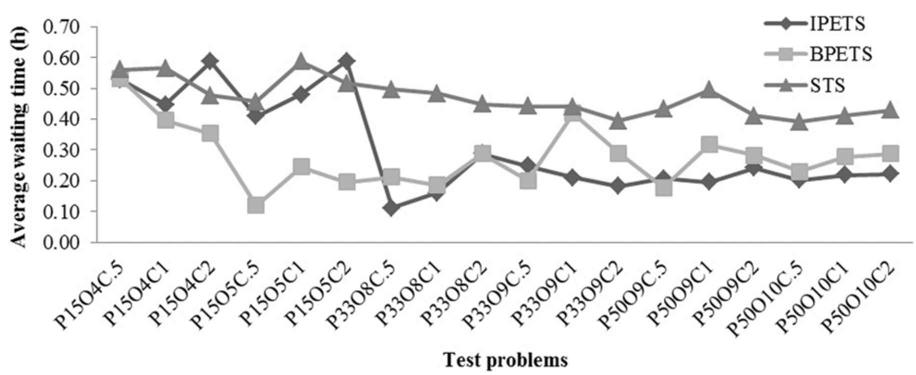


**Fig. 5.** Comparison of the average waiting time from the proposed methods.

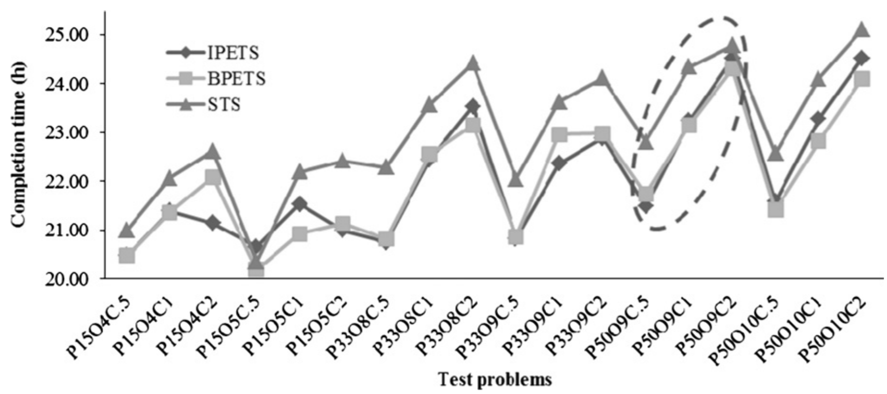


**Fig. 6.** Comparison of the completion time of the proposed methods over 18 test problems.

**Table 3**
Comparison of proposed methods in terms of waiting time and completion time.

| Test problem | No. of patients | No. of ORs | CV | 95% CI of average waiting time (h) | | | 95% CI of completion time (h) | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | IPETS | BPETS | STS | IPETS | BPETS | STS |
| P15O4C.5 | 15 | 4 | 0.5 | [0.41,0.65] | [0.45,0.62] | [0.46,0.66] | [20.22,20.72] | [20.19,20.76] | [20.73,21.27] |
| P15O4C1. | | | 1.0 | [0.33,0.56] | [0.29,0.50] | [0.41,0.72] | [21.06,21.73] | [21.07,21.65] | [21.76,22.38] |
| P15O4C2. | | | 2.0 | [0.48,0.70] | [0.26,0.45] | [0.35,0.60] | [20.86,21.43] | [21.75,22.42] | [22.34,22.90] |
| P15O5C.5 | | 5 | 0.5 | [0.32,0.50] | [0.08,0.16] | [0.37,0.55] | [20.34,20.97] | [19.96,20.40] | [20.09,20.60] |
| P15O5C1. | | | 1.0 | [0.39,0.57] | [0.21,0.28] | [0.47,0.71] | [21.17,21.89] | [20.78,21.06] | [21.77,22.61] |
| P15O5C2. | | | 2.0 | [0.53,0.64] | [0.18,0.21] | [0.43,0.60] | [20.62,21.35] | [20.87,21.39] | [22.14,22.70] |
| P33O8C.5 | 33 | 8 | 0.5 | [0.08,0.14] | [0.16,0.27] | [0.43,0.57] | [20.55,20.95] | [20.66,20.98] | [22.04,22.52] |
| P33O8C1. | | | 1.0 | [0.11,0.21] | [0.12,0.26] | [0.43,0.54] | [22.18,22.71] | [22.24,22.87] | [23.31,23.83] |
| P33O8C2. | | | 2.0 | [0.19,0.38] | [0.20,0.38] | [0.37,0.53] | [23.09,23.97] | [22.69,23.62] | [24.11,24.74] |
| P33O9C.5 | | 9 | 0.5 | [0.15,0.34] | [0.14,0.26] | [0.36,0.52] | [20.53,21.12] | [20.62,21.11] | [21.81,22.28] |
| P33O9C1. | | | 1.0 | [0.12,0.29] | [0.28,0.56] | [0.38,0.50] | [22.08,22.64] | [22.64,23.29] | [23.38,23.87] |
| P33O9C2. | | | 2.0 | [0.12,0.24] | [0.20,0.37] | [0.33,0.45] | [22.62,23.15] | [22.71,23.25] | [23.77,24.47] |
| P50O9C.5 | 50 | 9 | 0.5 | [0.15,0.26] | [0.14,0.22] | [0.38,0.49] | [21.26,21.71] | [21.46,22.01] | [22.52,23.10] |
| P50O9C1. | | | 1.0 | [0.18,0.21] | [0.23,0.40] | [0.44,0.55] | [22.91,23.57] | [22.77,23.54] | [24.02,24.67] |
| P50O9C2. | | | 2.0 | [0.17,0.31] | [0.21,0.35] | [0.34,0.48] | [24.10,24.93] | [23.88,24.74] | [24.49,25.08] |
| P50O10C.5 | | 10 | 0.5 | [0.13,0.27] | [0.18,0.28] | [0.34,0.44] | [21.31,21.85] | [21.08,21.74] | [22.27,22.88] |
| P50O10C1. | | | 1.0 | [0.17,0.27] | [0.22,0.34] | [0.38,0.44] | [22.90,23.66] | [22.59,23.04] | [23.83,24.39] |
| P50O10C2. | | | 2.0 | [0.15,0.29] | [0.22,0.36] | [0.34,0.51] | [24.14,24.91] | [23.71,24.48] | [24.80,25.43] |

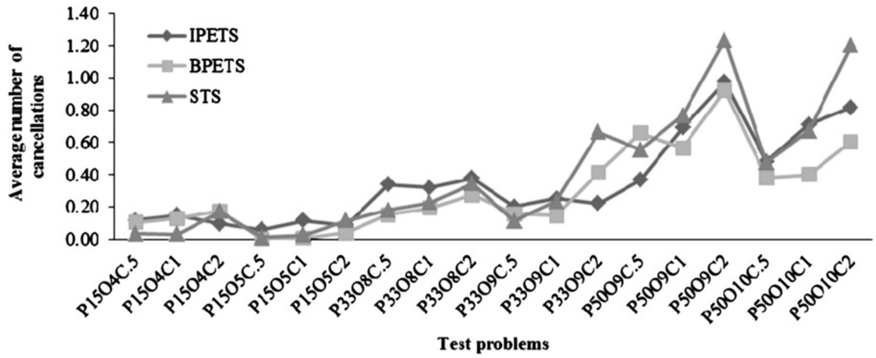


**Fig. 7.** Comparison of the average number of cancellations of proposed methods.

**Table 4**
Comparison of CI on the average number of cancellations.

| Test problem | No. of patients | No. of ORs | CV | 95% CI of the number of cancellations | | |
|---|---|---|---|---|---|---|
| | | | | IPETS | BPETS | STS |
| P15O4C.5 | 15 | 4 | 0.5 | [0.00,0.24] | [0.00,0.22] | [0.00,0.06] |
| P15O4C1. | | | 1.0 | [0.08,0.21] | [0.05,0.21] | [0.01,0.05] |
| P15O4C2. | | | 2.0 | [0.06,0.13] | [0.09,0.26] | [0.12,0.22] |
| P15O5C.5 | | 5 | 0.5 | [0.01,0.10] | [0.00,0.00] | [0.00,0.03] |
| P15O5C1. | | | 1.0 | [0.02,0.20] | [0.00,0.01] | [0.00,0.04] |
| P15O5C2. | | | 2.0 | [0.07,0.11] | [0.03,0.04] | [0.09,0.14] |
| P33O8C.5 | 33 | 8 | 0.5 | [0.10,0.57] | [0.03,0.27] | [0.02,0.34] |
| P33O8C1. | | | 1.0 | [0.05,0.59] | [0.05,0.34] | [0.12,0.33] |
| P33O8C2. | | | 2.0 | [0.13,0.63] | [0.15,0.40] | [0.23,0.45] |
| P33O9C.5 | | 9 | 0.5 | [0.00,0.55] | [0.02,0.29] | [0.03,0.19] |
| P33O9C1. | | | 1.0 | [0.06,0.44] | [0.00,0.31] | [0.03,0.44] |
| P33O9C2. | | | 2.0 | [0.15,0.29] | [0.18,0.65] | [0.41,0.92] |
| P50O9C.5 | 50 | 9 | 0.5 | [0.24,0.49] | [0.52,0.79] | [0.26,0.85] |
| P50O9C1. | | | 1.0 | [0.53,0.85] | [0.32,0.81] | [0.55,0.98] |
| P50O9C2. | | | 2.0 | [0.76,1.19] | [0.72,1.13] | [0.97,1.50] |
| P50O10C.5 | | 10 | 0.5 | [0.15,0.81] | [0.15,0.61] | [0.21,0.73] |
| P50O10C1. | | | 1.0 | [0.45,0.97] | [0.31,0.49] | [0.43,0.91] |
| P50O10C2. | | | 2.0 | [0.62,1.01] | [0.47,0.74] | [0.91,1.49] |

**Table 5**
95% CIs on average computation time of proposed methods.

| Test problem | No. of patients | No. of ORs | CV | 95% CI of computation time (s) | | |
|---|---|---|---|---|---|---|
| | | | | IPETS | BPETS | STS |
| P15O4C.5 | 15 | 4 | 0.50 | [1030,1045] | [1038,1046] | [1020,1037] |
| P15O4C1. | | | 1.00 | [1024,1032] | [1032,1042] | [1024,1034] |
| P15O4C2. | | | 2.00 | [1022,1037] | [1028,1036] | [1022,1032] |
| P15O5C.5 | | 5 | 0.50 | [1062,1073] | [1076,1092] | [1061,1073] |
| P15O5C1. | | | 1.00 | [1064,1075] | [1074,1089] | [1061,1076] |
| P15O5C2. | | | 2.00 | [1068,1078] | [1070,1084] | [1058,1071] |
| P33O8C.5 | 33 | 8 | 0.50 | [1376,1385] | [1378,1393] | [1384,1400] |
| P33O8C1. | | | 1.00 | [1375,1398] | [1380,1396] | [1379,1398] |
| P33O8C2. | | | 2.00 | [1377,1401] | [1372,1389] | [1381,1395] |
| P33O9C.5 | | 9 | 0.50 | [1427,1440] | [1421,1441] | [1421,1439] |
| P33O9C1. | | | 1.00 | [1421,1441] | [1414,1429] | [1419,1440] |
| P33O9C2. | | | 2.00 | [1417,1435] | [1417,1431] | [1409,1435] |
| P50O9C.5 | 50 | 9 | 0.50 | [1624,1646] | [1650,1669] | [1624,1645] |
| P50O9C1. | | | 1.00 | [1626,1649] | [1646,1672] | [1625,1643] |
| P50O9C2. | | | 2.00 | [1622,1649] | [1635,1656] | [1602,1625] |
| P50O10C.5 | | 10 | 0.50 | [1650,1672] | [1669,1690] | [1646,1666] |
| P50O10C1. | | | 1.00 | [1645,1663] | [1666,1685] | [1651,1671] |
| P50O10C2. | | | 2.00 | [1647,1667] | [1668,1690] | [1638,1656] |

of ORs, and CV of the problems. This table contains 95% confidence intervals (CI) on the mean for each test problem based on the results obtained over multiple runs. Considering the result presented in Table 3 and Fig. 5 and Fig. 6, we conclude that IPETS and BPETS outperform STS in terms of waiting time and completion time, as their CIs do not overlap the CIs of STS for most cases (Table 3) while they present lower average values (Fig. 5, Fig. 6).

Fig. 7 compares the average number of cancellations for proposed methods. It is shown that the number of cancellations delivered by all methods is comparable in small size problems. However, in the large size problems with large CV, the performance of BPETS is superior to the other methods (e.g., P50O10C2). Overall, the number of cancellations grows as the number of patients and CV increase in the test problems.

Table 4 addresses the cancellation for the proposed methods. This table includes the 95% confidence intervals on the mean number of cancellations for each test problem. The number of cancellations rises as the number of patients and CV increase.

Table 5 shows the 95% CIs on the average of computation times of the proposed methods. Computation time for BPETS and IPETS includes the time needed for solving the MP model and tabu search method. Overall, the computation times of the methods are comparable. For current test problems, we observed that the integer programming model does not require a long computation time to solve. However, based on our experiments, if a larger number of patient types is considered (e.g., if each specialty is broken down into major procedures), the computation time of IPETS would be significantly longer, which makes it a less appealing approach in industry.

Fig. 8 compares the penalized objective value of the results of IP model and IPETS. The result of the IP model has been evaluated through the simulation model using Expression (1). Based on the managerial preferences and our observations in the case study department, the objective function coefficients $\alpha$, $\eta$ and $\xi$ are set to be equal 1, 1, and 100 h, respectively, reflecting the significant negative impact of cancellations on the performance of the department. This figure shows that in most of the test problems, IPETS finds better solutions compared with the initial solutions offered by the IP model.

In summary, we observed that using mathematical programming in metaheuristics improves the performance of methods in terms of solution quality. Furthermore, while BPETS may not outperform IPETS in all criteria of solution quality, it yields quality results for large size problems in reasonable computation time.

### 5.2. Scheduling rules

In order to study the performance of the scheduling rules compared with the proposed methods, we considered the test problem with 15 patients. We considered several scheduling rules along with several hedging factors, $\alpha$. Scheduling rules include two stages of sequencing patients and determining appointment time. In sequencing steps, we utilize the mean of the service time distributions for scheduling rules (SPT, LPT, DSR, SCV, and SVAR). In sequencing stage, SPT, LPT, SVR, SCV, and DSR use mean of

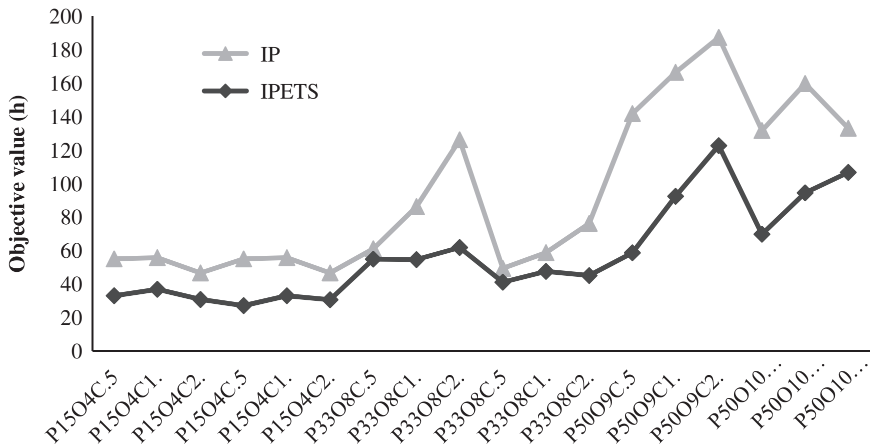


**Fig. 8.** A comparison of the result obtained from IP model and IPETS.

service time, whereas SSPT and SDSR are based on adjusted service time (which uses the mean and standard deviation) instead of only the mean service time. Using the increments of 0.05 for the hedging factor ($\alpha$), 16 values of hedging factor are determined which ranges from 0 to 0.75 for each scheduling rule. Consequently, we deliver 16 schedules for each scheduling rule.

After the determination of appointment times, the simulation model evaluates the schedule. We monitor three criteria, namely, average waiting time of patients, completion time, and the number of cancellations. We report the average value of these metrics over 10 simulation runs using different random seeds. Each run includes 30 replications of the simulation.

Our preliminary experiments show that scheduling rules result in many cancellations, as these approaches do not address the surgeons' time window constraints. We then focus on studying a new set of test problems by relaxing the surgeon's schedule in order to evaluate the performance of the scheduling rules. Fig. 9 presents the performance of scheduling rules on the 15-patient test problem with relaxed surgeon schedule. In this figure, each scheduling rule consists of 16 points corresponding to 16 levels of the hedging factor, $\alpha$. For each scheduling rule, these points are spread from left ($\alpha=0.75$) to the right ($\alpha=0$) in the figure. This figure also includes the results generated by IPETS, BPETS, and STS methods. IPETS, BPETS, and STS are presented each by a single point in the plot because they produce a single solution for the given problem (no hedging factor involved).

Our experiments show that DSR, SDSR, SVAR, and SCV have superior performance among the studied scheduling rules. DSR and SDSR present schedules with short waiting time and long completion time. On the other hand, SCV and SVAR deliver schedules with short completion time and long waiting time values. In terms of cancellations, DSR and SDSR present the lowest number of cancellations. Results suggest that extreme values for the hedging factor might result in either long completion times or long waiting times. Our experiments indicate that scheduling rules are not a competent approaches when dealing with providers with time window constraint (e.g., surgeons in OR departments). Scheduling rules, however, can be utilized in systems that have relaxed time windows for the service providers.

Concerning the performance of proposed simulation-based methods, we observed that STS method presents inferior results in terms of completion time, while delivering schedules with the smallest number of cancellations. Although STS presents promising waiting time and number of cancellations, completion time is significantly larger than those generated by IPETS and BPETS. We observed that the proposed methods enhanced with MP models deliver significantly better results, especially in terms of completion time. The large completion time for STS makes it an inferior option, as there are scheduling rules (e.g., SVR) leading to comparable waiting time with less completion times. Also, recall that using scheduling rules to schedule the appointment results in many cancellations when the surgeons' time window constraint is considered; Fig. 9 shows only results when this constraint is lifted. Results of proposed methods are circled.

In summary, we conclude that in the problems without time constraints, ordinary metaheuristic methods (such as tabu search) may not offer better schedules compared with scheduling rules. This conclusion confirms the insights achieved by Gül et al. (2011). However, using MP models to reinforce the metaheuristics may significantly improve the performance of metaheuristics where availability of surgeons is restricted by time window constraints.

## 6. Case study on completion times

In this section, we apply the BPETS method, which is found to be promising in previous sections, to a case study OR department. In the previous section experiments indicated that both IPETS and BPETS present quality solutions. However, IPETS relies on integer programming and it may present difficulty in solving large problems due to its required large computational efforts.

Our case study is an OR department of a major hospital in Canada, which includes 13 ORs to serve elective and emergency patients. The ORs are used for most procedures and are shared among the specialties. The space of the OR department is separated into three sections including the holding area, ORs, and post anesthesia unit (PACU).

The holding area is used for the preparation of the patients before entry to the OR which includes procedures such as the identification of patients, checking the consent form and lab tests, etc. In addition to a surgeon, usually three nurses along with an anesthetist are required for each surgery. The nurses and anesthetists are assigned to an OR for a complete shift. However, each surgeon is available throughout the whole operation of the patient, which is supposed to fall in the time window of his or her availability. The PACU is the main recovery unit, which can accommodate up to 12 patients. In cases where more beds are needed for recovery, the surgical intensive care unit (SICU) is able to accommodate 10 patients. SICU often serves patients with major surgical procedures. Historical data indicates that on normal days 70–80% of the cases are elective surgeries, which are assigned to the ORs and resources in the surgery scheduling. The patient processing starts at 7:30 am at the OR department.

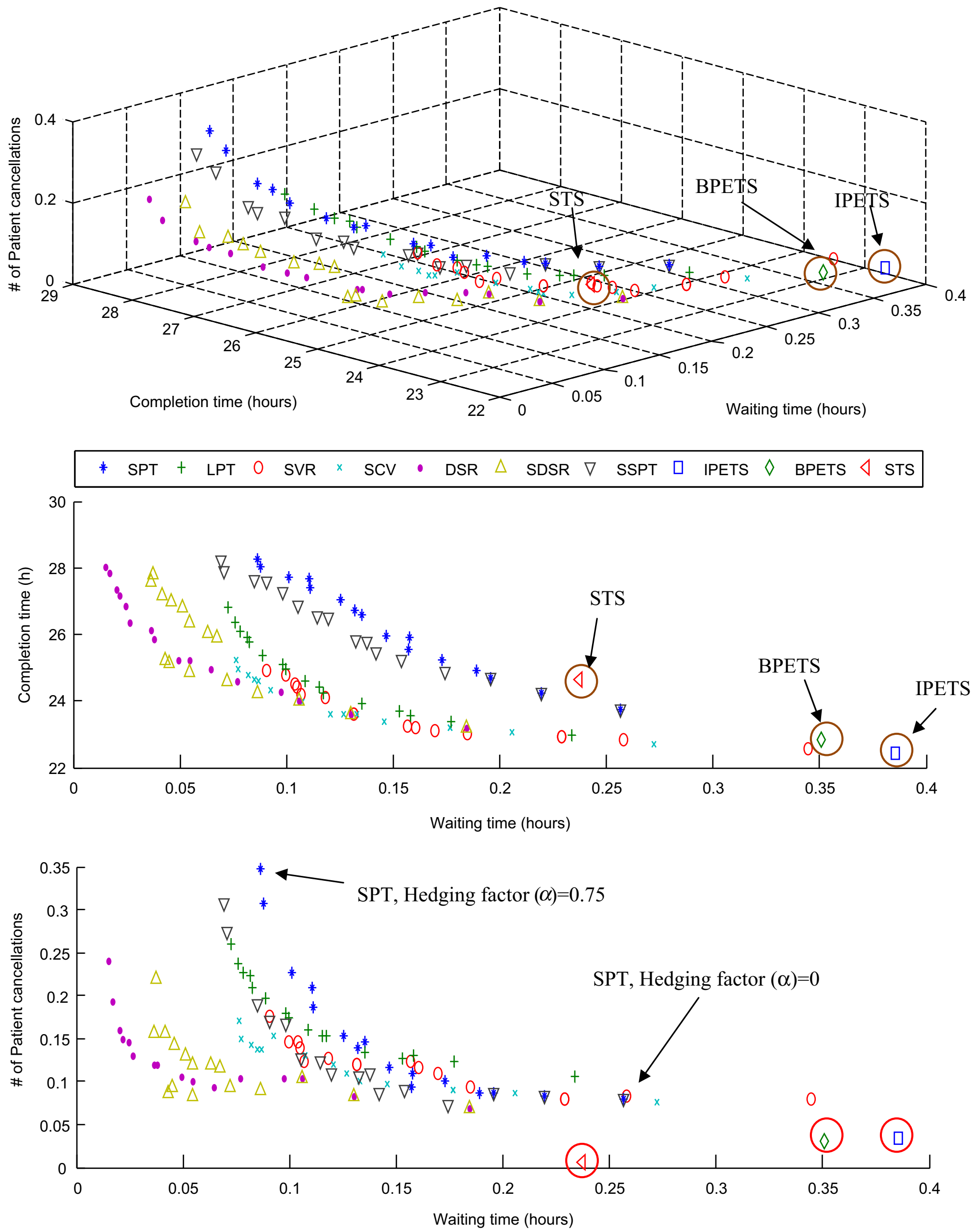


**Fig. 9.** Comparison of the performance of multiple scheduling rules with STS, IPETS, and BPETS on a test problem with 15 patients and relaxed surgeon schedule.

The PACU unit is open until 12:00 am. However, if a patient is not ready to be discharged, she or he will be sent to SICU.

We broke down the activities in a patient's visit into three stages of pre-operation, operation, and post-operation. Based on the available data, we developed service time probability density functions of different patient types for each stage. We considered 11 patient types based on the 11 existing specialties in the OR department. The service time of each patient type at each stage has been determined using available historical data. Table 6 presents the service times in this case study for each specialty.

We used the case study OR department in order to evaluate the performance of BPETS in comparison with the actual appointment schedules. We considered 24 days of studied OR department and applied BPETS method in order to schedule the same set of patients for each day. The performance of BPETS, in terms of completion time, is then compared with (a) the actual completion time of each day extracted from the historical data, and (b) the completion time of the actual schedule of each day using the simulation model. Fig. 10 presents the comparison between BPETS and actual practice in terms of completion time.

The comparison of BPETS and simulated actual schedules indicates that the application of BPETS results in a significant reduction of completion time in the studied OR department.

## 7. Conclusion

Our survey on scheduling outpatient surgeries points out a gap in existing literature for efficient and effective methods that present competitive optimization schemes while addressing different aspects of complex real world systems. To address this gap, we proposed three simulation-based tabu search methods which benefit from the flexibility of simulation, and the power of mathematical programming optimization to find quality solutions for outpatient scheduling in OR departments. The proposed simple tabu search (STS) method integrates simulation with optimization. IPETS and BPETS methods improve on STS by incorporating integer and binary programming models, respectively.

We conducted three studies to examine the performance of our methods. First, we compared the performance of the three proposed methods in terms of waiting time of patients, completion time, number of cancellations, and computation times. Second, we studied the performance of scheduling rules in appointment scheduling of OR departments and compared them to the proposed simulation-based methods. Finally, we applied the BPETS to a case study OR department. To analyze the performance of the proposed methods, we designed experiments over a range of important factors—the number of patients, number of ORs, and coefficient of variability of service times. We determined the range of the factors based on the insights obtained in the case study.

In the performance study of the proposed methods, our observation suggests that although STS, BPETS and IPETS require approximately the same amount of computation time, STS results in consistently inferior solutions as compared to IPETS and BPETS. Thus, STS is not recommended for practical purposes. This further confirms the significant effects of employing mathematical programming in improving performance of metaheuristic approaches.

Comparison of IPETS and BPETS indicates that both methods present quality solutions. However, since IPETS relies on integer programming, it may present difficulties in solving large problems since it requires long computational time. Therefore, we recommend application of BPETS for practical purposes based on its effectiveness and efficiency in solving test problems ranging from small to large.

Our experiments indicate that methods based on scheduling rules are not competent approaches when dealing with providers restricted with time window constraints (e.g., surgeons in OR departments). Application of scheduling rules may result in several cancellations because these rules do not consider the time constraints. However, scheduling rules can be utilized in the systems that include providers with relaxed time window constraints. DSR, SDSR, SVAR and SCV have superior performance among the studied scheduling rules. DSR and SDSR present schedules with small waiting times and large completion times. This result suggests that the appointment rules that present a dome pattern may result in minimum patients waiting times. On the other hand, SCV and SVAR deliver schedules with short completion time and long waiting time values.

Applying BPETS in the case study suggested that application of metaheuristics enhanced with MP models improve the appointment scheduling of the case study in terms of completion time.

Future research may be pursued in several directions. First, the proposed methods can be extended in addressing multi-objective problems such as minimizing waiting time and completion time while maintaining a minimum utilization in the system. Second, using mathematical programming models proposed in this study to develop a dynamic scheduling method is another option to extend this study. Dynamic scheduling can help clinic administrators to react efficiently and effectively to rapidly changing environments such as failure of equipment, sudden changes in staffing level, or in emergency cases. Finally, the proposed architecture of combining tabu search, simulation, and mathematical programming with different objective functions could be used to develop applications in other sectors, which involve flow of customers, patients, or parts.

**Table 6**
Processing time of different type of patients in the case study OR department.

| Specialty | Distribution | Mean (min) | Standard deviation (min) |
|---|---|---|---|
| **Cardiac** | LOGN(48.8, 12.4) | 48.8 | 12.4 |
| | 391BETA(2.94, 4.96) | 145.5 | 63.3 |
| | 240+EXPO(71.5) | 311.5 | 71.5 |
| **Vascular** | 12+WEIB(31.6, 1.77) | 40.1 | 16.4 |
| | LOGN(98.5, 154) | 98.5 | 154.0 |
| | LOGN(221, 91.5) | 221.0 | 91.5 |
| **Neurology** | TRIA(14.5, 50, 77.5) | 47.3 | 12.9 |
| | 490BETA(2.16, 6.3) | 151.2 | 69.5 |
| | 45+270BETA(1.76, 4.06) | 126.6 | 47.5 |
| **Orthopedic** | 14.5+GAMM(17.7, 1.39) | 39.1 | 5.8 |
| | LOGN(132, 200) | 132.0 | 200.0 |
| | 40+WEIB(93.2, 1.22) | 134.3 | 71.9 |
| **Oncology** | 6.5+65BETA(1.98, 3.77) | 28.9 | 11.9 |
| | LOGN(98.1, 201) | 98.1 | 201.0 |
| | 25+EXPO(73.9) | 98.9 | 73.9 |
| **Thoracic** | 2.5+81BETA(1.71, 3.73) | 28.0 | 14.8 |
| | LOGN(79.7, 147) | 79.7 | 147.0 |
| | 70+EXPO(99.7) | 169.7 | 99.7 |
| **Urology** | 12.5+ERLA(7.93, 2) | 28.2 | 5.6 |
| | EXPO(86.2) | 86.2 | 86.2 |
| | 50+WEIB(88.4, 1.15) | 134.1 | 73.3 |
| **Gastro-Intestinal** | 3.5+65BETA(3.2, 5.57) | 27.2 | 10.0 |
| | LOGN(111, 163) | 111 | 163.0 |
| | 45+505BETA(1.72, 8.02) | 134.1 | 58.8 |
| **Plastic** | 1.5+GAMM(7.67, 3.55) | 28.7 | 9.8 |
| | 430BETA(0.538, 1.54) | 111.3 | 107.3 |
| | 30+WEIB(89.1, 1.32) | 112.0 | 62.7 |
| **Oral/dental** | 2.5+53BETA(2.99, 4.07) | 24.9 | 9.2 |
| | LOGN(81.7, 54.2) | 81.7 | 54.2 |
| | 30+WEIB(55.4, 0.903) | 88.2 | 64.5 |
| **Otolaryngology** | 9.5+WEIB(19.1, 2.04) | 26.4 | 8.7 |
| | 238BETA(2.25, 7.59) | 54.4 | 30.3 |
| | TRIA(25, 53.8, 140) | 72.9 | 24.4 |

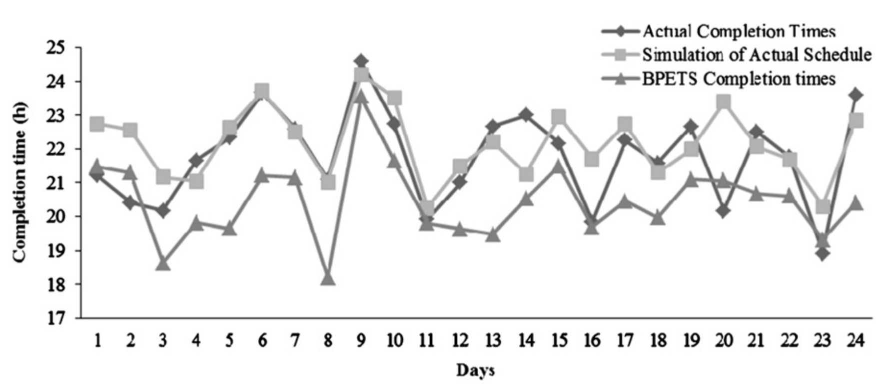


**Fig. 10.** Comparison of the completion time of BPETS with the actual schedule in case study OR department.

## References

Blake, J.T., Carter, M.W., 1997. Surgical process scheduling: a structured review. Journal of the Society for Health Systems 5 (3), 17–30.
Cardoen, B., Demeulemeester, E., Belien, J., 2010. Operating room planning and scheduling: a literature review. European Journal of Operational Research 201 (3), 921–932.
Cayirli, T., Veral, E., 2003. Outpatient scheduling in health care: a review of literature. Production and Operations Management 12 (4), 519–549.
Denton, B., Rahman, A., Nelson, H., Bailey, A., 2006. Simulation of a multiple operating room surgical suite. In: Perrone, L., Wieland, F., Liu, J., Lawson, B., Nicol, D.M., Fujimoto, R. (Eds.), Proceedings of the 2006 Winter Simulation Conference, Piscataway. Institute of Electrical and Electronics Engineers, Inc, New Jersey, pp. 414–424.
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.
Glover, F., 1989. Tabu search-part I. ORSA Journal on Computing 1 (3), 190–206.
Glover, F., Laguna, M., 1997. Tabu Search. Kluwer Academic Publishers, Boston.
Guinet, A., Chaabane, S., 2003. Operating theatre planning. International Journal of Production Economics 85 (1), 69–81.
Gül, S., Denton, B.T., Fowler, J.W., Huschka, T., 2011. Bi-criteria scheduling of surgical services for an outpatient procedure center. Production and Operations Management 20 (3), 406–417.
Hsu, V.N., de Matta, R., Lee, C.-Y., 2003. Scheduling patients in an ambulatory surgical center. Naval Research Logistics 50 (3), 218–238.
Jula, P., Leachman, R.C., 2010. Coordinated multistage scheduling of parallel batch-processing machines under multiresource constraints. Operations Research 58 (4), 933–947.
Klassen, K.J., Yoogalingam, R., 2009. Improving performance in outpatient appointment services with a simulation optimization approach. Production and Operations Management 18 (4), 447–458.
Lamiri, M., Grimaud, F., Xie, X.L., 2009. Optimization methods for a stochastic surgery planning problem. International Journal of Production Economics 120 (2), 400–410.
Lowery, J.C. and Davis, J.A. (1999) Determination of operating room requirements using simulation. Proceedings of 1999 Winter Simulation Conference, Washington DC, 2, 1568-1572.
Marcon, E., Dexter, F., 2006. Impact of surgical sequencing on post anesthesia care unit staffing. Health Care Management Science 9 (1), 87–98.
Min, D., Yih, Y., 2010. Scheduling elective surgery under uncertainty and downstream capacity constraints. European Journal of Operational Research 206 (3), 642–652.
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.
Robinson, L.W., Chen, R.R., 2003. Scheduling doctors' appointments: optimal and empirically-based heuristic policies. IIE Transactions 35 (3), 295–307.
Yellig, E., Mackulak, G.T., 1997. Robust deterministic scheduling in stochastic environments: the method of capacity hedge Points. International Journal of Production Research 35 (2), 369–379.
Zhou, J., Dexter., F., 1998. Method to assist in the scheduling of add-on surgical cases, upper prediction bounds for surgical case durations based on the log-normal distribution. Anesthesiology 89 (5), 1228–1232.