



# A novel deterministic model for simultaneous weekly assignment and scheduling decision-making in operating theaters

M. Haghi<sup>a</sup>, S.M.T. Fatemi Ghomi<sup>b,\*</sup> and P. Hooshangi-Tabrizi<sup>b</sup>

a. *Department of Industrial Engineering, Sharif University of Technology, Tehran, Iran.*

b. *Department of Industrial Engineering, Amirkabir University of Technology, Tehran, Iran.*

Received 6 June 2015; received in revised form 21 May 2016; accepted 20 August 2016

## KEYWORDS

Operating room  
 scheduling;  
 Elective patients;  
 Recovery beds;  
 Mixed integer linear  
 program;  
 Meta-heuristic  
 methods.

**Abstract.** This paper studies a simultaneous weekly assignment and scheduling decision-making problem in operating theaters with elective patients. Because of limited resources in hospitals, considering assignment and scheduling decisions simultaneously can help managers exploit the available resources more efficiently and make the work-load uniformly distributed during the planning horizon. This procedure can significantly reduce hospital costs and increase satisfaction of patients and personnel. This paper formulates the mentioned problem as a Mixed Integer Linear Program (MILP) considering applicable assumptions like finite recovery beds and limitation of equipment. Since the problem is NP-hard, in order to solve large-scale instances and deal with the complexity, two effective and efficient algorithms are designed. Finally, as a practical case of study, a real data set of a surgery department of a big hospital in Iran (Aalinasab-e Tabriz) is used to solve the studied problem by the proposed algorithms.

© 2017 Sharif University of Technology. All rights reserved.

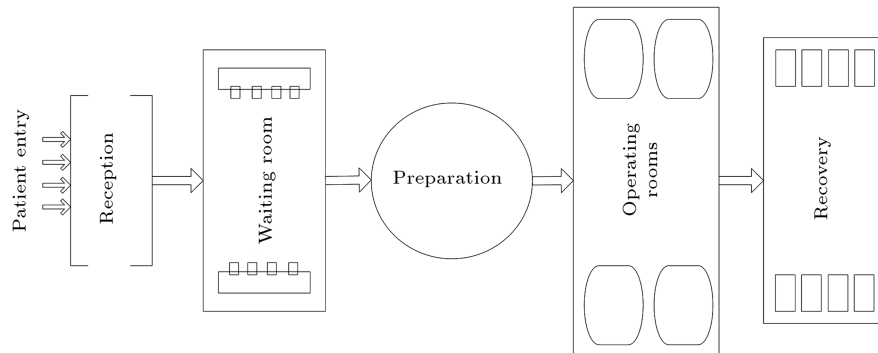
## 1. Introduction

Nowadays, health care costs have grown significantly and hospital expenditures account for approximately a third of this total amount (Centers for Medicare and Medicaid Services, 2007). Surgery operations incur a large portion of a hospital's total expenses and revenues (Healthcare Financial Management Association 2003). Thus, operating theater as a critical part and the largest consumer of resources in the hospital requires efficient and effective planning to use resources and facilities. Each surgery operation needs a lot of resources, including operating rooms, staffs (surgeons, nurses, anesthesiologists, etc.), recovery beds, and surgery

equipment. Therefore, due to the increasing demand for surgery operations, proper management of these limited resources is highly essential. Furthermore, attention to the correlations between operating rooms and other departments, such as recovery room or Intensive Care Unit (ICU), significantly increases the importance and complexity of the problem. Figure 1 generally shows a simple flow of patients from entrance to the recovery room in a hospital.

The common procedure for the planning and scheduling of surgery department in almost all hospitals and medical centers has two main stages: the assignment of patients to operating rooms and the sequencing of the assigned patients in each room. In the first stage, a day and a specific operating room are assigned to each patient. Then, in the second stage, the sequencing of patients in each room is determined. These two sequential main steps have been often considered independently in most papers.

\*. Corresponding author. Tel.: +98 21 64545381  
 E-mail addresses: M\_hagh@encs.concordia.ca (M. Haghi);  
 fatemi@aut.ac.ir (S.M.T. Fatemi Ghomi);  
 P\_hoosha@encs.concordia.ca (P. Hooshangi-Tabrizi)



**Figure 1.** The flow of patients in a hospital.

Jebali et al. [1] proposed a two-stage procedure for the operating room scheduling problem. At the first stage, surgery operations were allocated to the rooms and, then, in the second stage, the surgery operations were scheduled and the start time of each surgery was determined. Cardeon et al. [2] only focused on the patient sequencing stage in their studied problem. They formulated the problem as a Mixed Integer Linear Program (MILP) with some assumptions such as operating pediatric or priority patients in the beginning of each day.

In other related works, Denton et al. [3] investigated the problem of determining the number of rooms to open on each day and daily allocation of surgery blocks to the rooms in a hospital with several identical rooms (in terms of equipment). In continuation of their work, Batun et al. [4] extended the problem to the case that the daily assignment and scheduling of elective patients were concerned. They assumed that the sequence of surgery operations of each surgeon was predetermined and they only had to schedule the surgeons. Table 1 classifies most related papers to the problem in the literature review.

To the best of our knowledge, there are few studies in which the assignment and scheduling decisions are made simultaneously. Furthermore, weekly planning of the resources in surgery departments and using both generated and real data in evaluating solution methodologies have been neglected by most papers in the literature. On the other hand, the lack of a comprehensive study incorporating recovery bed and equipment limitation in weekly operating scheduling problems can be seen in the existing studies. Therefore, this paper proposes a mixed integer linear program to determine the weekly assignment of patients to the specific operating rooms and days in the planning horizon and the sequencing of patients within each room on each day simultaneously. Moreover, the proposed MILP considers the availability of the equipment in operating rooms and the recovery beds after the patient's surgery. For large-scale instances of the problem, two algorithms based on genetic and

simulated annealing meta-heuristics are developed to solve the studied problem. A set of real test problems obtained by a hospital in Iran is used to evaluate the performance of the proposed algorithms.

The remainder of the paper is as follows. Section 2 formulates the studied problem as MILP. The proposed algorithms are explained in Section 3. Section 4 deals with the computational experiments and, finally, Section 5 concludes the paper.

## 2. The MILP formulation of the problem

This section formulates the problem as a mixed integer linear program. Tables 2 and 3 represent the sets, indices, parameters, and decision variables used in the proposed MILP. The proposed MILP of the problem is given in the following:

$$\text{Min} \sum_{i=1}^5 w_i \times \text{Obj}_i$$

$$\text{Obj}_1 = \sum_{d \in D} \sum_{r \in R} \sum_{p \in P} CO_{p,d} \times x_{p,r,d},$$

$$\text{Obj}_2 = \sum_{d \in D} \sum_{r \in R} COT \times \frac{Ot_{r,d}}{Ot_{\max}},$$

$$\text{Obj}_3 = \sum_{d \in D} CLC \times \frac{Ct_d}{Ct_{\max}},$$

$$\text{Obj}_4 = \sum_{p \in P} CRO \times \frac{z_p}{z_{\max}},$$

$$\text{Obj}_5 = \sum_{p \in P} CE \times \frac{\max\{0, St_p^R - PT\}}{Dev_{\max}},$$

$$\sum_{i=1}^5 w_i = 1$$

**Table 1.** Overview of the research on scheduling of operating rooms.

Ref.	Problem	Patients		Model		Data		Solution methodology	
	Sequencing	Assignment	Emergency	Elective	Stochastic	Deterministic	Based on real cases		Generated
[5]				✓	✓			✓	DES
[6]		✓				✓		✓	Heuristic
[7]		✓		✓		✓	✓	✓	GP, Monte Carlo
[8]		✓			✓			✓	DP
[9]	✓			✓	✓			✓	LP, heuristic, AP
[10]	✓			✓	✓		✓		DES
[11]		✓		✓	✓		✓		LP
[12]			✓	✓	✓			✓	DES
[13]	Independently		✓	✓	✓	✓	✓		DES, heuristic, SA
[14]	✓			✓	✓		✓		Heuristic, two stage SLP
[15]	Independently				✓				VIA
[16]		✓	✓	✓	✓			✓	Monte Carlo, CG
[17]		✓		✓		✓	✓	✓	MIP
[18]	Independently			✓	✓	✓	✓		MIP, DES
[19]			✓	✓	✓		✓		DES
[20]		✓		✓		✓		✓	DP, B&P
[21]		✓	✓	✓	✓			✓	MIP, AP
[22]		✓	✓	✓	✓			✓	DP, CG, heuristic
[23]		✓		✓		✓		✓	DP, B&P
[24]	Simultaneous		✓	✓		✓		✓	MILP
[25]			✓	✓	✓		✓		DES
[26]	✓			✓		✓	✓		MILP
[27]		✓		✓	✓		✓		QP, MIP, SA
[28]	Independently			✓		✓	✓		HGA
[29]		✓		✓		✓		✓	VIA
[30]		✓		✓	✓			✓	SAA
[31]	Independently			✓		✓	✓		GP
[32]	✓			✓	✓			✓	SAA, Bender's decomposition
[33]	✓			✓	✓		✓		TS
[34]		✓		✓		✓	✓		MP, heuristic
[35]		✓		✓		✓	✓		0-1 LP, metaheuristic
[36]		✓		✓		✓	✓		LP, heuristic
[37]	Sequencing based on pre-defined rules	✓		✓		✓	✓		IP, heuristic

Mixed Integer Programming (MIP), Discrete Event Simulation (DES), Goal Programming (GP), Tabu Search (TS), Dynamic Programming (DP), Linear Programming (LP), Analytical Procedure (AP), Simulated Annealing (SA), Stochastic Linear Programming (SLP), Column Generation (CG), Quadratic Programming (QP), Branch and Price (B&P), Heuristic, Hybrid Genetic Algorithm (HGA), Sample Average Approximation (SAA), Value Iteration Algorithm (VIA), Mathematical Programming (MP), Integer Programming (IP).

**Table 2.** Sets, indices, and parameters used in the proposed MILP model.

$P$	A set of elective patients
$R$	A set of operating rooms
$B$	A set of recovery beds
$D$	A set of days in planning horizon
$S$	A set of surgeons
$p, p'$	Indices for the patients ( $p, p' \in P$ )
$r$	Index for an operating room ( $r \in R$ )
$s$	Index for a surgeon ( $s \in S$ )
$d$	Index for a day ( $d \in D$ )
$b$	Index for a recovery bed ( $b \in B$ )
$DD_p$	The deadline for surgery of patient $p$
$HD_p$	Hospitalization date of patient $p$
$Sur_{p,s}$	A parameter taking 1 if patient $p$ is assigned to surgeon $s$ and 0 otherwise
$Op_{r,d}$	A parameter taking 1 if operating room $r$ is opened on day $d$ and 0 otherwise
$Eq_{p,r}$	A parameter taking 1 if room $r$ has essential equipment for the surgery of patient $p$ and 0 otherwise
$Exp_p$	A parameter taking 1 if an experiment before the operation is needed for patient $p$ and 0 otherwise
$Pre_p$	The preparation time for patient $p$
$P_p^{ST}$	Surgery duration of patient $p$
$P_p^{RT}$	Recovery duration of patient $p$
$Dur_{r,d}$	The available time of operating room $r$ on day $d$
$Cl_p$	The required time to clean the operating room after the surgery of patient $p$
$SR$	Daily opening time of operating rooms
$RefT$	Reference time for the surgery of patients with incomplete records
$AT_{s,d}$	The presence time of surgeon, $s$ on day $d$
$CLC$	The cost of operating room's occupation
$CRO$	The cost of spending recovery in operating room
$COT$	The cost of operating room's overtime per excessive work-hour
$CE$	The cost of eliminating a patient from the planning horizon
$CO_{p,d}$	The cost of operating patient $p$ on day $d$
$PT$	A predetermined time indicating that the surgery with start time greater than that are eliminated from current planning horizon
$Ot_{\max}$	The maximum possible overtime
$Ct_{\max}$	The maximum possible completion time
$Z_{\max}$	The maximum recovery time incurring in operating room
$Dev$	The maximum possible deviation from $PT$
$M$	A big number

$$\sum_{d=1}^{DD_p} \sum_{r \in R} x_{p,r,d} = 1 \quad \forall p, \quad (1) \quad x_{p,r,d} \leq \frac{op_{r,d} + Eq_{p,r}}{2} \quad \forall p, r, d, \quad (3)$$

$$\sum_{r \in R} x_{p,r,d} = 0 \quad \forall p, d \leq HD_p, \quad (2) \quad OSe_{p,p',r,d} + OSe_{p',p,r,d} \leq \frac{x_{p,r,d} + x_{p',r,d}}{2} \quad \forall p, p' > p, r, d, \quad (4)$$

**Table 3.** Decision variables used in the proposed MILP model.

$x_{p,r,d}$	$\begin{cases} 1 & \text{If patient } p \text{ is assigned to room } r \text{ on day } d \\ 0 & \text{Otherwise} \end{cases}$
$OSe_{p,p',r,d}$	$\begin{cases} 1 & \text{If the surgery of patient } p \text{ precedes that of patient } p' \text{ in room } r \text{ on day } d \\ 0 & \text{Otherwise} \end{cases}$
$RSe_{p,p',d}$	$\begin{cases} 1 & \text{If the recovery of patient } p \text{ precedes that of patient } p' \text{ on day } d \\ 0 & \text{Otherwise} \end{cases}$
$DSe_{p,p',d}$	$\begin{cases} 1 & \text{If the surgery of patient } p \text{ precedes that of patient } p' \text{ on day } d \\ 0 & \text{Otherwise} \end{cases}$
$y_{p,b}$	$\begin{cases} 1 & \text{If patient } p \text{ is assigned to recovery bed } b \\ 0 & \text{Otherwise} \end{cases}$
$St_p^S$	The starting time for the surgery of patient $p$
$St_p^R$	The starting time for the recovery of patient $p$ in the recovery room
$z_p$	The portion of the recovery time for patient $p$ elapsed in the operating room
$Ot_{r,d}$	The overtime of room $r$ on day $d$
$Ct_d$	The completion time of all the surgery and recovery of all patients on day $d$

$$OSe_{p,p',r,d} + OSe_{p',p,r,d} \geq x_{p,r,d} + x_{p',r,d} - 1$$

$$\forall p, p' > p, r, d, \quad (5)$$

$$RSe_{p,p',d} + RSe_{p',p,d}$$

$$\leq \frac{\sum_{r \in R} x_{p,r,d} + \sum_{r \in R} x_{p',r,d}}{2} \quad \forall p, p' > p, d, \quad (6)$$

$$RSe_{p,p',d} + RSe_{p',p,d} \geq \sum_{r \in R} x_{p,r,d}$$

$$+ \sum_{r \in R} x_{p',r,d} - 1 \quad \forall p, p' > p, d, \quad (7)$$

$$DSe_{p,p',d} + DSe_{p',p,d}$$

$$\leq \frac{\sum_{r \in R} x_{p,r,d} + \sum_{r \in R} x_{p',r,d}}{2} \quad \forall p, p' > p, d, \quad (8)$$

$$DSe_{p,p',d} + DSe_{p',p,d}$$

$$\geq \sum_{r \in R} x_{p,r,d} + \sum_{r \in R} x_{p',r,d} - 1 \quad \forall p, p' > p, d, \quad (9)$$

$$St_{p'}^t + Pre_{p'} \geq St_p^S + Pre_p + P_p^{ST}$$

$$-M \left( 3 - DSe_{p,p',d} - Sur_{p,s} - Sur_{p',s} \right)$$

$$\forall p, p' \neq p, r, d, \quad (10)$$

$$St_p^S + Pre_p \geq Sur_{p,s} \times AT_{s,d} - M \left( 1 - \sum_{r \in R} x_{p,r,d} \right)$$

$$\forall p, s, d, \quad (11)$$

$$St_{p'}^R \geq St_p^R + P_p^{RT} - z_p$$

$$-M \left( 3 - RSe_{p,p',d} - y_{p,b} - y_{p',b} \right)$$

$$\forall p, p' \neq p, b, d, \quad (12)$$

$$St_{p'}^R \geq St_p^R - M(1 - RSe_{p,p',d}) \quad \forall p, p' \neq p, d, \quad (13)$$

$$St_p^R = St_p^S + Pre_p + P_p^{ST} - z_p \quad \forall p, \quad (14)$$

$$z_p \geq P_p^{RT} \quad \forall p, \quad (15)$$

$$\sum_{b \in B} y_{p,b} = 1 \quad \forall p, \quad (16)$$

$$St_p^S + M \left( 1 - \sum_{r \in R} \sum_{d \in D} x_{p,r,d} \right) \geq SR \quad \forall p, \quad (17)$$

$$Ot_{r,d} \geq St_p^R + CL_p - M(1 - x_{p,r,d})$$

$$-Dur_{r,d} - SR \quad \forall p, r, d, \quad (18)$$

$$St_p^S + 2M \left( 1 - \sum_{r \in R} \sum_{d \in D} x_{p,r,d} \right) \geq RefT - M(1 - Exp_p) \quad \forall p, \quad (19)$$

$$St_{p'}^S \geq St_p^S + Pre_p + P_p^{ST} + CL_p + z_p - M(1 - OSe_{p,p',r,d}) \quad \forall p, p' \neq p, r, d, \quad (20)$$

$$Ct_d + M(1 - \sum_{r \in R} x_{p,r,d}) \geq St_p^R + P_p^{RT} - z_p \quad \forall p, d, \quad (21)$$

$$St_p^S, St_p^R, z_p \geq 0 \quad \forall p, \quad (22)$$

$$Ot_{r,d} \geq 0 \quad \forall r, d, \quad (23)$$

$$x_{p,r,d}, y_{p,b}, OSe_{p,p',r,d}, RSe_{p,p',d}, DSe_{p,p',d} \in \{0,1\} \quad \forall p, p', b, r, d. \quad (24)$$

The objective function consists of five terms. The first term is related to minimizing the surgery costs of patients, which is directly associated with the patients' priorities and the day of surgery. In this term, the value of  $CO_{p,d}$  for each patient is usually different from one day to the next day. If the patient's health is in a dangerous situation,  $CO_{p,d}$  increases from the first day of the planning horizon to the last one, indicating that more delay in surgery imposes more costs. The second term minimizes the overall overtime in operating rooms. The third term tries to complete all patients' surgery as soon as possible. The fourth term minimizes the part of the patients' recovery times spent in the operating rooms as a result of the assigned recovery bed's occupation. Finally, the last term tries to minimize the overall cancelation cost of surgery. According to the latter term, the model tries to start surgery before a predefined time. It should be noted that, as there are both monetary and time-based terms in the objective function, all terms are normalized. Furthermore, the same values of  $w$  for the objective terms are assumed.

Constraint (1) ensures that each patient is assigned to only one operating room and one day. Constraint (2) guarantees that the surgery of each patient is operated after his/her hospitalization date. Constraint (3) says that each patient can be assigned to an operating room only if it is opened and has the required equipment. Constraints (4) and (5) determine the precedence relation between each pair of surgery operations assigned to the same operating room on

the same day. Constraints (6) and (7) determine the precedence relation between the recoveries of patients assigned to the same day. Constraints (8) and (9) state that a precedence relation exists between the start times of the surgery operations of the same surgeon on each day. Constraint (10) implies that a surgeon can only do surgery on one patient at a time. The presence of the surgeon at the hospital after preparing the patient for his/her surgery is ensured by Constraint (11). Constraint (12) says that each recovery bed cannot be occupied by more than one patient at a time. Constraint (13) relates the starting times of recovery of the consecutive patients assigned to the same day. The relation between the starting times of surgery and recovery for each patient is determined by Constraint (14). Constraint (15) implies that for each patient, the recovery time spent in the operating room has to be less than the total recovery time required for him/her. Assigning only one recovery bed to each patient is enforced by Constraint (16). Constraint (17) ensures that all the surgery operations begin after the opening time of the operating theater. The overtime for each operating room is calculated by Constraint (18). In Constraint (19), a reference time is used to provide a time for patients who need to do some specific tests and experiments before their surgery. Constraint (20) represents the relation between starting times of the surgery for the consecutive patients. In this constraint, the cleaning time of the operating room between successive surgery operations is also considered. Constraint (21) computes the completion time of all the surgery on each day. Constraints (22) to (24) define the decision variables.

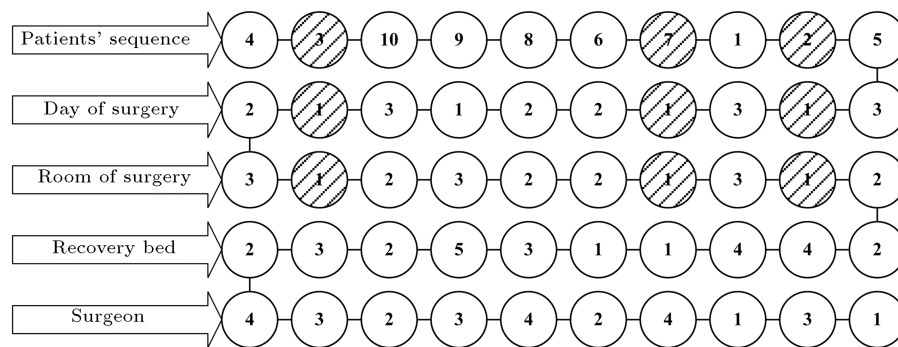
### 3. Algorithmic approach

The studied problem in this paper is placed in the category of the NP-hard problems and exact solution methods cannot be used to solve it, even in medium sized instances. Therefore, in order to solve the real-size instances of the problem, two meta-heuristic methods based on genetic and simulated annealing algorithms are developed. Since both the proposed algorithms utilize the similar solution representation and neighborhoods, these subjects are described at first and, then, the algorithms will be explained.

#### 3.1. Solution representation

The solution to the studied problem is represented with a five-row matrix in which the first row defines the sequence of patients for surgery. The second to the fourth rows indicate the day, the room, and the recovery bed assigned to the patients, respectively.

To determine the sequence of the surgery from solution matrix, first, we should find the patients with the same assigned day and room, i.e. find the patients



**Figure 2.** The solution matrix of the given example.

whose corresponding entries in the second and third rows are similar. The sequence of patients is in the similar order with the numbers in the first row of the solution matrix.

For example, assume that we have 10 patients that should be scheduled in 3 days and 3 rooms. Furthermore, 5 recovery beds and 4 surgeons are supposed. Figure 2 gives the solution matrix for this example.

According to the solution matrix shown in Figure 2, for example, the surgery of patients 3, 7, and 2 will be performed in the same room on the same day of planning horizon and the surgery sequence of them is  $3 \rightarrow 7 \rightarrow 2$ .

### 3.2. Neighborhoods definition

This paper uses well-defined mutation and crossover operators exploiting the structure of the problem to find the adjacent solutions described in the following. It should be noted that in the definition of the neighborhoods, a chromosome means a solution matrix.

#### 3.2.1. Mutation operator

The mutation is a way to enlarge the search space and prevent the algorithm from falling into a local optimum. In this paper, three types of mutation operators are used as follows:

1. Depending on the size of the problem, select a number of patients randomly and regenerate their days, rooms, and recovery beds randomly;
2. Select a number of patients randomly and regenerate their days, rooms, or recovery beds randomly;
3. Select two patients randomly and replace their days, rooms, and recovery beds with each other.

#### 3.2.2. Crossover operator

A well-defined crossover operator is the one that maintains the efficient part of parent chromosomes and has enough freedom to provide searching within a larger portion of the solution space. This subsection introduces a hybrid crossover operator that keeps the Longest Common Subsequence (LCS) of parents

in children chromosomes. Using this operator, the algorithm can search in a larger solution space while retaining the good part of the parent chromosomes. The following explains how the LCS works on given solutions:

1. Find the LCS in either parents. Put the patients of the common subsequence in the first child with the same place as Parent 1;
2. Find the first patient of Parent 2 which is not a member of the LCS and copy it in the first available location in the first child;
3. Repeat the same process defined in step 2 with the other components of Parent 2 which are not members of the LCS until the first row of the first child is completed;
4. Repeat steps 1-3 in order to create the first row of the second child.

To clarify how the proposed crossover works, the following example is given. Consider the first row of two chromosomes of a problem with 4 patients as follows:

Parent 1 = 4 6 9 3 7 2 8 1 5

Parent 2 = 1 7 4 2 9 3 8 6 5

Longest common sub-sequence = 4 9 3 8 5

Child 1 = 4 \_ 9 3 \_ \_ 8 \_ 5

Child 2 = \_ \_ 4 \_ 9 3 8 \_ 5

Child 1 = 4 1 9 3 7 2 8 6 5

Child 2 = 6 7 4 2 9 3 8 1 5

In this way, the first row of each child chromosome is formed as it was explained before. To complete the next rows of the children's matrices, two approaches are considered. In the first approach, for the first child, the components of rows corresponding to the LCS take values from Parent 1 and other components are taken from the corresponding ones in Parent 2. In the second approach, for the first child, the components

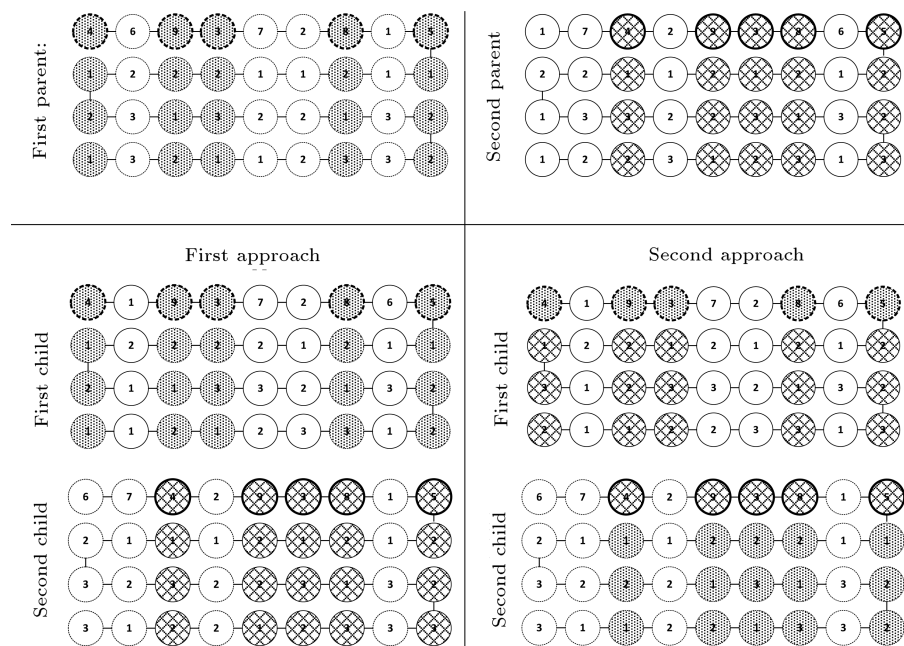


Figure 3. The approaches for crossover operator.

of all rows corresponding to the LCS are taken from Parent 2, and for the second child, they are taken from Parent 1. Figure 3 represents these two approaches for the mentioned example.

In the following subsections, two algorithms are developed in accordance with the problem under study.

### 3.3. Simulated annealing algorithm

Simulated annealing approach is a form of local search heuristic. For minimizing problems, traditional local search methods usually use reduction strategies that always move in reduction direction and often lead to convergence on the local optimum solution. To escape from the local optimum, the algorithm must also have some restricted and controlled uphill moves. In simulated annealing algorithm, if the objective function increases with an uphill movement, the new state is accepted with the probability obtained by equation  $p = \exp\left(\frac{-\Delta F}{T}\right) > r$ . In this equation,  $r$  is a random number from uniform distribution in the interval  $[0,1]$ . The acceptance probability of uphill move is a function of both temperature and changes in the objective function. When the temperature of the system decreases or reaches zero, just the best movements will be accepted. Elements of simulated annealing algorithm are presented as follows:

- Initial temperature ( $T_0$ ): The initial temperature must be large enough so that in  $T_0$  more unfavorable transfers are accepted;
- Temperature reduction function: The geometric functions can be used to reduce the temperature as

follows:

$$T_{k+1} = \alpha T_k, \quad 0.8 \leq \alpha \leq 0.99,$$

where  $\alpha$  represents the temperature reduction and remains constant throughout the algorithm;

- Period length as a balance condition: Period length is the number of repetitions that take place at a constant temperature to reach equilibrium. This number is constant for all temperatures;
- Stop condition: Stop condition can be expressed in different ways. Two of the known conditions are presented in the following:
  1. To achieve a predetermined final temperature ( $T_f$ ), which is a very small fraction of the initial temperature ( $T_f = \beta T_0$   $\beta \leq 0.01$ );
  2. To reach a maximum value of a predetermined number of iterations.

The procedure of the simulated annealing algorithm to solve the studied simultaneous scheduling and assignment problem is as follows.

At first, the initial parameters are given as input of the algorithm. Then, an initial solution is randomly generated. The overall structure of generating initial solutions is to produce a randomized sequence of patients. After that, the available days, rooms, and beds are randomly assigned to each patient. It should be noticed that the procedure applied in this paper produces only feasible solutions in terms of assigned rooms and days to each patient. At the next stage, the objective function of the initial solution is calculated.



Afterwards, the algorithm enters into two loops, which control two parameters including the temperature and the number of repetitions in each temperature, respectively. In each run of the first loop, the second loop produces  $N$  solutions. In the second loop, according to the current temperature range, one of the designed neighborhood structures is applied to the current solution to produce a new solution and its corresponding objective function is computed. It should be noted that some of these adjacent solutions are infeasible. Therefore, a function is required to check the feasibility of the solutions and modify them if it is necessary.

In the next stage, the objective function is compared with the best objective function. In case of improvement, the current solution and its objective function are replaced by the best solution. Otherwise, if the current solution is worse than the best solution, a random uniform number between zero and one ( $r$ ) is generated. If  $\exp\left(\frac{-\Delta F}{T}\right) > r$ , in spite of the fact that the current solution is worse than the best, it is replaced by the best solution. At the end of this phase, the number of iterations increases by one and the inner loop is repeated with the corresponding designed neighborhood structures (the defined mutations and LCS crossover). This procedure will continue until the number of inner iterations reaches  $N$ . Then, the temperature decreases by the reduction rate and the process is repeated again until the final temperature is reached.

Neighborhood structures in the high temperatures are generated in order to increase the diversification and to search a vast solution space. For low temperatures, structures with less perturbation are used to make intensification. Figure 4 shows the flowchart of the proposed simulated annealing algorithm schematically.

### 3.4. Genetic algorithm

This subsection designs another solution method, Genetic Algorithm (GA), and compares it with the developed simulated annealing algorithm. In general, GA is comprised of the following components:

- **Chromosomes:** Chromosome represents a feasible solution to the problem in the search space. In Subsection 3.1, the solution to the studied problem was represented;
- **Population:** A set of chromosomes make up a population. Through the effect of performing the defined mutation and crossover operators on a population, a new population with the same number of chromosomes is formed;
- **Fitness function:** For each chromosome, fitness function returns a non-negative integer that represents

the objective function of the individual chromosomes;

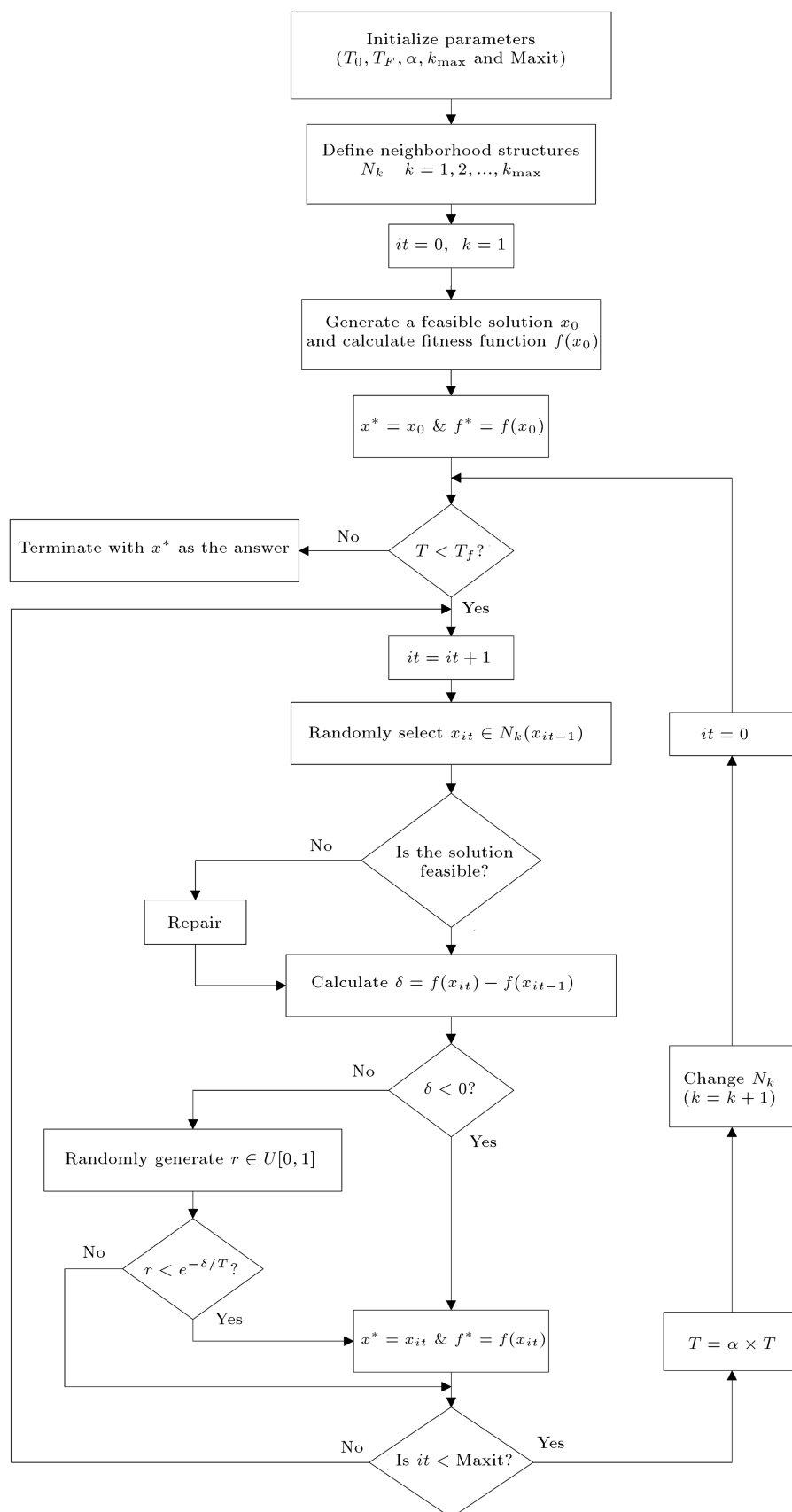
- **Genetic operators:** In genetic algorithms, operators are used during the reproduction stage. By implementing these operators on a population, the population of the next generation is produced. Selection, crossover, and mutation operators have the most usage in genetic algorithms. These operators are introduced in the following separately:
  1. *Selection:* By performing the selection operator, a number of chromosomes in the current population are selected for reproduction. More graceful chromosomes have more chances to be selected for reproduction. A *roulette wheel* selection is used as the selection operator in this paper;
  2. *Crossover:* The crossover operator is applied to a pair of chromosomes in the current generation and produces a new pair of chromosomes for the next generation. This operator is implemented on each pair of chromosomes with a pre-defined probability ( $P_c$ ). See Subsection 3.2.2 for more details on the used crossover;
  3. *Mutation:* After performing the crossover operation, mutation operator is implemented on the chromosomes with a probability ( $P_m$ ). This operator is used to avoid trapping into the local optimal solutions. See Subsection 3.2.1 for more details on the used mutations.

The following steps give a brief description of the GA's procedure:

- **Step 0:** Initialize parameters;
- **Step 1:** Generate an initial population randomly;
- **Step 2:** Evaluate the fitness for each individual in the current population;
- **Step 3:** Define selection rule and choose the best individuals to produce a new population;
- **Step 4:** Call the crossover operator for the selected individuals with probability  $P_c$  to generate new individuals;
- **Step 5:** Call the mutation operator for each member with probability  $P_m$ ;
- **Step 6:** Call a repair function for each member if the solution is infeasible;
- **Step 7:** Calculate affinity function, i.e. the similarity degree to the best individual of the previous generation, for each individual and transfer the individuals whose affinities are less than  $P_{\text{affinity}}$ . In our proposed genetic algorithm, the affinity function is defined as follows:

$$Aff = \frac{1}{1 + k/n},$$

where  $k$  is the number of gens that have the same



**Figure 4.** The flowchart of the proposed simulated annealing algorithm.

**Table 4.** The information of the surgical ward of the hospital.

Surgical specialties	Surgery of each specialty	The range of operation time	The number of surgeons in each specialty
<b>Urology</b>	Removal of the bladder	360-480 minutes	5
	Lithotripsy	30-120 minutes	
	Prostate surgery	60-180 minutes	
	Varicocelelectomy	30-90 minutes	
	Prostatectomy	90-120 minutes	
	Hernia	60-90 minutes	
<b>Brain and nerves</b>	Disk	120-300 minutes	3
	Brain tumor	180-360 minutes	
<b>General</b>	Appendix	30-90 minutes	6
	Gallbladder	45-90 minutes	
	Hernia	60-180 minutes	
	Stomach tumor	180-300 minutes	
	Esophagus tumor	180-360 minutes	
<b>Ear, Nose, and Throat (ENT)</b>	Mastoidectomy	120-240 minutes	4
	Tonsil	30-90 minutes	
	Rhinoplasty	180-300 minutes	
<b>Orthopedics</b>	Total knee	180-240 minutes	4
	Total hip	180-240 minutes	
	Plaque removal	60-180 minutes	
	Knee ligament repair	180-240 minutes	
<b>Eye</b>	Eye	15-30 minutes	2

contents and  $n$  denotes dimension of the chromosomes;

- Step 8: If the termination condition is met, stop; otherwise, repeat Steps 2-8.

#### 4. Computational study

The proposed MILP was coded in Lingo 11.0 and was solved using CPLEX 12.1 solver. The proposed solution algorithms were implemented in MATLAB R2011b and run on a PC with 2.66 GHz Intel Core i5 and with 4 GB of RAM memory.

##### 4.1. Description of the real data

The required data of the studied problem have been collected from a hospital in Iran (Aalinasab-e Tabriz). These data are used as a basis to create the problem instances in this paper.

The studied hospital has six operating rooms shared among all specialties in all days of the week. Most of the surgery operations can be performed in all rooms. Just some eye and orthopedic surgery can be

done only in specific rooms due to the requirement of special equipment. There are 9 recovery beds in the operating theater. The recovery may take, on average, between twenty to forty minutes, depending on each patient. The time required to prepare a patient for a surgery operation usually takes an average of ten minutes. Finally, the total cleaning and sterilizing time of the operating room is taken between 15 and 30 minutes depending on the duration of the surgery. Table 4 summarizes the information of the surgery performed in surgical ward of the hospital.

##### 4.2. Taguchi design of experiments

Meta-heuristic algorithms have a random nature and, as a result, the diversity of the obtained solutions might be high. Thus, an effective and efficient algorithm is one that its solutions converge on the global optimal solution. In this case, the solutions of the algorithm can be trusted. One of the most important factors to achieve an efficient algorithm is the appropriate tuning of parameters. Therefore, in this paper, the Taguchi design of experiments has been applied to tune the

**Table 5.** Parameter tuning of the genetic algorithm.

Parameter	Value
Number of generations	150
Population size	70
Mutation probability	0.35
Affinity	0.9

**Table 6.** Parameter tuning of the simulated annealing algorithm.

Parameter	Value
Maximum number of iterations	150
Initial temperature	150
Reduction rate of the temperature	0.95
Final temperature	20

parameters at their proper values and to achieve a good performance of GA and SA algorithms.

Taguchi design of experiments has some predetermined patterns, each of which applies a different number of factors and different levels for each factor. Therefore, the investigator must select one of these predetermined patterns to define his/her tests. In this paper, the pattern of  $L_9(4^3)$  is used. This pattern shows that the algorithm has four factors, each at three levels such that the combination of them forms 9 orthogonal scenarios. Tables 5 and 6 show the levels considered for each parameter of the proposed algorithms after applying the Taguchi method.

In order to evaluate the performance of designed algorithms, a Relative Deviation Percentage (RDP) is defined as follows:

$$RDP = \frac{ALG_{sol} - OPT_{sol}}{OPT_{sol}} \times 100,$$

where  $ALG_{sol}$  is the solution obtained from algorithm and  $OPT_{sol}$  is the optimal solution of the MILP.

The mathematical model and the proposed algorithms have been implemented on three instances of the problem: the first sample consists of planning and scheduling 10 patients for one day, the second instance schedules 15 patients for two days, and the third instance considers 20 patients for two days.

It should be noted that for each problem instance, algorithms are run five times and the average of RDPs is taken for comparisons. Table 7 presents the average RDPs obtained by the proposed algorithms.

As it is obvious from the above table, both algorithms perform very well in small-size instances and reach the optimum solution.

#### 4.3. The comparative studies

This subsection applies the proposed algorithms to five different sized instances of the problem created based on real data collected from the Alinasab-e Tabriz Hospital in Iran. Each of the algorithms has been run five times for each sample and the results of the comparison are presented in Table 8. In this table, best-known solutions of the algorithms from different runs are used instead of optimal solution in calculating RDPs.

Figures 5 and 6 show the comparison of the average RDPs and average CPU times obtained by both developed algorithms for different sizes of the problem. As it is obvious from these figures, both algorithms have good performance in small-size instances; but, as the size of the problem gradually increases, GA

**Table 7.** The CPU times and average RDPs of the two proposed meta-heuristic algorithms.

Problem size	MILP	SA		GA	
	CPU time (s)	CPU time (s)	Average RDPs (%)	CPU time (s)	Average RDPs (%)
10	0.5	16	0.00	38	0.00
15	3	8	0.00	104	0.00
20	1095	10	0.00	71	0.00

**Table 8.** Comparison results of the proposed algorithms.

Problem instance	Number of patients	Number of days	Average RDPs (%)		Average CPU times (s)	
			SA	GA	SA	GA
1	15	2	0.00	0.00	7.808	104.16
2	30	2	0.10	0.19	13.356	189.114
3	50	3	0.29	1.01	25.138	381.38
4	100	4	1.58	26.93	53.48	959.772
5	250	7	4.34	36.28	190.462	3582.8

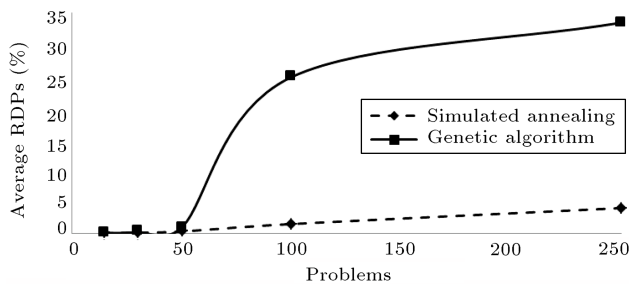


Figure 5. The average RDPs obtained by the algorithms.

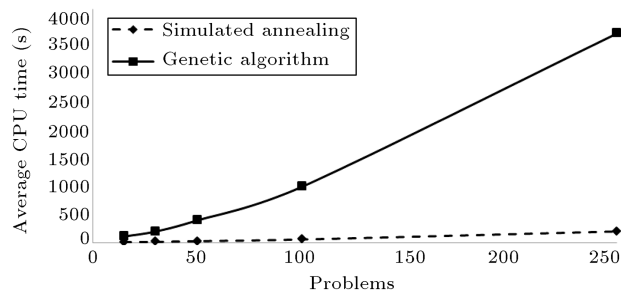


Figure 6. The average CPU times obtained by the algorithms.

loses its efficiency, but simulated SA indicates better performance in terms of time and accuracy measures.

In order to have a better comparison, a single factor analysis of variance (ANOVA) is used to examine the effectiveness of the algorithms. For this analysis, the equality of the average solutions obtained from the two levels of treatments (GA and SA) is considered as hypothesis testing. For each level, five observations have been collected and the confidence level of 95% is considered. The analysis of variance has been applied for four different sized samples of the studied problem including 30, 50, 100, and 250 patients, respectively. The summary of the obtained results is shown in Table 9 and, as an example, the details of the ANOVA for the instance with 100 patients are shown in Table 10.

Table 9. The obtained results of ANOVA.

Problem instance	<i>p</i> -value
30	0.429917
50	0.391644
100	0.000247
250	0.004193

Table 10. The detailed results of ANOVA for an instance with 100 patients.

Groups	Count	Sum	Average	Variance
Algorithm (100)	5	222811	44562.2	394698.7
Error	5	278400	55680	15442571

Source of variation	SS	df	MS	F	p-value
Between groups	3.09E+08	1	3.09E+08	39.02361	0.000247
Within groups	63349079	8	7918635		
Total	3.72E+08	9			

According to the *p*-values of Table 9, for a confidence level of 95%, there is no significant difference between the proposed algorithms for small-size instances, but for larger problems, there is significant difference between two algorithms and the proposed SA algorithm provides better performance than the GA does.

## 5. Conclusion and future research

This paper studied operating room scheduling problem for elective patient. The contributions of this paper were threefold. First, for the first time in the literature, a comprehensive Mixed Integer Linear Program (MILP) was provided to formulate the simultaneous weekly assignment and scheduling operating room problem. Second, in order to represent the problem more realistically, recovery bed limitations, equipment requirements, and priorities of patients based on their health conditions were incorporated in the modeling procedure. Finally, two well-defined and effective solution procedures with appropriate diversification and intensification mechanisms were proposed, which were able to solve real large-size instances of such a complex problem. The efficiency of the proposed algorithms was evaluated by comparing their solutions with the exact solutions obtained by solving the MILP on a set of small-size test problems. Although both methods performed well in small sizes of the problem, the developed simulated annealing algorithm outperformed genetic algorithm in terms of CPU-time and accuracy measures in real-case based instances. Sufficient amount of analysis, including an analysis of variance (ANOVA), was carried out to show the superiority of the proposed SA method in solving large-size problems. For future studies on the studied problem, the following recommendations can be made:

- Extension of the problem to the case that includes emergency patients as well as elective patients;
- Consideration of the uncertainty in the arrival time of patients and the duration of surgery;
- Developing a more realistic model that takes some other constraints into account related to the nursing staff, correlation between the operating theatre and other wards like intensive care unit, etc.

## References

- Jebali, A., Hadj Alouane, A.B. and Ladet, P. "Operating rooms scheduling", *Int. J. of Prod. Econ.*, **99**(1), pp. 52-62 (2006).
- Cardoen, B., Demeulemeester, E. and Beliën, J. "Sequencing surgical cases in a day-care environment: an exact branch-and-price approach", *Comput. and Oper. Res.*, **36**(9), pp. 2660-2669 (2009).
- Denton, B.T., Miller, A.J., Balasubramanian, H.J. and Huschka, T.R. "Optimal allocation of surgery blocks to operating rooms under uncertainty", *Oper. Res.*, **58**(4-Part-1), pp. 802-816 (2010).
- Batun, S., Denton, B.T., Huschka, T.R. and Schaefer, A.J. "Operating room pooling and parallel surgery processing under uncertainty", *INFORMS J. on Comput.*, **23**(2), pp. 220-237 (2011).
- Harper, P.R. "A framework for operational modeling of hospital resources", *Health Care Manag. Sci.*, **5**(3), pp. 165-173 (2002).
- Guinet, A. and Chaabane, S. "Operating theatre planning", *Int. J. of Prod. Econ.*, **85**(1), pp. 69-81 (2003).
- Ogulata, S.N. and Erol, R. "A hierarchical multiple criteria mathematical programming approach for scheduling general surgery operations in large hospitals", *J. of Med. Syst.*, **27**(3), pp. 259-270 (2003).
- Marcon, E., Kharraja, S. and Simonnet, G. "The operating theatre planning by the follow-up of the risk of no realization", *Int. J. of Prod. Econ.*, **85**(1), pp. 83-90 (2003).
- Denton, B.T. and Gupta, D. "A sequential bounding approach for optimal appointment scheduling", *IIE Trans.*, **35**(11), pp. 1003-1016 (2003).
- Sciomachen, A., Tanfani, E. and Testi, A. "Simulation models for optimal schedules of operating theatres", *Int. J. of Simul.*, **6**(12-13), pp. 26-34 (2005).
- Pérez Gladish, B., Arenas Parra, M., Bilbao Terol, A. and Rodríguez Uría, M.V. "Management of surgical waiting lists through a possibilistic linear multi-objective programming problem", *Appl. Math. and Comput.*, **167**(1), pp. 477-495 (2005).
- Marcon, E. and Dexter, F. "Impact of surgical sequencing on post anesthesia care unit staffing", *Health Care Manag. Sci.*, **9**(1), pp. 87-98 (2006).
- Van der Lans, M., Hans, E., Hurink, J.L., Wullink, G., van Houdenhoven, M. and Kazemier, G. "Anticipating urgent surgery in operating room departments", *Tech. Rep.*, WP-158 (2006).
- Denton, B.T., Viapiano, J. and Vogl, A. "Optimization of surgery sequencing and scheduling decisions under uncertainty", *Health Care Manag. Sci.*, **10**(1), pp. 13-24 (2007).
- Gupta, D. "Surgical suites' operations management", *Prod. and Oper. Manag.*, **16**(6), pp. 689-700 (2007).
- Lamiri, M., Dréo, J. and Xie, X. "Operating room planning with random surgery times", *Automation Science and Engineering, IEEE International Conference on*, Scottsdale, AZ, pp. 521-526 (2007).
- Santibáñez, P., Begen, M. and Atkins, D. "Surgical block scheduling in a system of hospitals: An application to resource and wait list management in a British Columbia health authority", *Health Care Manag. Sci.*, **10**(3), pp. 269-282 (2007).
- Testi, A., Tanfani, E. and Torre, G. "A three-phase approach for operating theatre schedules", *Health Care Manag. Sci.*, **10**(2), pp. 163-172 (2007).
- VanBerkel, P.T. and Blake, J.T. "A comprehensive simulation for wait time reduction and capacity planning applied in general surgery", *Health Care Manag. Sci.*, **10**(4), pp. 373-385 (2007).
- Beliën, J. and Demeulemeester, E. "A branch-and-price approach for integrating nurse and surgery scheduling", *Eur. J. of Oper. Res.*, **189**(3), pp. 652-668 (2008).
- Lamiri, M., Xie, X., Dolgui, A. and Grimaud, F. "A stochastic model for operating room planning with elective and emergency demand for surgery", *Eur. J. of Oper. Res.*, **185**(3), pp. 1026-1037 (2008).
- Lamiri, M., Xie, X. and Zhang, S. "Column generation approach to operating theater planning with elective and emergency patients", *IIE Trans.*, **40**(9), pp. 838-852 (2008).
- Fei, H., Chu, C., Meskens, N. and Artiba, A. "Solving surgical cases assignment problem by a branch-and-price approach", *Int. J. of Prod. Econ.*, **112**(1), pp. 96-108 (2008).
- Pham, D.N. and Klinkert, A. "Surgical case scheduling as a generalized job shop scheduling problem", *Eur. J. of Oper. Res.*, **185**(3), pp. 1011-1025 (2008).
- Cardoen, B. and Demeulemeester, E. "Capacity of clinical pathways - a strategic multi-level evaluation tool", *J. of Med. Syst.*, **32**(6), pp. 443-452 (2008).
- Cardoen, B., Demeulemeester, E. and Beliën, J. "Optimizing a multiple objective surgical case sequencing problem", *Int. J. of Prod. Econ.*, **119**(2), pp. 354-366 (2009).
- Beliën, J., Demeulemeester, E. and Cardoen, B. "A decision support system for cyclic master surgery scheduling with multiple objectives", *J. of Sched.*, **12**(2), pp. 147-161 (2009).
- Fei, H., Meskens, N. and Chu, C. "A planning and scheduling problem for an operating theatre using an open scheduling strategy", *Comput. and Ind. Eng.*, **58**(2), pp. 221-230 (2010).
- Min, D. and Yih, Y. "An elective surgery scheduling problem considering patient priority", *Comput. and Oper. Res.*, **37**(6), pp. 1091-1099 (2010).

30. Min, D. and Yih, Y. "Scheduling elective surgery under uncertainty and downstream capacity constraints", *Eur. J. of Oper. Res.*, **206**(3), pp. 642-652 (2010).
31. Tan, Y., El Mekkawy, T.Y., Peng, Q. and Openheimer, L. "Mathematical programming for the scheduling of elective patients in the operating room department", In *Proc. of the 2007 CDEN and CCEE Conference* (2007).
32. Mancilla, C. and Storer, R. "A sample average approximation approach to stochastic appointment", *IIE Trans.*, **44**(8), pp. 655-670 (2012).
33. Saremi, A., Jula, P., ElMekkawy, T. and Wang, G. "Appointment scheduling of outpatient surgical services in a multistage operating room department", *Int. J. of Prod. Econ.*, **141**(2), pp. 646-658 (2013).
34. Agnetis, A., Coppi, A., Corsini, M., Dellino, G., Meloni, C. and Pranzo, M. "A decomposition approach for the combined master surgical schedule and surgical case assignment problems", *Health Care Manag. Sci.*, **17**(1), pp. 49-59 (2014).
35. Aringhieri, R., Landa, P., Soriano, P., Tànfani, E. and Testi, A. "A two level metaheuristic for the operating room scheduling and assignment problem", *Comput. and Oper. Res.*, **54**, pp. 21-34 (2015).
36. Dios, M., Molina-Pariente, J.M., Fernandez-Viagas, V., Andrade-Pineda, J.L. and Framinan, J.M. "A decision support system for operating room scheduling", *Comput. and Ind. Eng.*, **88**, pp. 430-443 (2015).
37. Durán, G., Rey, P.A. and Wolff, P. "Solving the operating room scheduling problem with prioritized lists of patients", *Ann. of Oper. Res.*, pp. 1-20 (2016).

## Biographies

**Maryam Haghi** received her BSc degree in Industrial Engineering from Amirkabir University of Technology in 2012 and her MSc degree in Industrial Engineering from Sharif University of Technology in 2014. Her

present research interests include stochastic optimization and healthcare management.

**Seyyed Mohammad Taghi Fatemi Ghomi** was born in Ghom, Iran, on March 11, 1952. He received his BSc degree in Industrial Engineering from Sharif University of Technology, Tehran, in 1973, and the PhD degree in Industrial Engineering from University of Bradford, England, in 1980. He worked as planning and control expert in the group of construction and cement industries, a group of Organization of National Industries of Iran, during the years 1980-1983. Also, he founded the Department of Industrial Training in the aforementioned organization in 1981. He joined Amirkabir University of Technology in Tehran, Iran, as a faculty member in 1983. He is the author and co-author of more than 360 technical papers and the author of six books in the area of industrial engineering. He is currently Professor in the Department of Industrial Engineering at Amirkabir University of Technology, Tehran, Iran. Amirkabir University of Technology recognized him as one of the best researchers of the years 2004 and 2006. Also, Ministry of Science and Technology recognized him as one of the best professors of Iran in the year 2010.

His research and teaching interests include stochastic activity networks, production planning and control, scheduling, queuing systems, and statistical quality control.

**Pedram Hooshangi-Tabrizi** received his BSc and MSc degrees in Industrial Engineering from Amirkabir University of Technology in 2010 and 2013, respectively. His areas of research include Flexible Manufacturing Systems (FMSs), healthcare optimization, resource scheduling problems, mathematical modeling, and designing algorithms to solve large-scale problems.