PRODUCTION AND OPERATIONS MANAGEMENT POMS
Vol. 20, No. 3, May–June 2011, pp. 406–417
ISSN 1059-1478 | EISSN 1937-5956 | 11 | 2003 | 0406
DOI 10.1111/j.1937-5956.2011.01232.x

Bi-Criteria Scheduling of Surgical Services for an Outpatient Procedure Center

Serhat Gul
School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, Arizona 85287, USA, serhat.gul@asu.edu

Brian T. Denton
Edward P. Fitts Department of Industrial and Systems Engineering, North Carolina State University, Raleigh, North Carolina 27695, USA, bdenton@ncsu.edu

John W. Fowler
School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, Arizona 85287, USA, john.fowler@asu.edu

Todd Huschka
Mayo Clinic, Division of Health Care Policy and Research, Rochester, Minnesota 55905, USA, huschka.todd@mayo.edu

Uncertainty in the duration of surgical procedures can cause long patient wait times, poor utilization of resources, and high overtime costs. We compare several heuristics for scheduling an Outpatient Procedure Center. First, a discrete event simulation model is used to evaluate how 12 different sequencing and patient appointment time-setting heuristics perform with respect to the competing criteria of expected patient waiting time and expected surgical suite overtime for a single day compared with current practice. Second, a bi-criteria genetic algorithm (GA) is used to determine if better solutions can be obtained for this single day scheduling problem. Third, we investigate the efficacy of the bi-criteria GA when surgeries are allowed to be moved to other days. We present numerical experiments based on real data from a large health care provider. Our analysis provides insight into the best scheduling heuristics, and the trade-off between patient and health care provider-based criteria. Finally, we summarize several important managerial insights based on our findings.

Key words: operating room; outpatient procedure; scheduling; simulation; genetic algorithm
History: Received: December 2008; Accepted: August 2010, after 2 revisions.

1. Introduction

Surgical services require the coordination of many activities including patient intake and preparation, the surgical procedure, and patient recovery. Designing schedules that achieve smooth patient flow is a complicated task due to the dependencies between these activities. Scheduling is further complicated by considerable uncertainty in the duration of activities. These problems are amplified for Outpatient Procedure Centers (OPCs) that typically perform a variety of elective procedures on an outpatient basis. A high volume of surgical procedures combined with significant uncertainty in the duration of activities and a fixed length of time that the surgical suite is open (typically 8–10 hours) give rise to difficult stochastic scheduling problems involving multiple, competing criteria.

The physical resources in a surgical suite include operating rooms (ORs), intake rooms, and recovery rooms, as well as equipment resources such as diagnostic devices and surgical instrument kits. There are also several human resources including surgeons, nurses, and nurse anesthetists. Surgical services occur in three major steps. The first, *intake*, starts when the patient arrives at the surgical suite to initiate his/her check-in process, and ends when the patient reaches an OR bed. The *intra-operative care* period starts when the patient is admitted to the OR area and ends when the patient is taken to a recovery bed. The surgical procedure itself is performed during this period. The last step, *recovery*, starts when the patient is admitted to a recovery area and ends when the patient is discharged. Even for very routine surgeries, the duration of each of these activities exhibits considerable variation (Berg et al. 2010).

In this article, we focus on *expected patient waiting time* and *expected surgical suite overtime*. These are among the most important performance measures that a manager (e.g., charge nurse) must consider on a daily basis. These criteria are in conflict because a schedule with small time intervals between procedures tends to have low surgical suite overtime and high patient waiting times, and vice versa. We perform a bi-criteria analysis to estimate the impact of three types of scheduling improvements and answer the following three questions:

1. What are the potential benefits of using easy-to-implement heuristics for daily appointment scheduling?
2. What are the potential benefits of optimization methods over commonly used and easy-to-implement heuristics for daily appointment scheduling?
3. What are the potential benefits of controlling daily procedure mix from day to day?

An OPC at Mayo Clinic, in Rochester, Minnesota, forms the testbed for our study. We first construct a discrete event simulation (DES) model and use it to evaluate easy-to-implement scheduling heuristics based on expected patient waiting time and expected surgical suite overtime. Our DES is a comprehensive model that includes all three major surgical service steps. Next, we embed the simulation model within a hybrid solution method that contains both a bi-criteria genetic algorithm (GA) and appointment time-setting heuristics to construct a (near) Pareto optimal set of schedules (i.e., the non-dominated set of solutions with respect to the two criteria). We further use the GA to examine the potential benefits of controlling the daily surgical mix.

The remainder of the paper is organized as follows. In the next section, we provide some background on OPCs. In section 3, we present a brief literature review of relevant studies. In section 4, we describe our simulation model. In section 5, we discuss the methodologies we have applied including the scheduling heuristics and our GA. In section 6, we present experimental results. Finally, we summarize the most significant managerial insights in section 7.

2. Background on OPCs

OPCs are complex systems, often with several surgical groups (e.g., departments or subgroups within departments) sharing resources on a given day. The layout of a typical suite is illustrated in Figure 1. The physical space used for patient care can be broken into three sections. The first is the patient waiting area, the second is the pre-/post-room area (used for patient intake and recovery), and the third is the OR area.

Typically there is some dedication of intake, operating and recovery rooms to surgical groups. For example, in the OPC we studied, ORs are dedicated as follows: *Pain Medicine* has one OR, each of *Urology* and *Ophthalmology* has two ORs, and *Oral Maxillofacial* (OMS) has three ORs. Thus there are eight ORs in total, which are shared by the three surgical groups. There are 20 pre-/post-rooms, four of which are dedicated to Pain Medicine. Oral Maxillofacial also has four dedicated pre-/post-rooms, but the remaining 12 pre-/post-rooms can be utilized by any one of the surgical cases of the other groups.

The OPC depicted in Figure 1 combines resources by using the same set of rooms for intake and recovery. This increasingly common layout is motivated by the desire to balance resources and reduce congestion (because intake areas tend to be heavily utilized early in the day while recovery areas are empty and vice versa at the end of the day). Patients first go to the check-in desk, and then to the patient waiting area, where they wait for an intake room to become available. After the intake process, they wait for their surgeon and OR to become available. Once the procedure is complete, they reenter the pre-/post-room area to recover, and exit the OPC when their recovery is complete.

There is significant uncertainty in the time necessary for completing activities in the OPC. In Figure 2, empirical estimates of probability density functions are plotted for intake, surgical procedure, and recovery, for procedures from the same surgical group. Surgical procedure durations can differ considerably among procedures even within the same surgical group and they tend to have a long tail, which represents unpredictable low probability complications that may occur during the procedure. Intake and recovery distributions are generally quite similar within a surgical group. Intake distributions are similar, because patients are going through similar intake processes. Recovery distributions also do not differ, as procedures within a surgical group tend to use similar levels of anesthetic.

The particular OPC we consider opens at 8 AM, which is the scheduled time of the first patient's arrival. The planned closure time is 5 PM. Overtime results in additional costs for those staff that stay beyond 5 PM. There is also a loss of goodwill on the part of staff because most staff members prefer not to work overtime. Furthermore, we have anecdotal evidence that long patient wait times, which lead to unhappy patients, reduce staff morale and can lead to turnover, particularly among nurses.

We use the process flow defined above, and the probability density functions for intake, surgical procedure, recovery, and other activity times, to construct our DES model, which we describe in detail in section 4.

3. Literature Review

Following is a brief literature review that covers several examples from the literature that are related to

Figure 1 Layout and Patient Flow Through an Outpatient Procedure Center Including the Patient Waiting Area, Pre-/Post-Rooms, and Operating Rooms

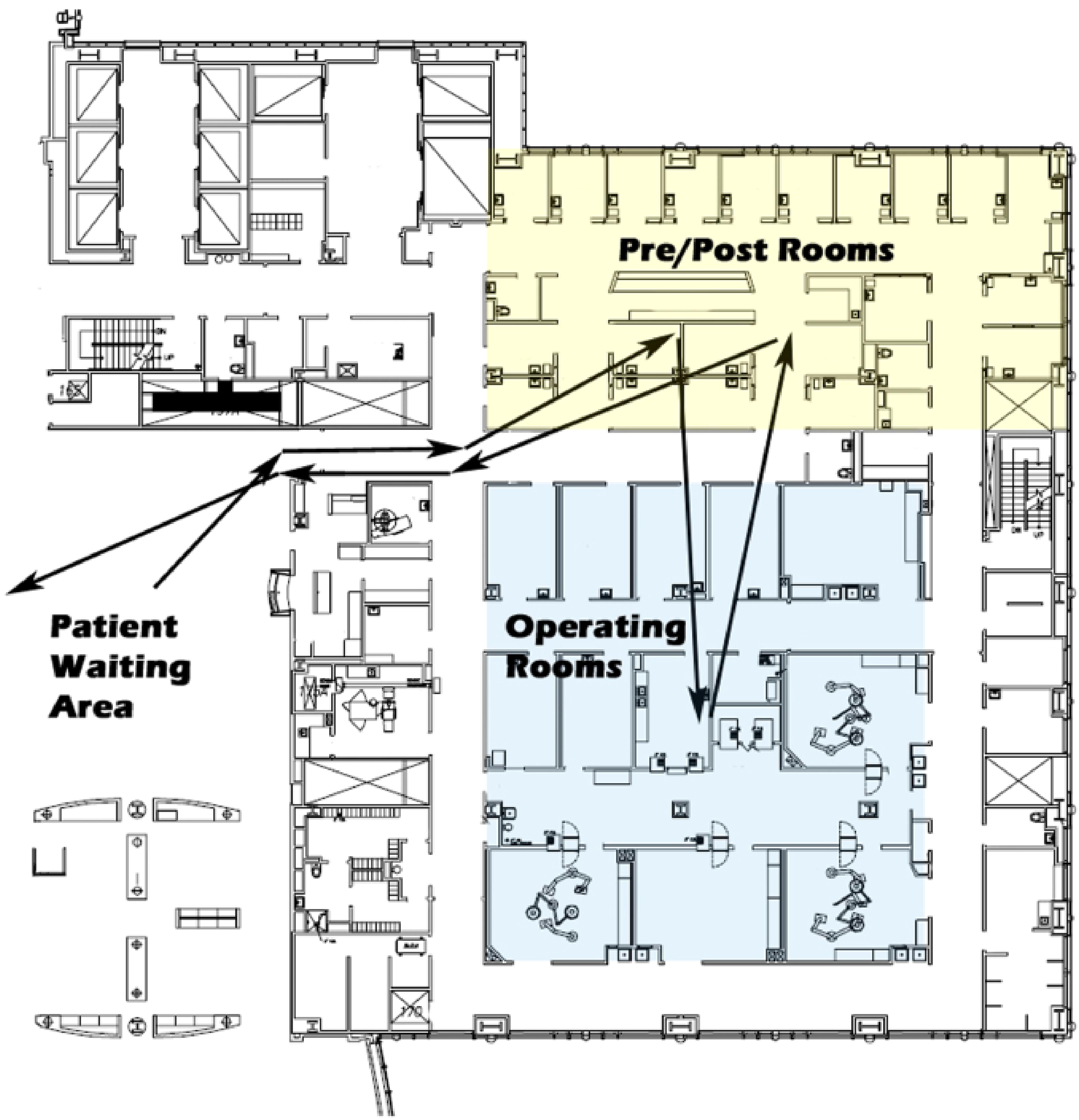
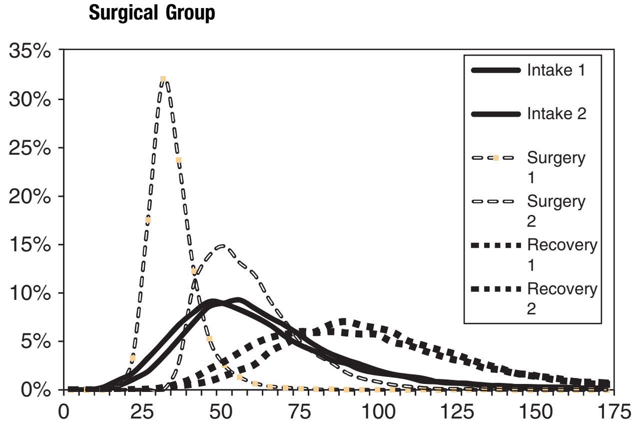


Figure 2 Probability Density Functions for Intake, Procedure, and Recovery Times for Two Different Types of Surgical Procedures within a Surgical Group



our work. We focus on studies that either (a) evaluate scheduling heuristics for multiple ORs using a DES model or (b) consider resources in addition to ORs (e.g., recovery area resources) or (c) analyze multi-criteria problems related to planning and scheduling. For a more extensive review of the literature on surgery planning and scheduling, the reader is referred to Magerlein and Martin (1978), Blake and Carter (1997), Gupta (2007), Gupta and Denton (2008), Cardoen et al. (2010), and May et al. (2011).

Dexter et al. (1999b) use simulation to test heuristics for allocating block time to surgeons, and schedule elective cases to maximize OR utilization. They evaluate four on-line bin packing algorithms to schedule elective cases: next fit, first fit, best fit, and worst fit. Dexter et al. (1999a) evaluate 10 different algorithms (on-line, off-line, and hybrid algorithms) for scheduling add-on cases into the open OR time available to evaluate their effectiveness in increasing OR suite utilization. Testi et al. (2007) use simulation to evaluate different surgery sequences with regard to the longest waiting time of the surgeries in the waiting list, longest processing time (LPT), and shortest processing time (SPT) after building the master surgery schedule.

Dexter and Marcon (2006) studied the impact of several different surgery sequencing heuristics on workload of a post-anesthesia care unit (PACU) including: random sequence, longest cases first (LCF), shortest cases first, Johnson's rule, and several others. The authors analyzed how sequencing affects OR over-utilization, PACU completion time, delays in discharging from the OR into PACU, and the maximum number of patients in the PACU throughout the day. They found that even though LCF is the most popular rule used in practice, it is one of the worst rules with regard to the performance measures of the study. Random sequencing is suggested if it is difficult to implement rules that performed better, due to the constraints (such as medical and equipment) that are not considered in the study, because implementation of random sequencing is trivial and it yields medium level results.

Berg et al. (2010) use a DES model to analyze an endoscopy suite with respect to surgeon-to-OR allocation scenarios. Competing performance measures such as overtime for the endoscopy suite and patient waiting time were analyzed in the model and a simulated annealing heuristic was used to improve the scheduled start time of cases with respect to expected overtime and patient waiting time. An endoscopy suite is a simplification of a general OPC because the case mix is limited to only upper and lower endoscopies. The suite considered in Berg et al. (2010) consists of three independent process areas (i.e., intake, procedure, recovery) and the authors assume that the capacities of intake and recovery areas are unlimited. In contrast we assume intake and recovery have fixed capacity and potentially limit patient flow through the suite. Finally, the authors use only a very simple simulated annealing approach to design schedules, whereas we provide a detailed comparison of standard heuristics as well as a more advanced bi-criteria GA. Lehtonen et al. (2007) built a simulation model to analyze the effect of six process interventions on open-heart surgery with respect to OR productivity and overtime amount.

Price et al. (2011) studied the problem of allocation of the surgical block times to days and surgical groups. The objective of their mixed integer programming (MIP) model is to balance the flow into the intensive care unit (ICU) with the flow out of the ICU. They provide evidence that their model achieves the goal of reducing overnight stays in the PACU, which occur due to congestion in the ICU. Chow et al. (2011) used a combination of Monte Carlo simulation and MIP models to build surgery schedules that reduce variation in bed occupancy in surgical wards. In their models, (1) block times for surgeries are scheduled into future, and (2) surgical mix within each block is determined. Marcon et al. (2003) simulate a surgical suite to estimate the number of PACU beds required. They also investigated the effect of a decrease in the number of porters (patient escorts) in the OR on the number of PACU beds needed. Lowery and Davis (1999) used a simulation tool to study the effect of decreasing the number of ORs in a hospital. They analyzed the effects of changes in the surgery schedule and in case times on the number of rooms required. Tyler et al. (2003) simulate an OR to determine the optimum OR utilization and analyze the important factors such as average patient waiting time and variability of case durations that impact OR utilization. Lowery (1992) uses a simulation model to simulate the patient flow through critical care units to determine the number of beds required.

Multi-criteria studies related to surgery planning and scheduling include the following. Jebali et al. (2006) developed a two-phase approach to solve the surgery assignment and sequencing problem formulated as an integer program. In their approach, operations are first assigned to ORs with the objective of minimizing hospitalization, undertime and overtime costs. Second, optimal sequences are sought for minimizing the total overtime cost for ORs. Guinet and Chaabane (2003) solved the weekly patient-to-OR assignment problem using a primal-dual heuristic. Patient satisfaction and resource efficiency are considered in this study where the objective includes the minimization of the number of days patients wait in the hospital and the overtime. Lamiri et al. (2008) proposed a stochastic programming model for the assignment of elective surgeries to ORs over a planning horizon. Uncertainty comes from the demand for emergent cases in this formulation. The study aims to minimize both OR utilization costs and patient-related costs. They solve the problem using a column generation method.

In the context of ambulatory care services, Cayirli et al. (2006) tested several sequencing and appointment rules for clinic visits using simulation with regards to patient waiting time, doctor idle time, and overtime. The most significant finding of this study is that the impact of sequencing on the criteria is more important than that of the appointment rule. Lovejoy and Li (2002) consider an OR capacity expansion problem. They focus on the trade-off between waiting time, procedure start time reliability, and hospital revenues.

Our work differs from the aforementioned papers in the following ways. First, we propose a hybrid solution technique mixing a bi-criteria GA with appointment time-setting heuristics to find the (near) Pareto optimal set of schedules and reveal the trade-off between factors affecting both the patient and the provider. Second, we test several commonly used scheduling heuristics against our GA to estimate the potential benefits of optimization-based methods for scheduling system improvements. Finally, we use our GA to estimate the potential benefits of optimizing daily procedure mix.

4. Simulation Model

Our DES model was developed based on an OPC in Rochester, MN (Huschka et al. 2007). It is a *terminating simulation* (Banks et al. 2005), in the sense that a finite number of procedures are scheduled each day within a pre-determined time in which the OPC is open each day. Patients arrive into the check-in area according to a deterministic schedule (constructed using one of the heuristics we discuss in section 5). We assume arrivals are on time and all patients show up for their scheduled procedure (extensions such as tardiness and no-shows are straightforward with our model; however, they are uncommon in the OPC we studied, and for simplicity we do not include them in our analysis). Subject to pre-/post-room and surgeon/OR availability, patients proceed through the OPC with activity start and completion times based on samples from the continuous probability density functions of Tables 1 and 2.

The number of surgeons per surgical group on a given day is equal to the number of ORs allocated to the group and surgeons may operate in any OR assigned to their group. While these policies are not necessarily in place in all OPCs, they are reasonably common, and representative of scheduling problems faced in practice.

We used data from the year 2006 for 4034 patients at Mayo Clinic (corresponding to the operations of the first 21 weeks of the year). Probability density functions were fit for all stages of a patient's movement through the surgical suite including intake, surgical procedure, and recovery (see Table 1 for a summary of data). We partitioned the procedure times into three parts (*pre-incision*, *incision*, and *post-incision times*) and fit distributions for each independently. This was necessary because these activities require different resources. For instance, the OR is utilized the entire time, but surgeons do not need to take part in the pre-incision and post-incision activities.

Distributions were fit separately for each surgical procedure type. We used the log-normal distribution for procedure times because it yielded a best fit based on maximum likelihood estimation and because it is commonly used in the literature (see, for example, Zhou and Dexter 1998). For intake and recovery we found Erlang, gamma, beta, Weibull, and exponential distributions were the most common best fit. OR turnover and transfer times were estimated by triangular distributions based on expert estimates of the minimum, mean, and maximum times (see Table 2).

Our validation is based on a comparison of model outputs such as the number of surgeries completed per day and expected daily overtime estimates with similar values from the particular outpatient procedure practice at Mayo Clinic in Rochester, MN (i.e., the baseline schedule). The results based on the model were also presented to experts at Mayo Clinic familiar with the system including an operations research analyst specializing in surgery in the Division of Health Care Policy and Research, an administrator for the surgical practice, and the group of nurses that work within the unit.

5. Methodology

We use our DES model to compare easy-to-implement heuristics used in practice with a GA-based heuristic on the basis of total expected patient waiting time and expected surgical suite overtime. Overtime is the difference between the time the last patient completes recovery and 5 PM (if it is non-negative). Total patient waiting time is the sum of the times a patient spends waiting for a pre-/post-room to initiate intake and waiting for an OR to begin the surgical procedure. As an aggregate measure, we calculate the average of the expected patient waiting times over all patients served across all days.

In section 5.1 we describe several combinations of sequencing and appointment time heuristics for selecting the schedule of patient arrivals to the check-in area of the OPC. In section 5.2 we discuss our GA-based approach.

5.1. Heuristics

To answer question 1 of section 1, we test several combinations of patient sequencing and appointment time heuristics. We sequence cases of each OR and day combination according to four different sequencing rules: increasing mean of procedure time (SPT), decreasing mean of procedure time (LPT), increasing variance of procedure time (VAR), and increasing coefficient of variation of procedure time (COV).

Given a specified sequence of patients, the first appointment is set to the beginning of the day, and subsequent appointments are set to the prior appointment time plus the estimated time for the previous patients' procedure. The estimate of the procedure time influences the patient waiting time and overtime. If the estimate is too large, it may lead to unnecessary overtime; if it is too low it may result in unnecessary patient waiting time. To explore this trade-off, we estimate the time using various percentiles of the

Table 1 Mean, Standard Deviations (in Minutes), and Distributions of the Intake, Procedure, and Recovery Times for Various Procedure Groups of the Surgical Groups Are Listed with the Number of Patients Data Used to Calculate Them

Surgical group	Procedure group	Process	Mean	Standard deviation	Number of operations	Distribution fit
Oral Maxillofacial procedure	1	Intake	42.02	21.92	1472	Weibull
		Procedure	33	19.11	1472	Lognormal
		Recovery	53.02	33.88	1472	Gamma
	2	Intake	0	0	0	—
		Procedure	36	33.88	1919	Lognormal
		Recovery	0	0	0	—
Pain Medicine	1	Intake	38.4	20.22	58	Erlang
		Procedure	19.78	12.12	58	Lognormal
		Recovery	21.09	9.74	58	Weibull
	2	Intake	38.72	24.37	244	Gamma
		Procedure	20.49	10.86	244	Lognormal
		Recovery	23.64	16.65	244	Erlang
	3	Intake	34.7	21.11	1551	Gamma
		Procedure	20.93	15.08	1551	Lognormal
		Recovery	19.94	14.17	1551	Erlang
	4	Intake	32.79	16.79	24	Triangular
		Procedure	40.5	26.12	24	Lognormal
		Recovery	52.58	29.93	24	Weibull
	5	Intake	36.46	21.47	970	Gamma
		Procedure	34.01	17.42	970	Lognormal
		Recovery	23.26	15.84	970	Beta
Ophthalmology	1	Intake	65.58	26.32	1696	Gamma
		Procedure	41.63	16.43	1696	Lognormal
		Recovery	29.84	14.56	1696	Weibull
	2	Intake	65.65	28.57	589	Triangular
		Procedure	77.66	44.03	589	Lognormal
		Recovery	42.75	26.9	589	Erlang
Urology	1	Intake	64.92	27.59	329	Weibull
		Procedure	53.3	27.7	329	Lognormal
		Recovery	89.33	39.18	329	Gamma
	2	Intake	58.14	26.56	640	Gamma
		Procedure	31.3	16.37	640	Lognormal
		Recovery	94.23	36	640	Erlang
	3	Intake	64.15	22.78	153	Beta
		Procedure	138.16	56.77	153	Lognormal
		Recovery	126.95	49.55	153	Weibull
	4	Intake	61.37	25.18	345	Erlang
		Procedure	55.78	22.89	345	Lognormal
		Recovery	99.91	33.13	345	Beta
	5	Intake	58.18	26.68	496	Gamma
		Procedure	80.33	43.76	496	Lognormal
		Recovery	96.56	44.97	496	Weibull

The procedures within a surgical group (e.g., Urology) were grouped manually based off the procedure codes by the Outpatient Procedure Center staff.

distribution. Appointment times are determined by the following recursion:

$$A_{i+1} = A_i + h_i,\ i = 2, \ldots, n,$$

where $A_1 = 0$, and h_i is the percentile of procedure i duration. This is known in the literature as *job hedging* (Yellig and Mackulak 1997) and it has been investigated extensively in the context of OR and single

Table 2 Distributions and their Parameters Set Subjectively by the Experts for the Transfer Times Between Units as Well as the Turnover Times for Different Rooms Are Listed

Transfer times	
Patient flow (from—to)	**Distribution (minimum, mean, maximum)**
Check-in desk—waiting area	Triangular (5, 6, 7)
Waiting area—pre-/post-room	Triangular (2, 3, 4)
Pre-/post-room—OR	Constant (2)
OR—pre-/post-room	Triangular (1, 2, 2)
Turnover times	
Room type	**Distribution**
Pain Medicine OR	Triangular (2, 3, 8)
Other ORs	Triangular (5, 6.5, 8)
Pre-/post-rooms	Triangular (5, 6.5, 8)

All parameter values are in minutes.
OR, operating room.

server appointment scheduling (for example, see Charnetski 1984; Ho and Lau 1992; Weiss 1990).

5.2. A Bi-Criteria GA

To answer questions 2 and 3 from section 1 we solve two different models using a GA. The first (model A) assumes the daily procedure mix each day is fixed based on a pre-defined schedule. The second (model B) assumes the daily procedure mix may be modified by rescheduling procedures among days within a time window of n days ($n = 1$ and $n > 1$ for models A and B, respectively). The remainder of this section provides a brief summary of our GA (more complete details are presented in Appendix A).

A GA is a local search algorithm based on the biological evolution paradigm (Holland 1975). An initial population is created and genetic operators are used to search the neighborhood of the initial population through successive improving iterations. At each iteration, a selection is made based on the *survival of the fittest* rule to determine the members of the next generation. This mechanism continues until a stopping criterion is met (e.g., after a fixed number of iterations, or if the solution is not sufficiently improved after a certain number of iterations).

Members of the population are called *chromosomes* and each chromosome represents a solution (in our context a solution is a surgery schedule). The chromosome stores the job hedging level, day, and known attributes of a procedure, i.e., type and the surgeon for each procedure.

The algorithm starts with an initial set of solutions (note that we use the term solution and chromosome interchangeably), which are generated as follows. One of the solutions in the initial population is the actual schedule used at the OPC in the year 2006. The rest of the solutions are created using a combination of the following techniques: (i) scheduling based on the heuristics described in section 5.1 and (ii) randomly assigning procedures to time slots available within the n days of time window at the actual schedule.

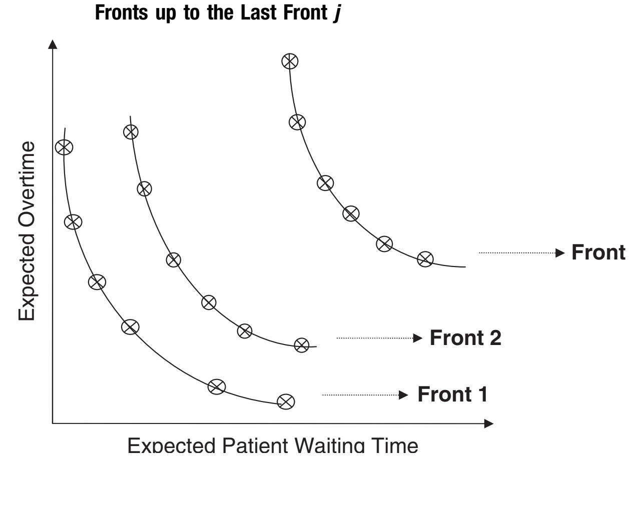
At each iteration, we evaluate solutions using the DES model and store the expected patient waiting time and expected surgical suite overtime. We rank solutions based on these two criteria. Our approach is based on the non-dominated sorting genetic algorithm II proposed by Deb et al. (2000) and is illustrated in Figure 3. The non-dominated solutions, i.e., the (near) Pareto optimal set, are assigned to the *first front*. We then compare the remaining solutions and assign the non-dominated ones with the *second front*. Using this approach, we determine the fronts of all the solutions in the population and solutions are ranked based on their associated front. Solutions on the same front are further prioritized using a c*rowding distance operator* (described in Appendix A) to diversify the solution set along a given front.

To create the next generation, pairs of solutions are selected based on the ranking and combined via a crossover operator to create new pairs of solutions. We also apply a mutation operator to create near neighbors of current solutions. Repeating the same steps a fixed number of times, a new solution set is constructed at each iteration of the GA. After a defined number of iterations are completed, the algorithm terminates and the solutions on the (near) Pareto optimal set (first front) are stored as the output.

6. OPC Case Study

Preliminary experiments were performed in which the number of simulation replications was varied to

Figure 3 Front 1 is the Set of Points Including (Near) Pareto Optimal Set of Solutions. In Case the Set of Points on Front 1 Are Deleted, then Front 2 Becomes the (Near) Pareto Optimal Set of the Remaining Solutions. The Same Rule Is Used Iteratively for the Rest of the Fronts up to the Last Front *j*



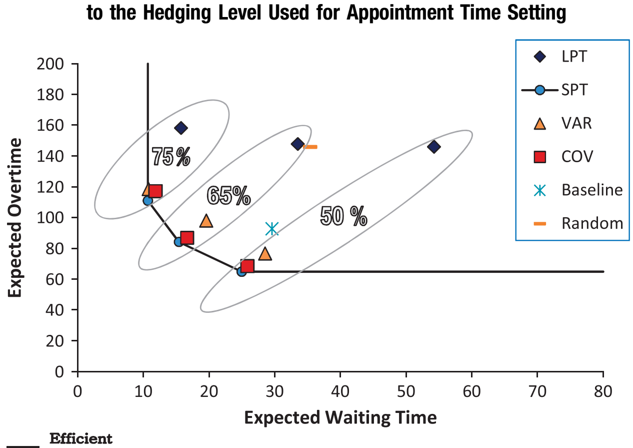
see how many were needed to obtain a satisfactory trade-off between computation time and half width of the generated confidence intervals. Based on these experiments, the results below include 20 simulation replications in the evaluation of each solution.

6.1. Analysis of Simple Heuristics

We analyzed combinations of four different sequencing heuristics (LPT, SPT, VAR, COV) with various hedging levels. Expected patient waiting time and expected surgical suite overtimes are estimated for each sequencing and scheduling heuristic combination. Figure 4 illustrates our results for 12 heuristics and 50%, 65%, 75% indicate the hedging (percentile) levels. The result for the baseline schedule as well as the result for a random schedule generated by randomly assigning procedures to the time slots available in the day of procedure are also plotted to serve as reference points. We calculated 95% *confidence intervals* for each of the criteria of the 12 heuristics, baseline schedule, and the random schedule and found them to be approximately 2% of the mean values.

Figure 4 provides several important insights. First, the baseline schedule is in the dominated set. Second, expected patient waiting time is very sensitive to the choice of percentile used for hedging. As the percentile increases the expected patient waiting time drops while the expected surgical suite overtime increases. Also, the trade-off between improvements in expected patient waiting time and expected overtime depends on the specific sequencing heuristic used to create an ordered list of surgeries. Third, among the four sequencing heuristics, SPT performs the best as it is always on the efficient frontier, while VAR and COV appear in the vicinity of the frontier. It is intuitive that there is not a considerable difference between the performance measure values from the SPT and VAR rules due to the fact that there is a positive correlation between mean and standard deviations of the procedure durations within a surgical group (see Table 1). Because of the correlation, the two procedure lists sequenced according to increasing mean and increasing variance are generally similar, and hence would yield indifferent criteria values. The correlation between these parameters is the reason for considering coefficient of variation as one of the reference for the sequencing heuristics; however, the COV heuristic is outperformed by SPT. Finally, the LPT heuristic generally performs poorly and is dominated by the other heuristics. This result supports the findings of Dexter and Marcon (2006) who found that LCF, while being the most popular rule used in practice, is one of the worst rules they considered with regard to the criteria of their study (see section 3 for more details). We find using LPT for sequencing, and 50th percentile for appointment time-setting heuristic, creates a schedule performing even worse than a random schedule. Intuitively, this seems to stem from the fact that LPT schedules procedures with higher variability first (due to the correlation between mean and standard deviations), which negatively affects the schedule later in the day, causing higher expected patient waiting time and expected surgical suite overtime (for a similar conclusion for a single OR case, see Denton et al. 2007).

Figure 4 **Expected Values (in Terms of Minutes) for the Resulting Criteria for All Heuristics, a Random Schedule, and the Baseline Schedule. Note that Longest Processing Time (LPT) Sequences According to Increasing Mean, Shortest Processing Time (SPT) Decreasing Mean, VAR Increasing Variance, and COV Increasing Coefficient of Variation of Procedure Durations. Results Are Clustered According to the Hedging Level Used for Appointment Time Setting**



The most notable finding of this section is the following: *Among the sequencing heuristics, SPT yields the best schedules; while the best choice for a job hedging level depends on the heuristic used for sequencing the surgeries.*

6.2. Optimization-Based Improvements to Simple Heuristics

Using the same data, we test our GA-based approach in two different contexts. First, we apply the GA to the daily procedure lists assuming the procedure day is fixed (model A). Based on preliminary numerical experiments, we chose the number of solutions in a population to be 40, and the number of generations to be 50. We use combinations of sequencing (SPT, LPT, VAR, COV) and time-setting heuristics (50, 55, 60, 65, 70, 75, 80, 85th percentiles) to provide 32 different initial solutions. The baseline schedule is also used as one of the initial solutions. The remaining seven solutions are generated by randomly assigning the procedures to the time slots available in the surgery schedule in the same day.

In Figure 5, we compare the GA solutions with the only solutions located on the efficient frontier of heuristics revealed in section 5.1 (see Figure 4). We observe

Figure 5 Comparison of the Genetic Algorithm (GA) Solutions (for 80, 70, 65, 60th Percentiles) with the Shortest Processing Time (SPT) Solutions (for 75, 65 and 50th Percentiles from Left to the Right, Respectively), the Only Solutions on the Efficient Frontier of Heuristics. Note that the Unit Values Are in Minutes

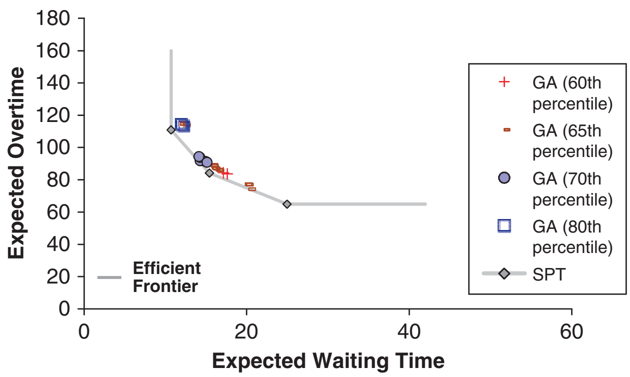
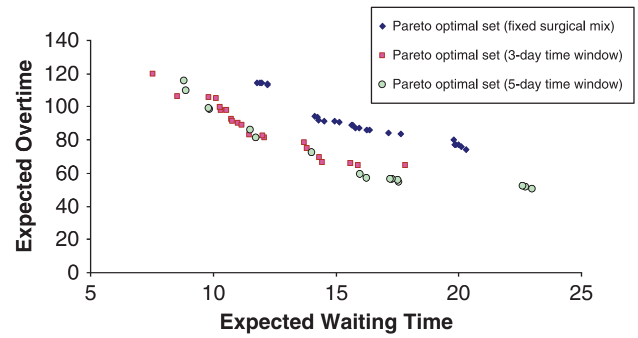


Figure 6 Comparison of Solution Values in Minutes for Different (Near) Pareto Optimal Set of Solutions of Genetic Algorithm for Different Configurations: (1) Fixed Surgical Mix, (2) Varying Surgical Mix with 3-Day Time Window, and (3) Varying Surgical Mix with 5-Day Time Window



that the (near) Pareto optimal set of solutions for the combination of the methods includes some GA solutions and all heuristics that use SPT as the sequencing heuristic. This indicates that the GA does not help us to improve the efficient solutions found by simple heuristics when the solution space is constrained by fixing the day of procedures. As SPT is easy to implement in practice, it is more advantageous for surgical suite managers compared with the GA that requires computational resources to yield a solution.

Figure 5 also indicates the distribution of the hedging levels used for the (near) Pareto optimal set of solutions. There are 23 efficient GA solutions plotted on Figure 5 and of all, the majority (56%) use the hedging level corresponding to the 65th percentile, while 21% utilize the 70th, 13% 80th, and 8% 60th percentiles. As it is used in the majority of the schedules on the (near) Pareto optimal set, and also provides a reasonable trade-off between expected patient waiting time and expected surgical suite overtime, the 65th percentile of the procedure time distributions seems to be a proper choice as the amount of time to allocate to procedures. On the other hand, expected surgical suite overtime values are found to be more than 1 hour for the other efficient schedules revealed. This would also direct managers toward the selection of 65th percentile. Another insight that the graph yields is that schedules having the same hedging value generally appear in regions close to each other in criteria space. This further supports the observation that the job hedging parameter has a significant effect on both criteria.

The most significant finding in this section is: *The performance of SPT-based heuristics is similar to performance of the GA when the day of the procedure is fixed. Because it is much easier to implement in practice, SPT-based heuristics are recommended over the GA.*

6.3. Optimization of Daily Procedure Mix

To answer the third research question we defined in section 1, we relax the requirement that daily mix be fixed (model B). This model provides more flexibility because the procedures are allowed to be assigned to any day within an n-day time window. We define the time windows as mutually exclusive windows (i.e., the days from 1 to n belong to one window, while the days from $[n+1]$ to $[2n]$ belong to a different window), so we shift the days of surgeries back and forth while fixing the time window they belong to. In our experiments, we tested $n = 3$ and $n = 5$. In the case of $n = 3$, for example, if the original day of the procedure was Wednesday of the first week, then it can be reassigned to Monday, Tuesday, or Wednesday of the first week. On the other hand, if the procedure day was originally set as Friday of the first week, then it can be moved to Thursday or Friday of the first week, or Monday of the second week. The solution space for $n = 5$ corresponds to allowing procedures to be moved within a given week (this is reasonable because procedures scheduled in the OPC are elective). Furthermore, it is consistent with some surgery scheduling practices where scheduling is executed in two steps; first by setting the week of surgery, and afterwards setting the specific times (Gupta 2007).

Figure 6 compares the (near) Pareto optimal sets of GA solutions for $n = 1$, 3, 5. Figure 6 illustrates that reorganizing procedures among days (e.g., $n = 3$ or 5) considerably improves the two criteria. The main reason for the realization of such an improvement is that the variation of the surgical load among days is better balanced in schedules obtained this way. Besides, the shares of procedure groups using an OR in a given day are now better set due to the flexibility of modifying procedure days. When the procedure mixes among days can be varied, some surgeries that would otherwise have induced overtime can then be assigned to another day where the OR utilization is lower. In Figure 6, we observe similarity between the (near) Pareto optimal sets for $n = 3$ and $n = 5$, i.e., two sets are very close to each other. This indicates the 3-day time window is sufficient to balance the surgical load among days.

The most essential finding to be re-emphasized is that *controlling surgical mixes among days may help achieve significant improvements in expected patient waiting time and expected surgical suite overtime; a time window of 3 days appears to be sufficient to achieve the benefits*.

7. Conclusions

OPCs require the coordination of many activities, including patient check-in, intake, surgical procedure, and recovery. In this article, we first develop easy-to-implement heuristics for scheduling of an OPC at a large medical center. We then compare the performance of these heuristics with a GA-based approach. We also illustrate the impact of varying the surgical mix among days using the GA. Following are the most significant general insights of our study:

- Simple heuristics can improve actual schedules used in practice for an OPC. Job hedging may be used to decrease patient waiting times at the expense of increasing surgical suite overtime. Furthermore, the level of trade-off between the patient waiting time and surgical suite overtime due to the increase in job hedging level varies as the heuristic used for sequencing the surgeries changes. Among the sequencing heuristics, LPT causes high expected overtime, and should be avoided, while SPT (first) performs quite well.
- Expending greater computational effort with a more sophisticated GA-based method under a restricted environment (no control over daily procedure mix) does not achieve substantial additional improvements. Owing to its easy-to-implement nature SPT should be favored over the GA.
- Controlling daily procedure mix may achieve substantial improvements in performance, though there are diminishing returns as the time window for moving surgeries is increased.

In this paper, we evaluate the schedules using a comprehensive model of an OPC and analyze the patient flow through the units (i.e., intake rooms, ORs, recovery rooms). However, as ORs are the major bottlenecks in our model, we consider only the durations of the surgical procedures and do not explicitly consider the other resources (e.g., mobile and specialized equipment, materials, nurses, nurse anesthetists, and other human resources) while designing the surgery schedules. As a future research direction, we are planning to examine the potential benefits of more complicated scheduling techniques considering the impact of other resource types into the schedule efficiency.

Acknowledgments
The first and third authors were partially supported by National Science Foundation (NSF) grant DMI-0620504 and the second author by NSF CMMI-0807144. We are grateful to the staff of the Mayo Clinic OPC who was very helpful in supplying us with information for our model.

Appendix A

The following provides additional information about our bi-criteria GA. In the first section, we provide pseudocode for the GA and in the second we provide specific details about various aspects of the GA.

A.1. Pseudocode

A.1.1. Parameters.

t = generation counter
i = chromosome index
G = number of generations
N = number of chromosomes in a generation
P_t = parent population in generation t
O_t = offspring population in generation t
C_t = pool of chromosomes in generation t
F_i = front value for chromosome i
CD_i = crowding distance value of chromosome i

Step 0: Set generation number t as 0. Form initial population P_0 having size N and set it as the current pool of chromosomes (C_0).

Step 1: Simulate chromosomes (surgery schedules). Take the two criteria values (expected patient waiting time and expected surgical suite overtime) as the returned parameter values. If $t = 0$, then skip step 2.

Step 2: Combine parent (P_t) and offspring (O_t) population to update the current pool (C_t).

Step 3: Rank each chromosome i in C_t based on the front they belong to (F_i) and their crowding distance (CD_i).

Step 4: Eliminate the poorest N chromosomes of C_t and hence leave the best N chromosomes of the current pool.

Step 5: Use binary selection tournament operator to select two candidate chromosomes from the current pool to generate a chromosome for the next generation.

Step 6: Apply crossover using the two chromosomes to generate offspring. If the GA model is A, then there is no need for resetting the day, skip step 7. Otherwise, go to step 7.

Step 7: Set the days of procedures by considering daily capacity thresholds set for each OR.

Step 8: Set the patient appointment times that are the key attributes of genes in the chromosomes using the time-setting heuristic type associated with the chromosome.

Step 9: Apply mutation by changing the orders of two random procedures selected from the surgery schedules. Increment generation number t and set the resulting population as O_t (offspring population). If $t > 1$, O_{t-1} becomes P_t. Otherwise (at the first iteration), P_0 is set as P_t.

Step 10: Check if the limit on the number of generations is reached (stopping criterion). If yes ($t \geq G$), terminate. Otherwise ($t < G$), go to step 1.

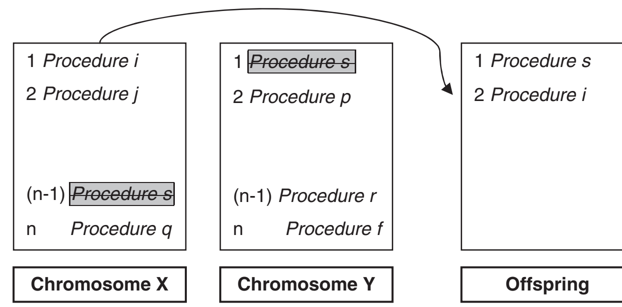
A.2. GA Operators

A.2.1. Selection. We sort chromosomes in the pool to have a lexicographical order of chromosomes according to the *front value* (has higher importance) and *crowding distance value* (see below). Then we eliminate the last N chromosomes in the sorted list to leave the N best chromosomes in the pool. We use the *binary selection tournament* method (Brindle 1981) to select mating chromosomes from the pool. The *binary selection tournament* operator works as follows: two chromosomes are selected randomly and compared with each other with respect to the front values. The crowding distance value is used as a tie breaker of the competition. The one that wins the tournament attends the crossover operation as one of the mating chromosomes. The other mating chromosome is also selected by applying the operator once again.

A.2.2. Crowding Distance. Chromosomes are ranked based on the front they appear on as well as a *crowding distance operator* (Deb et al. 2000). The crowding distance operator encourages diversity in the solutions with respect to the (near) Pareto optimal set to avoid generating a large number of solutions with similar expected patient waiting time and expected surgical suite overtime values.

A.2.3. Crossover. After selecting mating chromosomes, *uniform crossover* (Syswerda 1989) is applied to generate N offspring for the next generation. Crossover determines the order of procedures in a schedule as well as the *job hedging* level that would be used later in order to set appointment times. We apply the crossover operation independently for each procedure list of n-days to sequence procedures and then we combine the resulting independent partial sequences to have a full sequence. An illustration of the uniform crossover for determining the order of the procedures in the procedure list in the offspring can be seen in Figure 7.

Figure 7 Two Chromosomes (*X* and *Y*) Are Selected for Uniform Crossover to Determine the Sequence of a Surgical Procedure List. The First Procedure, *s*, Has Already Been Moved from *Y* to the Offspring in the Previous Iteration. Procedure *s* Was then Deleted from Both Chromosomes. At this Iteration, Chromosome *X* Is Selected for Determining the Next Surgical Procedure of the Offspring. Now, it Is Time to Insert the Second Procedure, *i*, which Is on the Top of the Remaining Procedure String of Chromosome *X*



A.2.4. Schedule Construction Using Heuristics. For model A, we directly apply the patient appointment time-setting method as the procedure day is kept fixed there. For model B, where we examine the change in a daily procedure mix, we first set the days of the procedures for each list independently. Following this, we set the appointment time of each patient for each OR and day combination. Day-setting method for model B is described first.

A.2.5. Procedure Day Setting. For each of n-days, we determine the surgical procedure list in each OR independently. We assign procedures iteratively to daily lists. To control the number of procedures in a daily list, we set a daily capacity that the OR can serve each day and therefore set a capacity *threshold* to prevent the method from leading to extreme values of overtime. We set the average daily workload for a surgical department during the study period as the threshold (see Table 3). These thresholds serve as an overtime control parameter in the study, i.e., the estimated duration of the procedures (the sum of the mean durations) is not permitted to exceed this threshold.

A.2.6. Mutation. Following the sequencing and appointment time-setting methods, we use a swap mutation operator by changing the orders of two randomly chosen procedures in the surgery schedules. The purpose of applying mutation is to avoid local minima or help sustain the evolution process by favoring further diversity among chromosomes.

Table 3 Daily Surgical Load Capacity Allocated for an Operating Room (OR) in Terms of Minutes for Different Departments Are Listed

Surgical department	Capacity (in minutes)
Oral maxillofacial procedure	480
Pain medicine	420
Ophthalmology	350
Urology	330

References

Banks, J., J. S. Carson, B. L. Nelson, D. M. Nicol. 2005. *Discrete-Event System Simulation*. 4th edn. Pearson Prentice Hall, Upper Saddle River, NJ.

Berg, B., B. T. Denton, H. Nelson, H. Balasubramanian, A. Rahman, A. Bailey, K. Lindor. 2010. A discrete event simulation model to evaluate operational performance of a colonoscopy suite. *Med. Decis. Making* **30**(3): 380–387.

Blake, J. T., M. W. Carter. 1997. Surgical process scheduling: A structured review. *J. Soc. Health Syst.* **5**(3): 17–30.

Brindle, A. 1981. Genetic algorithms for function optimization. Unpublished doctoral dissertation, University of Alberta, Edmonton, Alberta, Canada.

Cardoen, B., E. Demeulemeester, J. Belien. 2010. Operating room planning and scheduling: A literature review. *Eur. J. Oper. Res.* **201**: 921–932.

Cayirli, T., E. Veral, H. Rosen. 2006. Designing appointment scheduling systems for ambulatory care services. *Health Care Manag. Sci.* **9**(1): 47–58.

Charnetski, J. 1984. Scheduling operating room surgical procedure with early and late completion penalty costs. *J. Oper. Manage.* **5**: 91–102.

Chow, V. S., M. L. Puterman, N. Salehirad, W. Huang, D. Atkins. 2011. Reducing surgical ward congestion through improved surgical scheduling and uncapacitated simulation. *Prod. Oper. Manag.* **20**(3): 418–430.

Deb, K., S. Agrawal, A. Pratap, T. Meyarivan. 2000. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimisation: NSGA-II. *Lect. Notes Comput. Sci.* **1917**: 849–858.

Denton, B., J. Viapino, A. Vogl. 2007. Optimization of surgery sequencing and scheduling decisions under uncertainty. *Health Care Manage. Sci.* **10**: 13–24.

Dexter, F., A. Macario, R. Traub. 1999a. Which algorithm for scheduling add-on elective cases maximizes operating room utilization? *Anesthesiology* **91**: 1491–1500.

Dexter, F., A. Macario, R. Traub, M. Hopwood, D. Lubarsky. 1999b. 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. *Anesth. Analg.* **89**: 7–20.

Dexter, F., E. Marcon. 2006. Impact of surgical sequencing on post anesthesia care unit staffing. *Health Care Manage. Sci.* **9**: 87–98.

Guinet, A., S. Chaabane. 2003. Operating theatre planning. *Int. J. Prod. Econ.* **85**(1): 69–81.

Gupta, D. 2007. Surgical suites' operations management. *Prod. Oper. Manag.* **16**: 689–700.

Gupta, D., B. Denton. 2008. Appointment scheduling in health care: Challenges and opportunities. *IIE Trans.* **40**: 800–819.

Ho, C. J., H. S. Lau. 1992. Minimizing total cost in scheduling outpatient appointments. *Manage. Sci.* **38**: 750–764.

Holland, J. H. 1975. *Adaptation in Natural and Artificial Systems*. The University of Michigan Press, Ann Arbor, MI.

Huschka, T., B. Denton, S. Gul, J. Fowler. 2007. Bi-criteria evaluation of an outpatient surgery procedure center via simulation. Henderson, S. G., B. Biller, M. H. Hsieh, J. Shortle, J. D. Tew, R. R. Barton eds, *Proceedings of the 2007 Winter Simulation Conference*. Institute of Electrical and Electronics Engineers Inc., Piscataway, NJ, 1510–1518.

Jebali, A., A. B. H. Alouane, P. Ladet. 2006. Operating room scheduling. *Int. J. Prod. Econ.* **99**(1–2): 52–62.

Lamiri, M., X. Xie, S. Zhang. 2008. Column generation approach to operating theater planning with elective and emergent patients. *IIE Trans.* **40**: 838–852.

Lehtonen, J. M., J. Kujala, J. Kouri, M. Hippelainen. 2007. Cardiac surgery productivity and throughput improvements. *Int. J. Health Care Qual. Assur.* **20**: 40–52.

Lovejoy, W. S., Y. Li. 2002. Hospital operating room capacity expansion. *Manage. Sci.* **48**(11): 1369–1387.

Lowery, J. 1992. Simulation of a hospital's surgical suite and critical care area. Swain, J., D. Goldsman, R. Crain, J. Wilson eds, *Proceedings of the 1992 Winter Simulation Conference*. Institute of Electrical and Electronics Engineers Inc., Piscataway, NJ, 1071–1078.

Lowery, J., J Davis. 1999. Determination of operating room requirements using simulation. Farrington, P., H. Nembhard, D. Sturrock, G. Ewans eds, *Proceedings of the 1999 Winter Simulation Conference*. Institute of Electrical and Electronics Engineers Inc., Piscataway, NJ, 1568–1572.

Magerlein, J. M., J. B. Martin. 1978. Surgical demand scheduling: A review. *Health Serv. Res.* **13**: 418–433.

Marcon, E., S. Kharraja, N. Smolski, B. Luquet, J. Viale. 2003. Determining the number of beds in the postanesthesia care unit: A computer simulation flow approach. *Anesth. Analg.* **96**: 1415–1423.

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

Price, C., B. Golden, M. Harrington, R. Konewko, E. Wasil, W. Herring. 2011. Reducing boarding in a post-anesthesia care unit. *Prod. Oper. Manag.* **20**(3): 431–441.

Syswerda, G. 1989. Uniform crossover in genetic algorithms. Schaffer, J. ed., *Proceedings of the Third International Conference on Genetic Algorithms*. Morgan Kaufmann Publishers, Los Altos, CA, 2–9.

Testi, A., E. Tanfani, G. Torre. 2007. A three-phase approach for operating theatre schedules. *Health Care Manage. Sci.* **10**: 163–172.

Tyler, D., C. A. Pasquariello, C. H. Chen. 2003. Determining optimum operating room utilization. *Anesth. Analg.* **96**: 1114–1121.

Weiss, E. N. 1990. Models for determining estimated start times and case orderings in hospital operating rooms. *IIE Trans.* **22**: 143–150.

Yellig, E. J., G. T. Mackulak. 1997. Deterministic scheduling in stochastic environments: The method of capacity hedge points. *Int. J. Prod. Res.* **35**: 369–379.

Zhou, J., F. Dexter. 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.