Computers and Operations Research 87 (2017) 270–282

Contents lists available at ScienceDirect

Computers and Operations Research

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

A hybrid genetic approach for solving an integrated multi-objective operating room planning and scheduling problem

Rosita Guido*, Domenico Conforti

de-Health Lab – Laboratory of Decision Engineering for Healthcare Services Department of Mechanical, Energy and Management Engineering, University of Calabria, Ponte Pietro Bucci 41C, Rende (Cosenza 87036), Italy

A R T I C L E I N F O

Article history:
Received 12 June 2015
Revised 3 August 2016
Accepted 7 November 2016
Available online 19 November 2016

Keywords:
Multicriteria optimization
Integer linear programming
Genetic algorithms
Optimization in healthcare services

A B S T R A C T

In this paper, we propose a multi-objective integer linear programming model aiming at efficiently planning and managing hospital operating room suites. By effectively exploiting a novel hybrid genetic solution approach, the devised optimization model is able to determine, in an integrated way, (i) the operating room time assigned to each surgical specialty, (ii) the operating room time assigned to each surgical team, (iii) the surgery admission planning and (iv) the surgery scheduling. The resulting Pareto frontiers provide a set of "optimal" solutions able to support hospital managers in efficiently orchestrating the involved resources and planning surgeons and surgeries. On this basis, the proposed solution framework could represent a suitable engine for the development of advanced and effective health care management decision support systems.

1. Introduction

In the wide context of hospital healthcare activities, complex decision making processes strongly affect operating theatre (OT) organization and management, high dynamic environments where the first priority is to ensure a safe and efficient surgery scheduling. It is well known that operating room (OR) planning and scheduling processes are rather complex and particularly expensive in terms of involved procedures and resources [1]. Usually, hospital managers strive to satisfy conflicting objectives such as efficiency, OR team waiting, OR idling, overtime, quality of care, quality of labor, being aware that achieving all is quite difficult or sometimes impossible. This aspect could explain why waiting, delays, and cancellations seem often intrinsic in OR activities. The key to a successful OR management lies in satisfying at the best the aforementioned concerns. Devising and developing procedures for an optimal surgery planning and scheduling is thus an important task within an OT. Even though most decisions are ruled by simple strategies based on "common sense" [2,3], in the last decade there has been an increasing interest in developing quantitative approaches mainly to handle efficiently surgery planning and scheduling problems. Mathematical models and methods of Operational Research have been successfully applied to this field, as reported in recent literature surveys [3,4]. The relevant literature is rather vast and the list reported by Cardoen et al. [5] gives an idea of the amount of papers about this topic. Here, we only depict some key characteristics concerning OR planning and scheduling problems; further details can be found in several papers and recent surveys [e.g. [3–5]].

Surgeries (or surgical cases) are usually grouped into three categories [6]: emergency surgeries, urgent surgeries, and elective surgeries. Only elective surgeries can be planned in advance because emergency and urgent cases have to be treated within a few hours of their arrival at the hospital. In order to define tight OR schedules, non-elective surgeries should be carefully considered; usually, they are scheduled by adopting local policies and procedures.

OR scheduling systems can be classified as open scheduling, block scheduling, and hybrid scheduling. The main difference between open and block scheduling system is that the first aims to allocate surgery time to the first surgeon who requests it, whereas in a block scheduling system an OR block is assigned exclusively to a single surgical group, composed of one or multiple surgeons but of the same surgical specialty (SS).

In the relevant literature, the following three hierarchical decision levels are usually distinguished:

- *Strategic* level, also referred to as session planning problem. One of the main goals is OR time distribution among SSs [e.g., 7] by finding an efficient surgery mix. The planning horizon is usually one/two years; decisions are based on historical data and/or demand forecast.

* Corresponding author.
E-mail addresses: rosita.guido@unical.it (R. Guido), mimmo.conforti@unical.it (D. Conforti).

http://dx.doi.org/10.1016/j.cor.2016.11.009

- *Tactical* level concerns the development of a timetable (the so called Master Surgical Schedule, MSS) that defines number and type of available ORs, open time of ORs, and surgeons to whom the OR time is assigned [8,9]. Usually an MSS is built for periods ranging from few weeks to one year, based on the availability of surgeons and historical demand patterns. Waiting list and requests for surgery are input data.
- *Operational* level concerns a detailed planning of each elective case: case assignment to OR (i.e., surgery date) and sequence (i.e., start and end time) of the scheduled cases [10–12].

Each level is characterised by different resource allocation problems, proper goals, constraints, and requirements [13,14]. The goal of the strategic decision level basically is a budget distribution and resource utilization among the sharing SSs. The main goal of the tactical level is to allocate OR time to each surgeon/specialty [8,15]. An equitable MSS assigns OR time to SSs on the basis of their specific requirements and penalizes undersupply [16]. Usually, a developed MSS is used until there are major changes in demand or surgical groups. Finally, the main objectives pursued at operational level concern hospitalisation costs, surgery's waiting time, OR utilisation, over time cost, patient and personnel preferences. Marques et al. [17] propose an integer linear programming model for weekly advanced and allocation scheduling of elective surgeries with the goal of maximizing surgical suite occupation. They define constraints on daily and weekly operating time limit for each surgeon as such those in Conforti et al. [18]. The same authors develop a solution approach based on genetic algorithms [19] but no detail on the solved instances has been reported.

Typically, the most common approach proposed in the relevant literature is hierarchical for reducing the complexity of the whole problem. Consequently, both the assignment problem of ORs to SSs and the patients scheduling problem are treated and solved separately. Only a few papers dealt with more than one decision level at the same time, as underlined in [4]. Testi et al. [20] address the three hierarchical decision levels for only one surgical specialty by solving three optimization problems, one for every decision level. The tactical and operational levels have been considered in [21] with the goal of maximizing overall social benefit (measured in terms of reduced waiting times) and production cost. Agnetis et al. [22] propose a decomposition approach for solving tactical and operation decision level at the same time and compare it with an exact integrated approach.

We remark that although it has been recognized the usefulness of considering conflicting goals in OR planning and scheduling problems often these goals are reduced to a single one. Indeed, some authors [e.g., [20,23–25]] have developed multi-objective mathematical programming models but their solution approaches are based on only a single objective. Weighted-sum method, ϵ-constraint method, bounded objective function methods, goal programming method [26], hierarchical solution approaches [e.g., 27] are some examples of the methods applied for combining the objective functions into only one. A disadvantage is the dependency from the user's decision who assigns weight coefficients to objective functions. In such a way, it must know the importance of every objective function when they are combined into a single one; moreover, the user may change the weight values for defining several solutions.

On the basis of what has been remarked and despite OT planning and scheduling problem has attracted much attention, there are still some interesting open challenges: from a patient perspective, it has been recognised that reducing the time spent in a waiting list is important. However, most researchers based their work almost primarily on the reduction of patients waiting time on the day of surgery although "the time spent in waiting lists may be more important from the health outcomes perspective" [28]. Secondly, the majority of papers splits the OR management problem in a planning phase and a scheduling phase; each phase is thus solved separately although all decision levels strongly interact with each other. Thirdly, few authors have considered the priority of patients in the scheduling decision level even if the priority-setting is a challenging aspect in health care systems [29]. Moreover, many optimization models are based on a single objective and a few papers applied evolutionary multi-objective optimization techniques, as underlined in [4,5].

Our work tries to deal with all these challenges by considering at the same time conflicting goals relating to hospital managers, surgeons and patients. Then we define a multi-objective integer linear programming model to support OR manager in decision making by integrating tactical and operational decision levels. In particular, here we extend the work proposed by Conforti et al. [18] by defining schedules which are trade-offs among underutilization of OR capacity, balanced distribution of OR time among surgeon groups, minimization of surgeries' waiting times and overtime working hours. The Pareto solutions guarantee that no criterion can be improved without deteriorating another criterion. To this end, for finding non-dominated Pareto solutions in reasonable times, we present a hybrid approach based on genetic algorithms. The alternative solutions differ in cost saving and care delivery. To the best of our knowledge (1) a few papers deal with more than one decision level and (2) only one paper [18] constructs a Pareto frontier by using a genetic algorithm.

The paper is organized as follows. The multi-objective integer linear programming model (MILPM) formulation is presented in Section 2. In Section 3, we provide a short background on genetic algorithms and propose a hybrid approach aiming to improve the convergence in finding Pareto solutions. Computational results are reported and discussed in Section 4. Finally, a conclusion section completes the paper.

2. Operating room planning and scheduling: proposed management approach and model formulation

In multi surgical specialties and multiple OR settings, it is difficult for a decision maker to know on beforehand how surgeries can be scheduled once the OR time has been allotted to SSs. Assigning OR time to SSs is not an easy problem for two main reasons: (1) target allocation to a surgical group may not be represented as a multiple of whole blocks and it is not possible to divide a given block; (2) supply of staffed ORs and/or specialty equipment may restrict the amount of time that can be assigned to a surgical group. Usually, OR capacity is allocated on the basis of historical use of OR time. An OR block is thus assigned to a surgical group that selects surgeries. Such a management could cause long waiting time for some surgeries due to the fulfilling of an OR block; moreover, it cannot take into account surgery waiting time of SSs sharing ORs.

Typically, OR managers define "best" trade-offs among costs, quality, performance through a reviewing process of the current balance of surgical demands, resources, and needs. Known the OR availability along a defined planning horizon, the first problem is to define a fair distribution of time among the several SSs and their surgical teams; then, the scheduling surgeries problem can be dealt with. Here, we address these distinct problems at the same time in an OT with multifunctional ORs, by developing and exploiting a MILPM able to determine:

1. the case mix planning, that is the OR time assigned to each SS;
2. the OR time assigned to each surgical team on the basis of equity criteria and status of waiting surgeries;
3. the surgery admission planning, that is the set of scheduled surgeries;

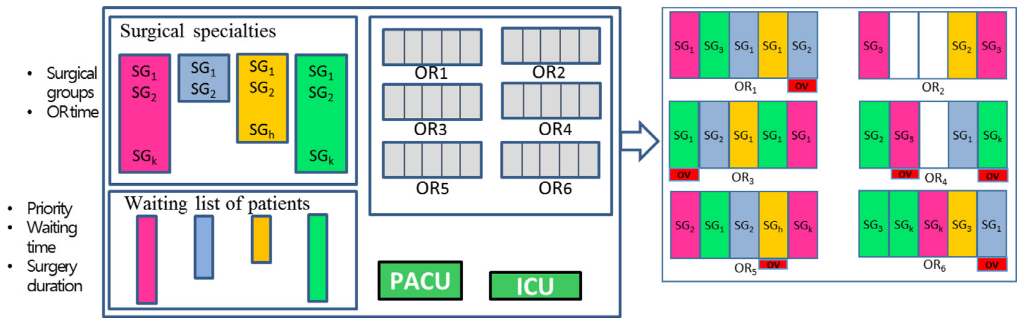

Fig. 1. Example of a synoptic schema of an OT: (left) there are four surgical specialties, their surgical groups and waiting lists of patients, six ORs and available OR blocks; (right) assignment of the surgical groups to the OR blocks. Some available overtime blocks have been allotted (red blocks denoted with *ov*). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

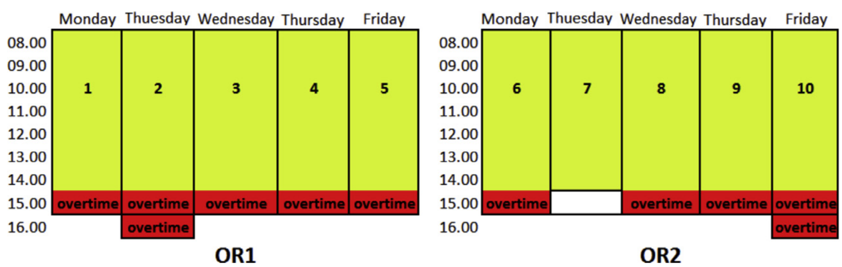

Fig. 2. Example of two operating rooms with OR blocks, set of overlapping blocks $\tilde{\mathcal{B}} = \{\{1, 6\}, \{2, 7\}, \{3, 8\}, \{4, 9\}, \{5, 10\}\}$, and possible overtime blocks.

4. the surgery scheduling, that is the surgery date and the assigned OR.

On this basis, the MILPM aims at managing an OR suite in a block scheduling system by considering that every SS is characterised by a minimum and maximum OR time allowable during the planning horizon, and by a set of surgical groups. Each surgical group has a proper waiting list of surgeries. Moreover, every surgery has a defined priority, maximum waiting time and estimated duration. The available OR blocks and possible overtime blocks are known. The goal is to define an MSS and the assignment of surgeons and their surgeries to OR blocks by optimizing conflicting objectives.

Fig. 1shows a schematization of an OT where there are 6 ORs shared among 4 SSs and each SS has a set of surgical groups. A schedule is a solution that defines both the assignment of a surgeon to the OR blocks and the assignment of surgeries to these OR blocks. A schedule is represented in Fig. 1 in terms of number of OR blocks assigned to an SS (OR blocks with the same colour) and to each surgical group/surgeon of an SS. It is evident that the OR time assigned to an SS can change dynamically on the basis of the specific status of waiting patients.

2.1. Problem statement

An OR block denotes the time interval, along a day, during which an OR is available for surgery. Just for simplifying the notation and the formulation of the model (by reducing the number of discrete structures to be considered), we combine the available blocks of each OR along the planning horizon. The overlapping blocks are identified and collected in a suitable set, namely $\tilde{\mathcal{B}}$, in order to avoid inconsistent assignments for each surgical group. An example with two ORs, ten OR blocks and some overtime blocks is showed in Fig. 2. Each weekday has a proper set $\tilde{\mathcal{B}}$ of overlapping blocks. The five sets of overlapping blocks $\tilde{\mathcal{B}}$ define $\tilde{\mathcal{B}} = \{\{1, 6\}, \{2, 7\}, \{3, 8\}, \{4, 9\}, \{5, 10\}\}$.

Before describing the optimization model formulation, we detail the assumptions, listed in Table 1.

Table 1
Assumptions.

1. All data are available and known before the planning and scheduling phase.
2. All patients are elective (emergency and urgent surgeries have dedicated ORs).
3. The assigned surgeon is known for each patient. Patients have to undergo one surgery or consecutive surgeries, which have to be performed by the same surgeon.
4. Multiple and multifunctional ORs are shared among different SSs.
5. Resources for anaesthetic procedures are available in each OR, hence every surgery can be performed in any OR (in case, restrictions can be easily defined).
6. Each OR has a defined number of available OR blocks during the planning horizon.
7. Nurses, support personnel and instrumental resources are available and well trained to respond to needed tasks.
8. The only resource constraints are due to opening hours of ORs and availability of surgeons/surgical groups.
9. Bed resource and capacity of recovery area (such as post anaesthesia centre unit and intensive care units) are available.
10. Once a surgery is started, it cannot be interrupted.
11. Within the planning horizon, for each SS, the following data are known:
(i) minimum and maximum OR time to be assigned,
(ii) number of surgeons/teams, their availability and maximum working time.
12. Each surgery is characterized by clinical priority category (e.g., high, medium, and low), elapsed waiting time from referral date, and maximum acceptable waiting time (in days).
13. All surgeons/teams of a given SS have the same expertise.

Every SS has a set of surgeons/surgical groups and each of them has a list of waiting patients. Patients have to undergo one surgery or consecutive surgeries, which have to be performed by the same surgeon. A patient is scheduled if all his/her related surgeries are scheduled. Patients that have to undergo surgeries of different SSs along the planning horizon are not here considered. This specific case add complexity because of precedence constraints among patient's surgeries, coordination among involved surgeons of the SSs, and inconsistent patient assignments. However, our approach is still valid if a patient requires more than one main surgery. Indeed, if he/she has been scheduled and operated for a given surgery and can be scheduled for a next one, the patient is inserted in

Table 2
Sets, parameters and variables of the multi-objective optimization model.

h length (in days) of the planning horizon $\mathcal{H}$	
OR blocks	Surgical specialties
$\mathcal{B}$ set of all available OR blocks	$\mathcal{S}$ set of surgical specialties
$\tilde{\mathcal{B}}$ set of overlapping blocks	For each $s \in \mathcal{S}$:
d_b length of OR block $b \in \mathcal{B}$ (in min)	$Tmin^s$ minimum allowable OR time (in min)
C_{ov} overtime penalty cost	$Tmax^s$ maximum allowable OR time (in min)
d^b_{ov} length of overtime block of OR block $b \in \mathcal{B}$ (in min)	$\mathcal{T}_s$ set of surgical teams available
Surgical team	Surgeries
$Tmax^s_t$ maximum allowable OR time for team $t \in \mathcal{T}_s, s \in \mathcal{S}$ (in min)	Let p be a patient in $\mathcal{L}^s_t, s \in \mathcal{S}, t \in \mathcal{T}_s$:
$\mathcal{L}^s_t$ waiting list of patients assigned to team $t \in \mathcal{T}_s, s \in \mathcal{S}$	sd^s_{pt} expected surgery duration (in min)
$\mathcal{P}_s = \bigcup_{t \in \mathcal{T}_s} \mathcal{L}^s_t, s \in \mathcal{S}$ is the set of waiting patients of SS s	pr^s_{pt} clinical priority value
	et^s_{pt} elapsed waiting time (in days) from referral date
	mt^s_{pt} maximum acceptable waiting time (in days)
	$l^s_{pt} = et^s_{pt} - mt^s_{pt}$ lateness (in days)
	$\tilde{\mathcal{L}}^s_t = \left\{p \in \mathcal{L}^s_t : l^s_{pt} \geq -h\right\}$ set of patients that should be scheduled in the planning horizon
	$w^s_{\tilde{p}t}$ weight coefficient defined for patient $\tilde{p} \in \tilde{\mathcal{L}}^s_t$.
Binary decision variables	
y^s_{tpb} =1 if team $t \in \mathcal{T}_s$ operates patient $p \in \mathcal{L}^s_t$ in OR block b; 0 otherwise	
x^s_{tb}=1 if team $t \in \mathcal{T}_s$ works on OR block b; 0 otherwise	
o^s_{tb} =1 if team $t \in \mathcal{T}_s$ has overtime block of OR block b; 0 otherwise	

an appropriate surgical specialty waiting list. His/her priority value and elapsed waited time, which influence patient scheduling results, are defined accordingly. In the following of the paper, patient scheduling is equivalent to surgery scheduling.

The maximum working time defined for each surgeon/surgical group during the planning horizon depends on the hours actually worked before. The assignment of surgical cases has to comply with the opening hours of ORs, the availability of surgeons and their work time limit. Hard constraints are rigidly enforced: no resource (i.e., surgeons and their staff) can be demanded to be in more than one place at the same time (inconsistent assignment), and all surgeries booked into an OR block have to be completed within it. Overtime blocks can be booked by respecting the maximum work time defined for each surgeon and each SS.

The general goal to be pursued is the improvement of efficiency and effectiveness of the entire system, by trading-off the conflicting relevant objectives of the main involved stakeholders. These concern the efficient utilization of ORs shared among surgeon groups (by reducing the underutilization of OR blocks), the improvement of surgery flow throughput (by reducing waiting time and delivering surgery at the right time) and the parsimonious use of overtime. Hence, an MSS is created by avoiding inconsistent assignments to overlapping blocks, by taking into account surgery characteristics (clinical priority, maximum waiting time, elapsed waiting time, estimated operating time), and the maximum OR time fixed both for each SS and surgeon. The resulting optimal schedules (Pareto solutions) allow evaluating trade-offs among the considered objectives.

2.2. Optimization model formulation

We set out in Table 2 the notation (i.e., sets, parameters and binary decision variables for each surgical specialty). $\mathcal{S}$ denotes the set of surgical specialties indexed by s. $\mathcal{T}_s$, indexed by t, denotes the set of surgeons of s, and $\mathcal{L}^s_t$ is the set of patients assigned to surgeon group $t \in \mathcal{T}_s$. $\mathcal{P}_s = \bigcup_{t \in \mathcal{T}_s} \mathcal{L}^s_t$ is the set of all waiting patients of surgical specialty s. Each patient $p \in \mathcal{L}^s_t, s \in \mathcal{S}, t \in \mathcal{T}_s$ has a maximum acceptable waiting time (in days) defined by mt^s_{pt}. We introduce the lateness parameter $l^s_{pt} = et^s_{pt} - mt^s_{pt}$, for taking into account the patient's elapsed waiting time et^s_{pt} (in days). l^s_{pt} can take any integer value (positive or negative) and is evaluated whenever a scheduling procedure has to be realized. Let $\mathcal{H}$ be the planning horizon and $|\mathcal{H}| = h$ the number of days. Patient p should be scheduled in $\mathcal{H}$ if $l^s_{pt} \geq -h$: indeed, the defined maximum acceptable waiting time for patient p will be attained within $\mathcal{H}$ if $-h \leq l^s_{pt} \leq 0$, whereas it has been already attained in the past if $l^s_{pt} > 0$ (positive lateness is tardiness). In both cases, p is inserted into the set $\tilde{\mathcal{L}}^s_t$. $\tilde{\mathcal{P}}_s = \bigcup_{t \in \mathcal{T}_s} \tilde{\mathcal{L}}^s_t$ denotes the set of all patients of surgical specialty s that should be operated during the planning horizon $\mathcal{H}$. Note that $\tilde{L}^s_t \subseteq L^s_t$ and to simplify the MILPM, $\tilde{L}^s_t$ is in the head of L^s_t.

The multi-objective optimization model formulation is reported in the following.

$$
\begin{aligned}
&opt\, f(x) = [f_1(x), f_2(x), f_3(x), f_4(x), f_5(x)] \\
&= \max \sum_{s \in \mathcal{S}} \sum_{t \in \mathcal{T}_s} \sum_{p \in \mathcal{L}^s_t} \sum_{b \in \mathcal{B}} y^s_{tpb} \\
&\quad \max \sum_{s \in \mathcal{S}} \sum_{t \in \mathcal{T}_s} \sum_{p \in \mathcal{L}^s_t} \sum_{b \in \mathcal{B}} pr^s_{pt} y^s_{tpb} \\
&\quad \max \sum_{s \in \mathcal{S}} \sum_{t \in \mathcal{T}_s} \sum_{p \in \tilde{\mathcal{L}}^s_t} \sum_{b \in \mathcal{B}} w^s_{\tilde{p}t} y^s_{t\tilde{p}b} \\
&\quad \min \sum_{s \in \mathcal{S}} \sum_{t \in \mathcal{T}_s} \sum_{b \in \mathcal{B}} (d_b x^s_{tb} + d^b_{ov} o^s_{tb}) - \sum_{s \in \mathcal{S}} \sum_{t \in \mathcal{T}_s} \sum_{p \in \mathcal{L}^s_t} \sum_{b \in \mathcal{B}} sd^s_{pt} y^s_{tpb} \\
&\quad \min \sum_{s \in \mathcal{S}} \sum_{t \in \mathcal{T}_s} \sum_{b \in \mathcal{B}} C_{ov} o^s_{tb}
\end{aligned} \tag{1}
$$

$$\sum_{b \in \mathcal{B}} y^s_{tpb} \leq 1 \qquad \forall s \in \mathcal{S}, t \in \mathcal{T}_s, p \in \mathcal{L}^s_t \tag{2}$$

$$\sum_{b \in \tilde{B}} x^s_{tb} \leq 1 \qquad \forall s \in \mathcal{S}, t \in \mathcal{T}_s, \tilde{B} \in \tilde{\mathcal{B}} \tag{3}$$

$$\sum_{s \in \mathcal{S}} \sum_{t \in \mathcal{T}_s} x^s_{tb} \leq 1 \qquad \forall b \in \mathcal{B} \tag{4}$$

$$\sum_{t \in \mathcal{T}_s} \sum_{b \in \mathcal{B}} (d_b x^s_{tb} + d^b_{ov} o^s_{tb}) \geq Tmin^s \qquad \forall s \in \mathcal{S} \tag{5}$$

$$\sum_{t \in \mathcal{T}_s} \sum_{b \in \mathcal{B}} (d_b x^s_{tb} + d^b_{ov} o^s_{tb}) \leq Tmax^s \qquad \forall s \in \mathcal{S} \tag{6}$$

$$\sum_{b \in \mathcal{B}} (d_b x^s_{tb} + d^b_{ov} o^s_{tb}) \leq Tmax^s_t \qquad \forall s \in \mathcal{S}, t \in \mathcal{T}_s \tag{7}$$

$$\sum_{p\in\mathcal{L}_t^s} sd_{pt}^s y_{tpb}^s \quad \le d_b x_{tb}^s + d_{ov}^b o_{tb}^s \qquad \forall s\in\mathcal{S}, t\in\mathcal{T}_s, b\in\mathcal{B} \tag{8}$$

$$o_{tb}^s \quad \le x_{tb}^s \qquad \forall s\in\mathcal{S}, t\in\mathcal{T}_s, b\in\mathcal{B} \tag{9}$$

$$y_{tpb}^s = y_{t\tilde{p}b}^s \qquad \forall s\in\mathcal{S}, t\in\mathcal{T}_s, p\in\tilde{\mathcal{L}}_t^s, b\in\mathcal{B} \tag{10}$$

$$y_{tpb}^s \in \{0,1\} \qquad \forall s\in\mathcal{S}, t\in\mathcal{T}_s, p\in\mathcal{L}_t^s, b\in\mathcal{B} \tag{11}$$

$$x_{tb}^s \in \{0,1\} \qquad \forall s\in\mathcal{S}, t\in\mathcal{T}_s, b\in\mathcal{B} \tag{12}$$

$$o_{tb}^s \in \{0,1\} \qquad \forall s\in\mathcal{S}, t\in\mathcal{T}_s, b\in\mathcal{B} \tag{13}$$

The first objective function maximizes the number of scheduled patients and the second one maximizes the overall clinical priority of the scheduled patients. The third objective function aims to maximize the overall priority of those patients that should be scheduled during the planning period (i.e. those in $\hat{\mathcal{P}}_s, \forall s\in\mathcal{S}$). The fourth objective function minimizes the underutilization of the assigned OR blocks (defined as the difference between the overall OR time assigned to the surgical teams and the overall duration of scheduled surgeries). Finally, the last one minimizes the overall overtime cost. Constraints (2) assure that every patient is assigned at most to one OR block during the planning horizon. Constraints (3) avoid inconsistent assignments: each surgical group is assigned at most to one OR block among those listed in each subset $\tilde{B}\in\hat{\mathcal{B}}$; Constraints (4) assure that each OR block is assigned at most to one surgical group. Constraints (5)-(7) set balancing admission conditions among SSs by levelling of workload: the overall OR time assigned to a surgical specialty s is at least equal to $Tmin^s$ and cannot be greater than $Tmax^s$ (Constraints (5) and (6), respectively). Constraints (7) assure that the OR time assigned to each surgical group is at most equal to $Tmax_t^s$. Constraints (8) define that the overall surgery time in an OR block cannot exceed the OR block capacity plus an overtime block (if it is allowed). Obviously, an overtime block is assigned to a surgical team only if the related OR block has been allotted to it (Constraints (9)). Constraints (10) are on relations between variables related to patients. Finally, Constraints (11)–(13) impose 0-1 restrictions on the decision variables.

Under regular assumptions, the MILPM has feasible solutions. In fact, let $\overline{\mathcal{SS}}\subseteq\mathcal{SS}$ be the set of SSs having $Tmin^s > 0$. It is easy to verify that a feasible solution exists if and only if the following conditions are satisfied:

1. $|\mathcal{B}| \ge |\overline{\mathcal{SS}}|$. Since OR blocks cannot be shared, the number of available OR blocks has to be at least equal to the number of SSs with strictly positive minimum OR time.
2. $\sum_{t\in\mathcal{T}_s} Tmax_t^s \ge Tmin^s, \forall s\in\overline{\mathcal{SS}}$, that is the overall available OR time of surgeons of each surgical specialty $s\in\overline{\mathcal{SS}}$ has to guarantee the minimum demanded OR time $Tmin^s$ by the surgical specialty.
3. $\min_{p\in\mathcal{P}^s} sd_p{}^s \le Tmax^s, \forall s\in\overline{\mathcal{SS}}$. If $\min_{p\in\mathcal{P}^s} sd_p{}^s \le Tmin^s$, with $Tmin^s \le Tmax^s$ then the required condition is always true.
4. It is possible to determine a set partition satisfying Constraints (3)–(5) for $\overline{\mathcal{SS}}$, otherwise the MILPM is infeasible.

The MILPM can be interpreted as an integration of (i) a generalized multiple knapsack problem where every OR block is a "knapsack" that cannot be overfilled (Constraints (8), (9) and (11)) and time resource is the capacity of each OR block; (ii) a resource allocation problem where ORs are resources and patients are objects; (iii) a one-step job shop scheduling problem with multiple machines. Constraints (2)–(9) are hard (in terms of needed requirements) and those related to the assignment of OR blocks to surgeons and possible overtime blocks increase the complexity. Moreover, MILPM (1)–(13) shows a strong flexibility because some specific constraints can be easily formulated. For instance, (i) set $x_{\bar{b}\bar{t}} = 0$ if surgeon $\bar{t}$ is unavailable during OR block $\bar{b}$; (ii) if surgeon $\bar{t}$ has to perform surgeries in specific ORs, the constraints $\sum_{b\in\mathcal{SB}_{\bar{t}}} y_{\bar{t}pb}^s \le 1$, where $\mathcal{SB}_{\bar{t}}$ is the subset of suitable OR blocks, are added and Constraints (3) and (4) modified accordingly.

Further restrictions on capacity of post anaesthesia centre unit (PACU) and intensive care units (ICU) should be defined by limiting the number of scheduled surgeries during the planning horizon.

The MILPM (1)–(13) is combinatorial and challenging due to the sparse search space. Agnetis et al. [30] pointed out that it is better to find a compromise between efficiency of a solution method and the quality of a solution. To this end, we define and implement a suitable metaheuristic approach based on genetic algorithms for finding Pareto solutions.

3. Solution approach via genetic algorithms

A general multi-objective mathematical programming formulation is

$$\max F(x)$$
$$C(x) \le 0$$
$$E(x) = 0$$

where x denotes vector variables, $F(x) = [f_1(x), f_2(x), \ldots, f_k(x)]^T$ is a vector of k objective functions, $C(x)$ and $E(x)$ denote inequalities and equality constraints, respectively.

Let n be the number of design variables. The *search space* (or design space) is the set of all possible combinations of the design variables. The *feasible domain* X is the region in the search space where all constraints are satisfied. In Pareto optimization, at each solution $x\in X$ corresponds a vector objective values $F(x) = [f_1(x), f_2(x), \ldots, f_k(x)]^T$, trade-off among the k defined objectives. Let x_1 and $x_2 \in X$: x_1 is *non-dominated* with respect to x_2 if and only if none of the k objective values evaluated in x_1 is worse than the corresponding value in x_2 and, moreover, at least one of the objective functions in x_1 is strictly better than the corresponding value in x_2. For a multi-objective mathematical programming model with only two objective functions, f_1 and f_2, a solution x_1 dominates the solution x_2 iff: $[f_1(x_1) \ge f_1(x_2) \wedge f_2(x_1) > f_2(x_2)] \vee [f_1(x_1) > f_1(x_2) \wedge f_2(x_1) \ge f_2(x_2)]$. The *Pareto optimal front* is the set of all solutions that are not dominated by any other solution in X. In general, computational methods cannot guarantee global Pareto optimality, but at best local Pareto optimality. More formally, we report the following definitions.

Definition 1. Let x_1, $x_2 \in X$ be two decision vectors. $F(x_1)$ dominates $F(x_2)$ (denoted by $F(x_1)\succ F(x_2)$) if and only if $f_i(x_1) \ge f_i(x_2), \forall i = 1,\ldots,k$ and $\exists\, j\in\{1,\ldots,k\} : f_j(x_1) > f_j(x_2)$.

Definition 2. A point x^* is defined as globally Pareto optimal if and only if there is no $x\in X$, $x\ne x^*$ such that $F(x)\succ F(x^*)$. $F(x^*)$ is globally efficient and the image of the set of globally efficient points is called Pareto front.

Definition 3. A point x^* is locally Pareto optimal if and only if there exists an open neighbourhood $B(x^*)$ of x^* such that there is no $x\in B(x^*)\bigcap X$ with $F(x)\succ F(x^*)$. $F(x^*)$ is called locally efficient and the image of the set of locally efficient points is called local Pareto front.

During the last years researchers proposed evolutionary optimization approaches (see for instance recent survey papers such as [31–33]). Neighbourhood constraint genetic algorithm [34], strength Pareto evolutionary algorithm, and elitist non-dominated sorting genetic algorithm [35] are some efficient-solving algorithms especially when traditional methods failed to provide good solutions. However, their major disadvantage is the high computational cost [36].

Genetic algorithms (GAs) are an important class of metaheuristics [37] applied to several scheduling problems but not yet extensively in operating room planning and scheduling problems. GAs were proposed by Holland [38] as adaptation procedures based on Darwin principles of natural selection. One attractive is that they are problem-independent algorithms. Solution representation and design of evolutionary operators play a crucial role for a successful application. In the following the basic terminology and the main characteristics of GAs are briefly reported. For more details, the reader is referred to specific literature [e.g., 39,40]. A solution is encoded as a so called *chromosome*. Multiple chromosomes form a GA *population* that evolves by applying ordinary genetic operators (i.e., selection, crossover, and mutation); the fittest chromosomes are selected and combined as defined by crossover and mutation law until a defined criterion is reached.

Among the several genetic algorithms we choose the Non-dominated Sorting Genetic Algorithm II (NSGA-II) because, as reported in [41], it empirically shows better performance than others. NSGA-II [35] is an improved multi-objective non-dominated sorting version of the search algorithm originally proposed by Srinivas and Deb [42]. It is one of the most used evolutionary algorithms for funding multiple Pareto-optimal solutions (or near Pareto) of a multi-objective programming model. Based on the elitist principle, it uses an explicit diversity preserving mechanism and emphasizes non-dominated solutions. The population of N chromosomes is initialized randomly. The chromosomes are then sorted and ranked on the basis of the Euclidean distance between solutions. Those solutions far away from other solutions are preferred since this preserves a no crowded solutions set. The best chromosomes are picked from the current population and put into a mating pool to which the genetic operators are applied for defining the new population. These steps are repeated until the maximum number of generations has been reached. The best solution is the set of the chromosomes with the highest ranked Pareto non-dominated set from the population. Let k be the number of objectives and l the population size. NSGA-II requires a significantly smaller number of comparisons ($O(lk^2)$) with respect to the ($O(lk^3)$) of conventional non-dominated sorting. For more details, see specific papers on NSGA-II.

3.1. Hybrid genetic algorithm

This section proposes a hybrid genetic approach. In GAs the population size depends on the population structure, scales according to a given optimization problem and larger instances require larger populations to be solved, as remarked by Laredo et al. [43]. Moreover, the computational cost of evaluating the problem also scales depending on its computational order. Rahnamayan et al. [44] considered a new initialization approach which employs an opposition based learning method for defining a new population and accelerating evolutionary algorithm. Chandoul et al. [45] developed a multi-objective scheduling problem for a surgical unit and an evolutionary approach but the instances refer only to one one-day surgical unit and the size of the instances is not reported.

For tailoring NSGA-II, we carried out some preliminary computational experiments: we observed that often NSGA-II was unable to find a feasible solution despite the high number of generations (e.g., more than 100,000). In general, this is due to the high dimension of the search space. Therefore, we devise the following hybrid approach aiming to improve the convergence of NSGA-II in finding Pareto solutions.

We define a new initialization phase where an initial population *IniPop* with N chromosomes is generated by merging two distinct sub-populations: the first sub-population, namely $IniPop_1$, has only very few chromosomes which are feasible or suboptimal solutions; the second sub-population is $IniPop_2$, which has random semi-feasible chromosomes. The so constructed initial population turns out to be more meaningful than that generated randomly by NSGA-II, since the N chromosomes are good seeds for generating a set of solutions of next population. Following the classification about the approaches that combine exact and metaheuristic algorithms for combinatorial optimization [46], our approach is a collaborative combination. It is depicted in Fig. 3. More specifically, all N_1 individuals of $IniPop_1$ are feasible solutions of the optimization model formulation (2)–(13) with single objective (only one objective function is optimized). The $N_2 = N - N_1$ remaining individuals of *IniPop* are found randomly by imposing Constraints (2) and (4) into the initialization function of NSGA-II. These N_2 chromosomes are generated randomly but they are semi-feasible solutions. The so constructed N individuals constitute the initial population for NSGA-II algorithm: on this basis, the population evolves until the defined number of generations. The construction of $IniPop_1$ is reported in Algorithm 1.

Algorithm 1: Construction of $IniPop_1$.

```
Data: N_1
Result: IniPop_1 of N_1 individuals
i:=0;
IniPop_1 = Ø;
while i!=N_1 do
    Find a chromosome_i as a feasible solution (or suboptimal)
    of single objective optimization model;
    IniPop_1 = IniPop_1 ⋃ chromosome_i;
    i:=i+1;
end
```

In general, the N_1 feasible solutions could affect the convergence properties of NSGA-II and they are also depending on the choice of the single objective function to be optimized. By the proposed “warm start” semi-random procedure, driving the initialization of the population by selecting fitter individuals, we expect to reduce the solution time.

4. Computational experiments and discussion

In this section, we assess the effectiveness and efficiency of the proposed MILPM (1)–(13) and the hybrid genetic solution approach. We firstly provide a description of the real data collected at the General Hospital of Cosenza (Italy). Realistic random instances have been generated by these real data to study more complex OTs. With the aim to investigate mutual interaction among the objective functions and to study how a management strategy affects a waiting list and planning efficiency, we carried out different experiments. Computational results are thus presented and discussed.

4.1. Data description, experiment design and model coding

The case study refers to the surgical department of the General Hospital of Cosenza (a medium size public hospital in the South of Italy), where real data have been collected. The OT consists of 3 OR departments and 9 ORs: 5 ORs are shared among 5 SSs, 2

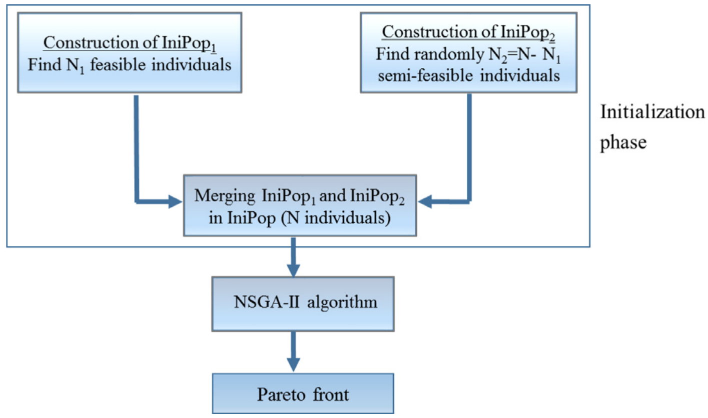

Fig. 3. Hybrid solution approach.

Table 3
Surgical specialties and surgery time (in terms of the overall estimated surgery time, mean surgery time and standard deviation), number of waiting patients.

Surgical specialty	Surgery time			Patients
	Total time (min)	Mean surgery (min)	St. dev.	$\vert\mathcal{P}\vert$
Orthopedics	3530	85.00	28.73	42
Odontostomatology	1180	45.40	31.80	26
Ophthalmology	820	37.30	10.10	22
General surgery	2880	34.28	15.63	84
Pediatrics	780	32.50	7.83	24
Urology	1260	52.50	23.79	24
Vascular	2140	76.42	19.35	26
Neurosurgery	2040	85.00	10.44	24

ORs among 3 SSs, and 2 ORs between 2 SSs; one OR is dedicated to emergencies. Every OR is open from Monday to Friday for 8 h per day with only a single block per day ($d_b = 480$ min). The collected data refer to the first two OR departments (with 5 ORs and 2 ORs, respectively) over 2 weeks (from Monday to Friday). The surgical groups of an SS carry out surgeries on some days because of other duties and contract law. In Table 3, we summarize the data for each SS: overall estimated surgery time, mean and standard deviation of surgery time, number of waiting patients. We point out that the expected surgery duration is the sum of three terms: (i) setup time for preparing OR and assembling instruments, supplies and equipment; (ii) time from when patient enters the OR until he/she leaves it and (iii) cleaning room for next surgery. These times vary with the type and complexity of a specific surgical procedure, as well as knowledge and expertise of surgeons and staff. Their estimate is thus crucial when assigning cases to an OR. Here, we have made the assumptions listed in Table 1. On this basis, we have defined nine OTs, which differ in number of available OR blocks, overall OT time (in min), sets $\tilde{\mathcal{B}}$ of inconsistent OR blocks, and number of available overtime blocks. The OTs are listed in Table 4. All OR blocks of $OT_1 - OT_6$ have the same duration (d_b=480 min); $OT_{5'}$ differs from OT_5 by OR block duration, i.e. d_b=240 min; $OT_7 - OT_9$ have two sets of OR blocks which differ by OR time, i.e., $\mathcal{B}_1$ and $\mathcal{B}_2$ with $d_b = 240, \forall b \in \mathcal{B}_1$ and $d_b = 180, \forall b \in \mathcal{B}_2$ (in this case $\mathcal{B} = \mathcal{B}_1 \bigcup \mathcal{B}_2$).

As underlined in Section 2, a defined schedule is revised throughout the day of surgery to compensate for cases that run longer or shorter than expected and the waiting list status may vary from week to week. To reduce continuous revisions, we suggest to use one week as planning horizon. We set thus $h = 7$. However, this setting does not affect the generality of the proposed approach. With the aim to extend computational results and generalize our approach, we have generated six random and realistic instances from the collected real data. These instances are summarised in Table 5: the value of $\sum_{s\in\mathcal{S}} |\tilde{\mathcal{P}}_s|$ is computed according to parameter-setting of h (i.e. one week). For each $s \in \mathcal{S}, t \in \mathcal{T}_s$ and $\tilde{p} \in \tilde{\mathcal{L}}_t^s$, we set $w_{\tilde{p}t}^s = pr_{\tilde{p}t}^s l_{\tilde{p}t}^s$, where $l_{\tilde{p}t}^s$ is set as tardiness if $l_{\tilde{p}t}^s > 0$ and 1 otherwise; $w_{\tilde{p}t}^s$ can be thus interpreted as priority value weighted by tardiness for all patients that should be operated during the planning period. Low, medium and high surgery priorities are codified as 1, 5, and 10, respectively. Constraints on weekly maximum working time and human resources are managed avoiding excessive time during the planning period. The time of all overtime blocks has been set as $d_{ov} = 60$ min and all overtime blocks have the same cost $C_{ov} = 1$.

Each surgery schedule is encoded in a binary string (chromosome) length of $m = |\mathcal{B}|(\sum_{s\in\mathcal{S}} |\mathcal{P}_s| + 2\sum_{s\in\mathcal{S}} |\mathcal{T}_s|)$. A general representation is reported in Fig. 4: for each OR block, the first $\sum_{s\in\mathcal{S}} |\mathcal{P}_s|$ positions involve the assignment of patients to the OR block; the subsequent $\sum_{s\in\mathcal{S}} |\mathcal{T}_s|$ positions involve the assignment of a surgeon to the OR block. The remaining positions involve the overtime block assignment.

MILPM (1)-(13) has been coded in C. Several values of evolutionary operators have been tested. Table 6 lists the parameter values used in our computational experiments. Observe that $mutation\ rate = 1/m$, where m is the specific string length of the related instance and the number of generations for evolving population is the stopping criterion.

4.2. Computational results and discussion

The computational experiments have been carried out by running NSGA-II [35] version 1.1.6 (available at http://www.iitk.ac.in/kangal/codes.html) and performed on a 3.60 GHz Xeon Dual Core 64-bits PC.

Referring to our hybrid approach, a feasible solution of Model (2)–(13) with f_1 as the single objective function is computed by setting $f_1 = constant$, whereas a suboptimal solution is found by

Table 4
Nine operating theatres different for the number of available OR blocks, overall OR time (in min), sets of inconsistent OR blocks, and number of available overtime blocks.

Operating theatre	$\|\mathcal{B}\|$	Time (min)	$\tilde{\mathcal{B}}$	Number of overtime blocks
OT_1	10	4800	{1, 6}, {2, 7}, {3, 8}, {4, 9}, {5, 10}	10
OT_2	15	7200	{1, 6, 11}, {2, 7, 12}, {3, 8, 13}{4, 9, 14}, {5, 10, 15}	15
OT_3	20	9600	{1, 6, 11, 16}, {2, 7, 12, 17}, {3, 8, 13, 18} {4, 9, 14, 19}, {5, 10, 15, 20}	20
OT_4	25	12000	{1, 6, 11, 16, 21}, {2, 7, 12, 17, 22}, {3, 8, 13, 18, 23}, {4, 9, 14, 19, 24}, {5, 10, 15, 20, 25}	25
OT_5	30	14400	{1, 6, 11, 16, 21, 26}, {2, 7, 12, 17, 22, 27}, {3, 8, 13, 18, 23, 28}, {4, 9, 14, 19, 24, 29}, {5, 10, 15, 20, 25, 30}	30
$OT_{5'}$	30	7200	{1, 6, 11, 16, 21, 26}, {2, 7, 12, 17, 22, 27}, {3, 8, 13, 18, 23, 28}, {4, 9, 14, 19, 24, 29}, {5, 10, 15, 20, 25, 30}	30
OT_6	40	19200	{1, 6, 11, 16, 21, 26, 31, 36}, {2, 7, 12, 17, 22, 27, 32, 37}, {3, 8, 13, 18, 23, 28, 33, 38}, {4, 9, 14, 19, 24, 29, 34, 39}, {5, 10, 15, 20, 25, 30, 35, 40}	40
OT_7	20 ($\|\mathcal{B}_1\|$=10 $\|\mathcal{B}_2\|$=10)	4200 (2400) (1800)	{1, 6, 11, 16}, {2, 7, 12, 17}, {3, 8, 13, 18} {4, 9, 14, 19}, {5, 10, 15, 20}	20
OT_8	30 ($\|\mathcal{B}_1\|$=15 $\|\mathcal{B}_2\|$=15)	6300 (3600) (2700)	{1, 6, 11, 16, 21, 26}, {2, 7, 12, 17, 22, 27}, {3, 8, 13, 18, 23, 28}, {4, 9, 14, 19, 24, 29}, {5, 10, 15, 20, 25, 30}	30
OT_9	40 ($\|\mathcal{B}_1\|$=20 $\|\mathcal{B}_2\|$=20)	8400 (4800) (3600)	{1, 6, 11, 16, 21, 26, 31, 36}, {2, 7, 12, 17, 22, 27, 32, 37}, {3, 8, 13, 18, 23, 28, 33, 38}, {4, 9, 14, 19, 24, 29, 34, 39}, {5, 10, 15, 20, 25, 30, 35, 40}	40

Table 5
Problem instances: number of surgical specialties, overall OR time, overall number of surgical groups, overall number of waiting patients, overall number of patients that should be operated during the planning period.

Instance	$\|\mathcal{SS}\|$	Overall OR time								$\sum_{s\in\mathcal{S}}\|\mathcal{T}_s\|$	$\sum_{s\in\mathcal{S}}\|\mathcal{P}_s\|$	$\sum_{s\in\mathcal{S}}\|\tilde{\mathcal{P}}_s\|$
		S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8			
I_1	2	5610	4180	–	–	–	–	–	–	4	136	70
I_2	3	5830	1190	1970	–	–	–	–	–	5	136	77
I_3	3	5601	1030	1780	–	–	–	–	–	6	136	77
I_4	3	10,820	1860	3220	–	–	–	–	–	5	250	147
I_5	4	11,420	4700	4520	7180	–	–	–	–	8	500	254
I_6	8	11,420	4700	4520	7180	11380	4500	4420	7000	16	1000	508

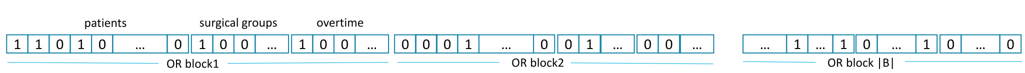

Fig. 4. Representation of a surgery schedule codified as a binary string.

Table 6
Parameter setting.

Parameters of the math model	Parameters of the NSGA-II
h=7	Crossover rate=0.9
pr={1,5,10}	Random seed=0.7
$C_{ov} = 1$	Mutation rate $= 1/m$
$d_b = 480, \forall b \in B$ (of $OT_1 - OT_6$)	$N = 200$
$d_b = 240, \forall b \in B_1$ (of $OT_{5'}$ and $OT_7 - OT_9$)	$N_1 = 4$
$d_b = 180, \forall b \in B_2$ (of $OT_7 - OT_9$)	$N_2 = 196$
$d_{ov} = 60$	Number of generations =10000

optimizing f_1 and running the general solver IBM ILOG CPLEX Optimization Studio V12.5.1 for a limited time (we set 30 s).

The proposed experimental framework is based on 9 OTs, 6 different problem instances, up to 1000 surgeries to be scheduled, and allows to develop 32 real-case scenarios spanning several characteristics in terms of complexity of the related MILPM. Table 7 summarizes our experiment designs and some details, i.e. E_{ij} denotes the couple operating theatre i and instance j, number of constraints, number of chromosomes (which reflects the number of variables), minimum and maximum required OR time by SSs, maximum OR time defined for surgeons. Observe that the $Tmin^s$, $Tmax^s$ and $Tmax_t^s$ values are equal for all involved SSs and surgeons in a given scenario, except that in $E_{55'}$. Indeed, $I_{5'}$ differs from I_5 only for $Tmin^s$, $Tmax^s$ and $Tmax_t^s$ values.

The optimization model solution yields schedules based on maximum utilization of the total OR capacity, balanced distribution of OR time among SSs and their surgical groups, minimization of waiting times of prioritized patients, and minimization of the total overtime since it means overtime work and then increased cost. Taking into account the status of each waiting patient, every schedule defines:

1. the amount of OR time allotted to every SS,
2. the OR blocks assigned to each surgeon group,
3. the set of scheduled patients for each surgical group and then for each SS,
4. the date and the OR block assigned to a scheduled patient.

A found Pareto local front represents a set of OR schedules with a trade-off among the five considered objectives. As we expect, the number of Pareto solutions, evaluated in terms of objective function value, is less than the population size N. In fact, different

Table 7
Experiment design: identifier of the experiment, number of constraints, number of bits, $Tmin^s$, $Tmax^s$, and $Tmax_t^s$ values (in min).

Exp. (OT-Instance)	Number of constraints	Length of chromosomes m	$Tmin^s$ (min)	$Tmax^s$ (min)	$Tmax_t^s$ (min)
E_{11} (OT_1 -I_1)	254	1440	480	4800	2400
E_{12} (OT_1 -I_2)	282	1460	480	4800	2400
E_{13} (OT_1 -I_3)	308	1480	480	4800	2400
E_{14} (OT_1 -I_4)	396	2600	480	4800	2400
E_{22} (OT_2 -I_2)	335	2190	480	4800	2400
E_{23} (OT_2 -I_3)	373	2220	480	4800	2400
E_{24} (OT_2 -I_4)	451	3900	480	4800	2400
E_{25}(OT_2 -I_5)	811	7740	960	4800	3200
E_{32} (OT_3 -I_2)	392	2920	480	4800	2400
E_{33} (OT_3 -I_3)	438	2960	480	4800	2400
E_{34} (OT_3 -I_4)	506	5200	480	4800	2400
E_{35} (OT_3 -I_5)	896	10320	960	4800	3200
E_{44} (OT_4 -I_4)	561	6500	480	4800	2400
E_{45}(OT_4 -I_5)	981	12900	960	4800	3200
E_{54} (OT_5 -I_4)	616	7800	480	4800	2400
E_{55}(OT_5 -I_5)	1066	15480	960	4800	3200
$E_{55'}$(OT_5 -$I_{5'}$)	1066	15480	960	1440-540-540-900	1440-1440-540-540-540-540-900-900
$E_{5'4}$ (OT_5 -I_4)	616	7800	480	4800	2400
E_{65}(OT_6 -I_5)	1236	20640	960	4800	3200
E_{66}(OT_6 -I_6)	2432	41280	240	2400	1444
E_{71}(OT_7 -I_1)	424	2880	480	4800	2400
E_{72}(OT_7 -I_2)	492	2920	480	4800	2400
E_{73}(OT_7 -I_3)	558	2960	480	4800	2400
E_{74}(OT_7 -I_4)	506	5200	480	4800	2400
E_{82}(OT_8 -I_2)	652	4380	480	4800	2400
E_{83}(OT_8 -I_3)	748	4440	480	4800	2400
E_{84}(OT_8 -I_4)	766	7800	480	4800	2400
E_{85}(OT_8 -I_5)	1306	15480	960	4800	2640
E_{93}(OT_9 -I_3)	938	5920	480	4800	2400
E_{94}(OT_9 -I_4)	926	10400	480	4800	2400
E_{95}(OT_9 -I_5)	1236	20640	960	4800	3200
$E_{95'}$(OT_9 -$I_{5'}$)	1236	20640	960	1440-540-540-900	1440-1440-540-540-540-540-900-900

Table 8
Ten non-dominated Pareto solutions for E_{11} characterised by 10 OR blocks, 2 surgical specialties (S_1 and S_2) and 136 waiting patients. In bold the best value of each objective function.

Obj. function	Schedules										min	max
	1	2	3	4	5	6	7	8	9	10		
$(max)f_1$	**89**	88	88	82	82	82	82	82	82	82	82	89
$(max)f_2$	384	383	392	488	475	483	**492**	479	484	484	383	492
$(max)f_3$	569	569	625	667	666	**683**	675	672	667	676	569	683
$(min)f_4$	7	6	7	4	**1**	4	5	3	2	4	1	7
$(min)f_5$	0	0	0	0	0	0	0	0	0	0	0	0

schedules may have the same objective function values but different assignments of OR time to SSs, OR blocks to surgical groups and/or different sets of scheduled surgeries. Moreover, the OR time assigned to an SS depends on its minimum and maximum required OR time (i.e. $Tmin^s$ and $Tmax^s$), on the maximum working time defined for surgeons $Tmax_t^s$, and on the waiting patients' status. Note that a surgeon will define how to sequence surgeries during the surgery day.

To de facto demonstrate the effectiveness of the proposed approach, we report in Table 8 specific details about the scenario E_{11} characterised by 10 OR blocks, 2 SSs and 4 surgeons. E_{11} is the simplest scenario and allows to synoptically visualize the behaviour of the conflicting objectives. Ten non-dominated Pareto solutions have been obtained. We observe that the first schedule gets the maximum number of scheduled surgeries (f_1) but the lowest value of f_3: in this case the number of scheduled surgeries increases at the expense of the number of scheduled "weighted" surgeries in $\tilde{\mathcal{P}}$ (the value of f_3). On the other end, the sixth schedule achieves the best value of f_3 but also the worst value of f_1. Further, it is worth to point out that the schedules 4-10 have the same number of scheduled surgeries. The underutilization is very low (value of function f_4) and no overtime blocks have been allotted (value of function f_5): this occurs because of the upper time limits defined for surgeons and/or their SS have been reached. More specifically, some differences are only among the values of f_2 and f_4: a worse value of f_2 generally corresponds to a better value of f_4 and vice-versa. The last two columns report the minimum and maximum optimal values of each objective function. For completeness and to show the main difference with a single objective approach, we have solved Model (2)–(13) by maximizing objective f_1 with the data of $OT_1 - I_1$. We run CPLEX and stopped it after 160 s. The found feasible solution has $f_1 = 89$ surgeries, $f_2 = 375$ (lower than the minimum reported in Table 8), 14 min of unused OR time and five overtime blocks assigned (note that no overtime blocks have been allotted in the Pareto solutions reported in Table 8).

Before to analyse the results, note that $OT_1 - OT_6$ can be viewed as 2, 3, 4, 5, 6 and 8 parallel ORs, respectively; $OT_7 - OT_9$ as 2, 3, 4 parallel ORs, respectively, but with two OR blocks in each OR. By comparing OT_1 with OT_7 (see Table 7) it is evident that they differ mainly in the number of available OR blocks and overtime blocks. OT_7 has less OR time than OT_1 but a double number of OR blocks: this implies a greater number of constraints and longer chromosomes, as reported in Table 7. An analogous comparison holds for the couples $OT_2 - OT_8$ and $OT_3 - OT_9$.

Table 9
Computational results: minimum and maximum values of the five objective functions, number *ND* of the non-dominated solutions, solution time (in min).

	$(max)f_1$		$(max)f_2$		$(max)f_3$		$(min)f_4$		$(min)f_5$			
Exp	min	max	min	max	min	max	min	max	min	max	*ND*	Solution time (min)
E_{11}	82	89	383	492	569	683	1	7	0	0	10	6
E_{12}	60	64	453	476	288	332	69	86	0	1	11	6
E_{13}	90	104	487	573	255	381	5	34	0	8	33	7
E_{14}	99	137	634	809	659	913	5	11	0	8	35	11
E_{22}	120	124	550	601	304	416	24	33	0	4	11	8
E_{23}	119	126	571	603	396	423	27	53	0	8	22	10
E_{24}	145	171	707	867	892	1078	7	14	0	8	17	17
E_{25}	174	219	931	1391	845	1224	3	9	0	1	13	6
E_{32}	98	118	476	560	350	417	53	76	0	2	8	11
E_{33}	127	128	596	605	421	425	159	168	0	2	4	11
E_{34}	129	192	848	1037	878	1207	5	34	0	7	16	25
E_{35}	197	201	1571	1601	1935	1960	19	27	0	3	13	33
E_{44}	170	196	877	968	952	1084	14	76	0	2	7	30
E_{45}	252	313	1441	1855	1461	2219	16	24	0	1	6	45
E_{54}	184	190	958	1027	1165	1207	227	234	0	0	3	31
E_{55}	262	354	1260	1926	1182	2189	18	53	0	1	5	64
$E_{55'}$	261	317	1260	1921	1182	2177	19	21	0	0	2	56
$E_{5'4}$	118	145	672	884	769	1121	15	40	0	4	15	31
E_{65}	407	408	1836	1848	1808	1881	66	71	0	0	3	87
E_{66}	245	512	1065	3271	1072	3700	19	35	0	9	5	150
E_{71}	67	79	290	470	272	652	12	20	0	0	11	10
E_{72}	73	99	427	563	191	365	4	23	0	8	13	12
E_{73}	78	99	436	564	194	361	9	29	0	7	15	12
E_{74}	73	87	433	489	657	717	8	15	0	1	16	21
E_{82}	102	109	530	582	288	383	27	44	0	4	10	18
E_{83}	102	108	499	584	269	388	38	85	0	7	20	18
E_{84}	104	140	636	878	736	1107	8	23	0	5	18	28
E_{85}	156	193	810	1273	754	1314	6	11	0	3	8	61
E_{93}	102	113	507	590	265	407	44	68	0	6	18	24
E_{94}	108	147	661	907	773	1082	14	29	0	4	10	33
E_{95}	189	240	813	1523	789	1859	14	30	0	5	5	83
$E_{95'}$	245	256	1065	1108	1072	1100	19	19	0	9	2	86

Table 9 summarizes the results of the computational experiments in terms of the minimum and maximum value of each objective function, and reports the related number of non-dominated solutions and the computational time (in min). As expected, the overall number of scheduled patients increases if OR time increases; however, the OR time distributed among surgeons could change in schedules with the same overall number of scheduled patients. In the following we discuss the results by analysing how the several instances reported in Table 5 have been settled in the different OTs of Table 4.

1. *Instance* I_1. It is the smallest demanded instance and has been solved in OT_1 and OT_7 by keeping almost the same level of total OR time. The lower granularity of the OR blocks of OT_1 allows to schedule a larger set of surgeries by a more efficient exploitation of the available OR time.
2. *Instance* I_2. The instance has been solved in OT_1, OT_2, OT_3, OT_7 and OT_8. More available OR time allows to efficiently treat the required levels of demand. OT_3 allows the maximum number of scheduled patients and also the lowest maximum number of overtime blocks.
3. *Instance* I_3. The instance is almost similar to the previous one, with slight differences in the total demanded OR time and number of surgical groups. The instance has been solved in $OT_1 - OT_3$ and $OT_7 - OT_9$. Note that OT_3 and OT_9 have more OR time and OR blocks than the other OTs: they allow then to schedule a higher number of surgeries with a smaller number of overtime blocks with respect to the other OTs.
4. *Instance* I_4. This instance has more demanded OR time and more patients than the previous instances. It has been solved in all OTs. Note that good results in terms of scheduled patients are found in OT_4, whereas OT_5 has the best results for prioritized surgeries (both f_2 and f_3): this result is explained by a different distribution of OR time among surgeons. Finally, we compare E_{54} and $E_{5'4}$ because OT_5 and $OT_{5'}$ differ in OR blocks duration: the overall underutilization in OT_5 is higher than the value in $OT_{5'}$ because OR time limits of surgeons have been reached and no more overtime OR blocks can be allotted to schedule more patients.
5. *Instance* I_5. This instance is more demanding than I_4 and has been solved in $OT_2 - OT_6$ and $OT_8 - OT_9$. The total number of scheduled surgeries is limited, in the different cases, by the capacities of the OTs. As expected, the best performance in terms of scheduled surgeries is obtained in OT_6, which is the biggest OT, but observe that in OT_6 the best value of f_3 is lower than the maximum value found in OT_5 (i.e., 1881 vs 2189). This unexpected result and the low number of found non-dominated solutions could be explained by the more longest chromosomes with OT_6 and then by a more huge solution space with respect to OT_5. A greater number of initial feasible solutions and more generations in NSGAII algorithm could be used to find a more accurate local Pareto front. Finally, we investigate how $Tmin^s$, $Tmax^s$, and $Tmax^s_t$ values influence the solutions of instance I_5 by slightly reducing their value: we have named as $I_{5'}$ this new instance. $I_{5'}$ has been solved in OT_5 and OT_9, as summarised in Table 7. In Table 9 we observe that the number of non-dominated solutions is low both in $E_{55'}$ and $E_{95'}$: this can be explained by a balanced distribution of OR blocks due to the used $Tmin^s$, $Tmax^s$, and $Tmax^s_t$ values.
6. *Instance* I_6. This most demanding instance is solved only in OT_6. The total number of scheduled surgeries is limited by the OR blocks capacities and by the maximum OR time allowable to surgeons. We can observe how the found set of non-dominated

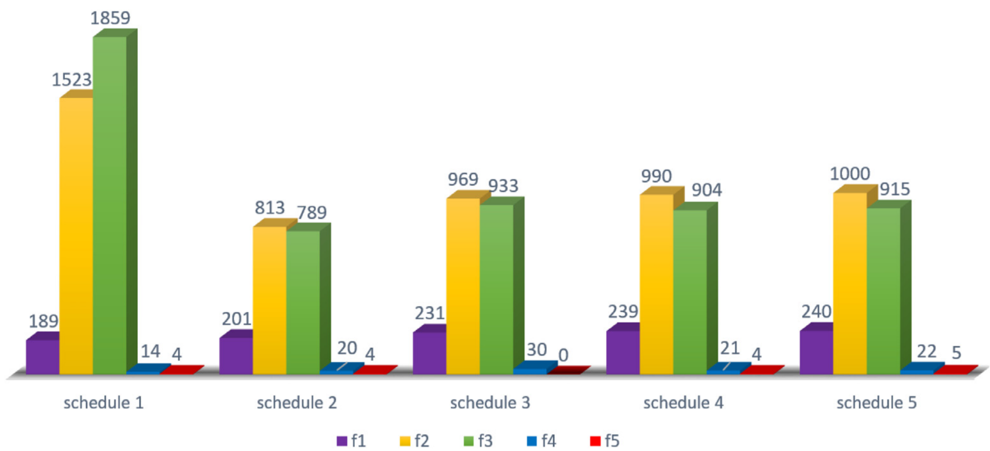

Fig. 5. A synoptical representation of the five non-dominated solutions of E_{95}.

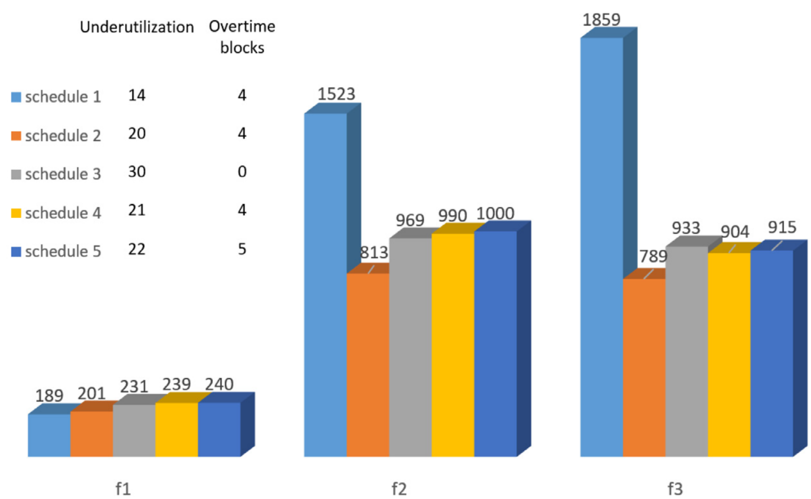

Fig. 6. Comparison of the first three objective function values of the five non-dominated solutions of E_{95}.

solutions allow to schedule patients ranging in number from 245 to 512, whereas sets of prioritised patients have a value in the range 1065-3271. The computational time is very long if compared with the other computational times.

Note that the number of non-dominated solutions decreases if a given instance is solved in OTs with increasing number of OR blocks (see, for instance, E_{14}, E_{24}, E_{34}, E_{44}, E_{54}). This is due to a "balanced distribution" of OR blocks among surgeons because of T^s_{max} values.

Our integrated approach is usefulness to generate Pareto solutions which are trade-offs among combinations of the considered multiple criteria, i.e. trade-offs among balanced distribution of OR time among surgeon groups based on patients status, minimization of surgeries' waiting times, underutilization of OR capacity and overtime working hours. Pareto solutions guarantee that no criterion can be improved without deteriorating another criterion and alternative schedules differ in care delivery and cost saving. A tractable number of non-dominated solutions can be represented in a diagram to help a decision-maker understand what a specific decision involves. Figs. 5 and 6 show the five non-dominated solutions of E_{95}, which is characterised by 40 OR blocks, 4 SSs, 8 surgeons, and 500 patients. Fig. 5 depicts the objective function values of the five schedules at the same time. Fig. 6 compares the values related to the first three objective functions, whereas the overall OR time underutilization (in min) and the number of overtime blocks are reported in correspondence of each schedule. We analyse three cases: (a) if the main goal of a decision-maker is to maximize the number of scheduled patients, that is the value of f_1, he/she chooses Schedule 5 but, as it easy to see, this decision/schedule implies the low value of f_3 and five overtime blocks; (b) if a current situation does not allow overtime blocks, there is only one schedule (i.e., Schedule 3); (c) if the main goal is to schedule those patients that should be operated during the planning horizon $\mathcal{H}$ (i.e. those in $\tilde{\mathcal{L}}^s_t$ set), the maximum value of f_3 is 1859 in correspondence of Schedule 1 but it implies the lowest number of scheduled patients (if compared with the other schedules) and four overtime blocks. Observe that, in these just analysed schedules, the underutilization values (i.e. f_4 values) are similar and could be neglectable in a decision-making process. Finally, we underline that each schedule differs from another one because of OR block distributions among SSs and surgeons, and scheduled patients. In case of Pareto sets with more numerous non-dominated solutions (i.e., more than ten), a decision support system (DSS) is helpful to support a decision-maker because can guide to decide which is the most suitable schedule among multiple solutions and then it can provide evidence on appropriate allocations of involved resources. Non-dominated solutions can be clustered, for instance, on the basis of specific values to simplify both an analysis and visualization of different alternatives, like those depicted in Figs. 5 and 6. Along these lines, it is more easy to select an appropriate schedule on the basis of specific situations and desiderata. Graphical representations about how ORs are allocated, how SSs requests are satisfied, how OR blocks are distributed among surgeons, which patients are scheduled and in which OR block are helpful to investigate how several schedules differ among them. A more sophisticated DSS could allow to analyse how overtime costs and underutilization of OR blocks change by selecting different schedules. Others and more complex DSS functionalities could be defined and implemented to make decisions more easily.

5. Conclusions

A crucial issue for effectively and efficiently tackling OR planning and scheduling problem is balancing surgical demands with available resources in a cost-effective manner. To this end, we have proposed an innovative approach for dealing with tactical and operational decision levels at the same time by allowing to update waiting list status, surgeon groups' availability and work hour of personnel for the next planning period.

The devised multi-objective optimization model yields a set of optimal decisions that aligns conflicting goals within an OT and related to hospital managers, surgeons, and patients. OR utilization balancing, maximization of scheduled elective surgeries by taking into account both surgery priority and waiting time, reduction of OR underutilization and overtime costs: all these aspects have not been considered to date at the same time. In fact, we have addressed, in an integrated way, MSS construction, admission planning problem and surgeries scheduling problem by matching relevant specific requirements.

In order to be rather effective in the solution approach and overcome convergence drawbacks of the NSGA-II algorithm, we have proposed and implemented a hybrid genetic algorithm. It focuses on a new initialization procedure that aims to construct an initial population with appropriate sets of feasible and semi-feasible chromosomes, which are seeds for the evolving computational process. The found Pareto solutions constitute a set of possible decisions that balance the conflicting goals and guarantee that no criterion can be improved without deteriorating a different one. Alternative solutions differ in quality of service delivery and cost savings. The overall map of multiple trade-off solutions could support managers and other actors involved in a decision process to make better choices. We are currently working on this matter.

The optimization model and the solution approach developed in this study may be the basis for further developments in a more effective and efficient OR scheduling system by considering other resources (such as ICU, PACU, and bed availability) and different objectives for measuring performance of an OR schedule.

Acknowledgement

The authors wish to thank the reviewers for their insightful comments that improved the quality of the paper.

References

[1] Dexter F, Epstein R, Traub R, Xiao Y. Making management decisions on the day of surgery based on operating room efficiency and patient waiting times. Anesthesiology 2004;101:53–1444.
[2] Sieber T, Leibundgut D. Operating room management and strategies in Switzerland: Results of a survey. Eur J Anaesthesiol 2002;19(6):415–23.
[3] Cardoen B, Demeulemeester E, Beliën J. Operating room planning and scheduling: A literature review. Eur J Oper Res 2010a;201(3):921–32.
[4] Guerriero F, Guido R. Operational research in the management of the operating theatre: a survey. Health Care Manag Sci 2011;14(1):89–114.
[5] Cardoen B, Demeulemeester E, Beliën J. Operating room planning and scheduling problems: a classification scheme. Int J Health Manag Inform 2010b;13(1):71–83.
[6] Pham D, Klinkert A. Surgical case scheduling as generalized job shop scheduling problem. Eur J Oper Res 2008;185(3):25–1011.
[7] Blake J, Carter M. A goal programming approach to strategic resource allocation in acute care hospitals. Eur J Oper Res 2002;140:541–61.
[8] Beliën J, Demeulemeester E. Building cyclic master surgery schedules with leveled resulting bed occupancy. Eur J Oper Res 2007;176(2):1185–204.
[9] van Oostrum JM, Bredenhoff E, Hans EW. Suitability and managerial implications of a master surgical scheduling approach. Annals Oper Res 2009;178:91–104.
[10] Ozkarahan I. Allocation of surgeries to operating rooms by goal programming. J Med Syst 2000;24(6):339–78.
[11] Guinet A, Chaabane S. Operating theatre planning. Int J Prod Econ 2003;85(1):69–81.
[12] Jebali A, Hadjalouane A, Ladet P. Operating rooms scheduling. Int J Prod Econ 2006;99:52–62.
[13] Kennedy M. Bin-packing, knapsack, and chance-constrained approaches to operating room scheduling. Ph.D. thesis Rensselaer Polytechnic Institute School Of Engineering Department of Decision Sciences Engineering Systems; 1992.
[14] Vissers J, Bertrand J, De Vries G. A framework for production control in health care organizations. Prod Plan Control 2001;12(6):591–604.
[15] Blake JT, Dexter F, Donald J. Operating room manager' s use of integer programming for assigning block time to surgical groups: A case study. Anesthesia Analgesia 2002;94:143–8.
[16] Zhang B, Murali P, Dessouky MM, Belson D. A mixed integer programming approach for allocating operating room capacity. J Oper Res Soc 2008;60(5):663–73.
[17] Marques I, Captivo M, Vaz Pato M. An integer programming approach to elective surgery scheduling. OR Spectrum 2012a;34(2):407–27.
[18] Conforti D, Guerriero F, Guido R. A multi-objective block scheduling model for the management of surgical operating rooms: New solution approaches via genetic algorithms. In: Proceedings of IEEE Workshop on Health Care Management (WHCM); 2010. p. 1–5.
[19] Marques I, Captivo ME, Vaz Pato M. Planning elective surgeries in a portuguese hospital: Study of different mutation rules for a genetic heuristic. Lect Notes Manag Sci 2012b;4:238–43.
[20] Testi A, Tanfani E, Torre G. A three-phase approach for operating theatre schedules. Health Care Manag Sci 2007;10:163–72.
[21] Testi A, Tanfani E. Tactical and operational decisions for operating room planning: Efficiency and welfare implications. Health Care Manag Sci 2009;12(4):363–73.
[22] Agnetis A, Coppi A, Corsini M, Dellino G, Meloni C, Pranzo M. A decomposition approach for the combined master surgical schedule and surgical case assignment problems. Health Care Manag Sci 2014;17(1):49–59.
[23] Ogulata S, Erol R. A hierarchical multiple criteria mathematical programming approach for scheduling general surgery operations in large hospitals. J Med Syst 2003;27(3):259–70.
[24] Cardoen B, Demeulemeester E, Beliën J. Optimizing a multiple objective surgical case sequencing problem. Int J Prod Econ 2009;119(2):354–66.
[25] Beliën J, Demeulemeester E, Cardoen B. A decision support system for cyclic master surgery scheduling with multiple objectives. J Sched 2009;12(2):147–61.
[26] Marler RT, Arora JS. The weighted sum method for multi-objective optimization: new insights. Struct Multidisc Optim 2010;41(6):853–62.
[27] Meskens N, Duvivier D, Hanset A. Multi-objective operating room scheduling considering desiderata of the surgical team. Dec Support Syst 2013;55(2):650–9.
[28] Erdogan SA, Denton BT. Surgery planning and scheduling. John Wiley & Sons, Inc.; 2011. ISBN 9780470400531. doi:10.1002/9780470400531.eorms0861.
[29] Oduncu F. Priority-setting, rationing and cost-effectiveness in the German health care system. Med Health Care Philos 2013;16(3):39–327.
[30] Agnetis A, Coppi A, Corsini M, Dellino G, Meloni C, Pranzo M. Long term evaluation of operating theater planning policies. Oper Res Health Care 2012;1(4):95–104.
[31] Coello C. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. Comput Meth Appl Mech Eng 2002;19(11-12):1245–87.
[32] Konak A, Coit DW, Smith AE. Multi-objective optimization using genetic algorithms: A tutorial. Reliab Eng Syst Saf 2006;91(9):992–1007.
[33] Coello C. Evolutionary multi-objective optimization: some current research trends and topics that remain to be explored. Front Comput Sci China 2009;3:18–30.
[34] Ranjithan SR, Chetan SK, Dakshina HK. Constraint method-based evolutionary algorithm for multiobjective optimization. In: Zitzler E, Thiele L, Deb K, Coello Coello C, Corne D, editors. Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization. *Lecture Notes in Computer Science*, Vol. 1993. Springer Berlin / Heidelberg; 2001. p. 299–313.
[35] Deb K, Agrawal S, Pratap A, Meyarivan T. A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans Evolut Comput 2002;6(2):182–97.
[36] Abraham A, Jain LC, Goldberg R. Evolutionary multiobjective optimization: Theoretical advances and applications (Advanced Information and Knowledge Processing). NJ, USA: Springer-Verlag New York, Inc. Secaucus; 2005.
[37] Glover FW, Kochenberger GA. Handbook of metaheuristics. In: International Series in Operations Research & Management Science; 2003.
[38] Holland JH. Adaptation in natural and artificial systems. Ann Arbor, MI: The University of Michigan Press; 1975.
[39] Mitchell M. An introduction to genetic algorithms. MIT Press; 1997.
[40] Ehrgott M, Gandibleux X. Hybrid metaheuristics for multi-objective combinatorial optimization. In: Blum C, Aguilera M, Roli A, Sampels M, editors. Hybrid Metaheuristics. *Studies in Computational Intelligence*, Vol. 114. Springer Berlin / Heidelberg; 2008. p. 221–59.
[41] Zitzler E, Deb K, Thiele L. Comparison of multiobjective evolutionary algorithms: Empirical results. Evol Comput 2000;8(2):173–95.

[42] Srinivas N, Deb K. Multiobjective optimization using nondominated sorting in genetic algorithms. Evol Comput 1994;2:221–48.
[43] Laredo J, Eiben A, Steen M, Castillo P, Mora A, Merelo J. P2P evolutionary algorithms: A suitable approach for tackling large instances in hard optimization problems. In: Luque E, Margalef T, Ben Atez D, editors. Euro-Par 2008 - Parallel Processing. *Lecture Notes in Computer Science*, Vol. 5168. Springer Berlin Heidelberg; 2008. p. 622–31.
[44] Rahnamayan S, Tizhoosh HR, Salama MMA. A novel population initialization method for accelerating evolutionary algorithms. Comput Math Appl 2007;53(10):1605–14.
[45] Chandoul W, Hammadi S, Camus H, Zgaya H, di Pompeo C, Trincaretto F. Evolutionary approach for multi-objective scheduling in surgical unit. In: Conférence Francophone Gestion et Ingénierie des Systémes Hospitaliers (GISEH 2012), Quebec; 2012. p. 1–8.
[46] Puchinger J, Raidl GR. Combining metaheuristics and exact algorithms in combinatorial optimization: a survey and classification. In: Mira J, lvarez J, editors. Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach. *Lecture Notes in Computer Science*, Vol. 3562. Springer Berlin Heidelberg; 2005. p. 41–53.